Analysis and development of finite volume methods for the new generation of cubed sphere dynamical cores for the atmosphere

Luan da Fonseca Santos

Thesis presented to the
Institute of Mathematics and Statistics
of the University of São Paulo
in partial fulfillment
of the requirements
for the degree of
Doctor of Science

Program: Applied Mathematics
Advisor: Prof. Pedro da Silva Peixoto

During the development of this work the author was supported by CAPES and FAPESP (grant number 20/10280-4)

São Paulo
July, 2023

Analysis and development of finite volume methods for the new generation of cubed sphere dynamical cores for the atmosphere

Luan da Fonseca Santos

This is the original version of the thesis prepared by candidate Luan da Fonseca Santos, as submitted to the Examining Committee.

*Education is what remains after one has for-
gotten everything he learned in school.*
— Albert Einstein

Acknowledgements

TBW

Resumo

Luan da Fonseca Santos. **Análise e desenvolvimento de métodos de volumes finitos para modelos da nova geração da dinâmica atmosférica baseados na esfera cubada**. Tese (Doutorado). Instituto de Matemática e Estatística, Universidade de São Paulo, São Paulo, 2023.

TBW

Palavras-chave: Núcleo dinâmico da atmosfera, esfera cubada, volumes finitos, dimension splitting, ponto de partida, corretor de massa.

Abstract

Luan da Fonseca Santos. **Analysis and development of finite volume methods for the new generation of cubed sphere dynamical cores for the atmosphere**. Thesis (Doctorate). Institute of Mathematics and Statistics, University of São Paulo, São Paulo, 2023.

TBW

Keywords: Dynamical core, cubed-sphere, finite-volume, dimension splitting, departure point, mass fixer.

List of abbreviations and acronyms

CFL	Courant–Friedrichs–Lewy
CS	Cubed-sphere
dg1	Lagrange interpolation method on duo-grid based on geodesic distances
dg2	Lagrange interpolation method on duo-grid based on the local coordinate system distances
DP1	First-order departure point
DP2	Second-order departure point
FV	Finite Volume
FV3	Finite-Volume Cubed-Sphere Dynamical Core
g0	Equiedge cubed-sphere
g1	Equidistant cubed-sphere
g2	Equiangular cubed-sphere
GFDL	Geophysical Fluid Dynamics Laboratory
hord0	Non-monotonic PPM reconstruction scheme
hord8	Monotonic PPM reconstruction scheme
IC	Initial Condition
LT	Average Lie-Trotter advection scheme
NOAA	National Oceanic and Atmospheric Administration
MPI	Message Passing Interface
ODE	Ordinary Differential Equation
PDE	Partial Differential Equation
PL	Putman and Lin advection scheme
SL	Semi-Lagrangian
VF	Velocity field

Contents

Appendixes

Chapter 1

Introduction

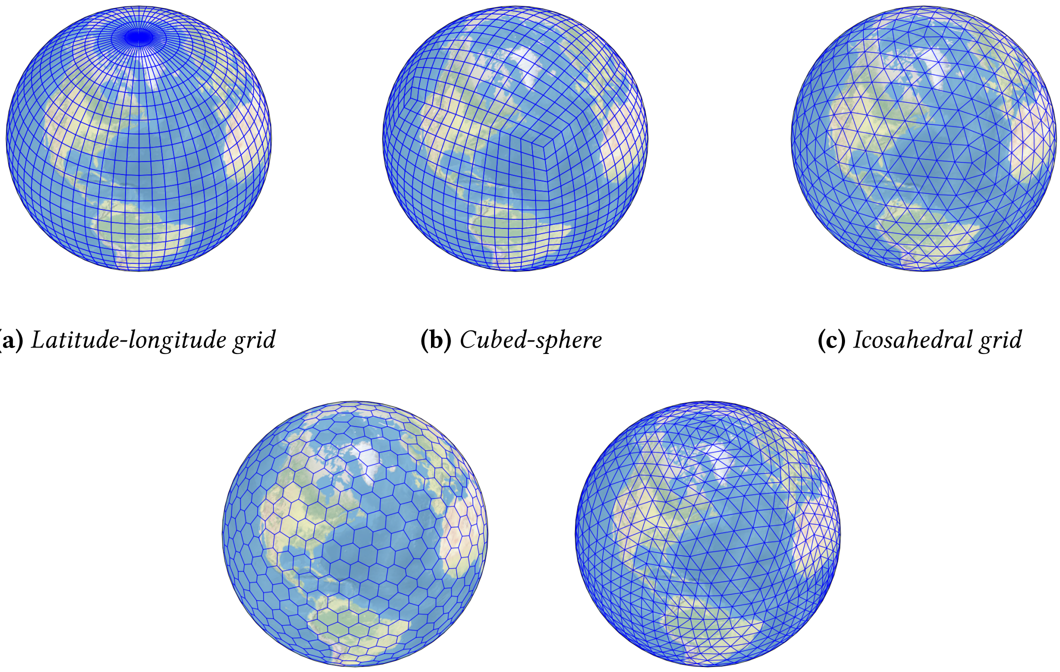
1.1 Background

Weather and climate predictions are recognized as a good for mankind, due to the information they yield for diverse activities. For instance, short-range forecasts are useful for public use, while medium-range forecasts are helpful for industrial activities and agriculture. Seasonal forecasts (one up to three months) are important to energy planning and agriculture. At last, longer-range forecasts (one century, for instance) are useful for climate change projections that are important for government planning.

The first global Numerical Weather Prediction models emerged in the 1960s with applications to weather, seasonal and climate forecasts. All these applications are essentially based on the same set of Partial Differential Equations (PDEs) but with distinct time scales (Staniforth & Wood, 2008). These PDEs are defined on the sphere and model the evolution of the atmospheric fluid given the initial conditions. One important component of global models is the dynamical core, which is responsible for solving the PDEs that governs the atmosphere dynamics on grid-scale (D. L. Williamson, 2007). The development of numerical methods for dynamical cores has been an active research area since the 1960s.

Global models use the sphere as the computational domain and therefore they require a discretization of the sphere. The first global models used the latitude-longitude grid (Figure 1.1a), which is very suitable for finite-differences schemes due to its orthogonality (D. L. Williamson, 2007). The major drawback of the latitude-longitude grid is the clustering of points at the poles, known as the "pole problem", which leads to extremely small time steps for explicit-in-time schemes due to the Courant-Friedrichs-Lewy (CFL) condition, making these schemes computationally very expensive (D. Randall, 2022).

The most successful method adopted in global atmospheric dynamical cores that overcomes the CFL restriction is the Semi-Implicit Semi-Lagrangian (SI-SL) scheme (D. A. Randall et al., 2018), which emerged in the 1980s and consists of the Lagrangian advection scheme applied at each time-step and the solution of fast gravity waves implicitly, allowing very large time steps despite the pole problem. The SI-SL approach combined with finite differences is still used nowadays, for instance in the UK Met Office global model ENDGame (Wood et al., 2014). The expensive part of the SI-SL approach is to solve an elliptic equation at each time step, that comes from the semi-implicit discretization, which requires global data communication, being inefficient to run in massive parallel supercomputers. Besides that, Semi-Lagrangian schemes are inherently non-conservatives for mass, which is critical for climate forecasts (D. L. Williamson, 2007).



(a) *Latitude-longitude grid* **(b)** *Cubed-sphere* **(c)** *Icosahedral grid*

(d) *Pentagonal/Hexagonal grid* **(e)** *Octahedral grid*

Figure 1.1: *Examples of spherical grids: latitude-longitude grid (a) and grids based on Platonic solids (b)-(d).*

The emergence of the Fast Fourier Transform (FFT) in the 1960s with the work from Cooley and Tukey (1965) allowed the computation of discrete Fourier transforms with $N\log(N)$ complexity. The viability of the usage of FFTs for solving atmospheric flows was shown by Orszag (1970), using the barotropic vorticity equation on the sphere, and by Eliasen et al. (1970), using the primitive equations. The spectral transform method expresses latitude-longitude grid values, that represent some scalar field, using truncated spherical harmonics expansions, which consists of Fourier expansions in latitude circles and Legendre functions expansions in longitude circles. The coefficients in the spectral expansions are known as spectral coefficients and are usually thought to live in the so-called spectral space. Given the grid values, the spectral coefficients are obtained by performing a FFT followed by a Legendre Transform (LT). Conversely, given the spectral coefficients, the grid values are obtained by performing an inverse LT followed by an inverse FFT. The main idea of the spectral method is to apply the spectral transform, in order to go the spectral space, and evaluate spatial derivatives in the spectral space, which consists of multiplying the spectral coefficients by constants. Then, the method performs the inverse spectral transform in order to get back to grid space, and the nonlinear terms are treated on the grid space (Krishnamurti et al., 2006).

The spectral transform makes the use of SI-SL methods computationally cheap, since the solution to elliptic problems becomes easy, once the spherical harmonics are eigenfunctions of the Laplacian operator on the sphere. Therefore, the spectral transform method gets faster when combined with the SI-SL approach due to the larger times-steps allowed in this case. Due to these enhancements, the spectral transform dominated global atmospheric modeling (D. A. Randall et al., 2018) since the 1980s. Indeed, the spectral method is still used in many current operational Weather Forecasting models such as the Integrated Forecast System (IFS) from European Centre for Medium-Range Weather Forecasts (ECMWF), Global Forecast System (GFS) from National Centers for Environmental Prediction (NCEP) and the Brazilian Global Atmospheric Model (BAM) (Figueroa et al., 2016) from Center for Weather Forecasting and Climate Research [Centro de Previsão de Tempo e Estudos Climáticos (CPTEC)].

With the beginning of the multicore era in the 1990s, the global atmospheric models started to move towards parallel efficiency aiming to run at very high resolutions. Even though the spectral transform expansions have a global data dependency, some parallelization is feasible among all the computations of FFTs, LTs and their inverses (Barros et al., 1995). However, the parallelization of the spectral method requires data transpositions in order to compute FFTs and LTs in parallel. These transpositions demand a lot of global communication using, for instance, the Message Passing Interface (MPI) (Zheng & Marguinaud, 2018). Indeed, the spectral transform becomes the most expensive component of global spectral models when the resolution is increased due to the amount of MPI communications (Müller et al., 2019).

The adiabatic and frictionless continuous equations that govern the atmospheric flow have conserved quantities. Among them, some of the most important are mass, total energy, angular momentum and potential vorticity (Thuburn, 2011). Numerical schemes that are known for having discrete analogous of these conservative properties are known as mimetic schemes. As we pointed out, Semi-Lagrangian schemes lack mass conservation. Nevertheless, these schemes have been employed in dynamical cores for better computational performance. However, dynamical cores should have discrete analogous of the continuous conserved quantities, especially concerning for longer simulation runs.

Aiming for better performance in massively parallel computers and conservation properties, new dynamical cores have been developed since the beginning of the 2000s. Novel spherical grids have been proposed, in order to avoid the pole problem. A popular choice are grids based on Platonic solids (Staniforth & Thuburn, 2012). The construction of these grids relies on a Platonic circumscribed on the sphere and the projection of its faces onto the sphere, which leads to quasi-uniform and more isotropic spherical grids. Some examples of spherical grids based on Platonic solids employed in the new generation of dynamical cores are the cubed-sphere (Figure 1.1b), icosahedral grid (Figure 1.1c), the pentagonal/hexagonal or Voronoi grid (Figure 1.1d) and octahedral grid (Figure 1.1e), which are based on the cube, icosahedron, dodecahedron and octahedron, respectively (Ullrich et al., 2017).

1.2 Motivations and the FV3 model

The cubed-sphere became a popular quasi-uniform grid for the new generation of dynamical cores. It was originally proposed by Sadourny (1972) and it was revisited by Ronchi et al. (1996). Some of the cubed-sphere advantages are: uniformity; quadrilateral structure, making the grid indexing trivial; no overlappings; it is cheap to generate. However, the major drawbacks of the cubed-sphere are: non-orthogonal coordinate system, which leads to metric terms on the differential operators; discontinuity of the coordinate system at the cube edges, which may generate numerical noise and demands special treatment of discrete operators at the cube edges.

Despite of its drawbacks, the cubed-sphere has been adopted in some of the new generation dynamical cores. For instance, the cubed-sphere is used in the Community Atmosphere Model (CAM-SE) from the NCAR using spectral elements (Dennis et al., 2012) and in the Nonhydrostatic Unified Model of the Atmosphere (NUMA) from the US Navy using Discontinuous Galerkin methods (Giraldo et al., 2013). The cubed-sphere was also chosen to be used in the next UK Met Office global model using mixed finite elements (Kent et al., 2023). At last, the Finite Volume Cubed-Sphere dynamical core (FV3) from the Geophysical Fluid Dynamics Laboratory (GFDL) and the National Oceanic and Atmospheric Administration (NOAA) (L. M. Harris & Lin, 2013; Putman & Lin, 2007) is another example of new generation dynamical core based on the cubed-sphere.

The FV3 dynamical core is an extension of the Finite-Volume dynamical core (FVcore) from latitude-longitude grids to the cubed-sphere. The numerical methods from FVcore started to be developed with the transport scheme from the work Lin et al. (1994), which is based on the piecewise linear scheme from Van Leer (1977). This scheme was later improved, using the Piecewise Parabolic Method (PPM) (Carpenter et al., 1990; Colella & Woodward, 1984) using dimension splitting techniques that guarantee monotonicity and mass conservation, for the transport equation (Lin & Rood, 1996) and the shallow-water equations (Lin & Rood, 1997). An important feature is that the FVcore combines the Arakawa C- and D-grids (Arakawa & Lamb, 1977), where the C-grid values are computed in and intermediate time step. The full global model was then presented by Lin (2004).

A computational disadvantage of the FVcore is its Semi-Lagrangian formulation that creates more demand for MPI communication when computing the accumulated fluxes (Lin & Rood, 1996). The FVcore was then adapted to the cubed-sphere grid (Putman, 2007; Putman & Lin, 2007), to reach better performance in parallel computers, leading to the FV3 model. Later, the FV3 also was improved to allow locally refinement grids through grid-nesting or grid-stretching (L. M. Harris & Lin, 2013).

Currently, the FV3 model is capable of perfoming hydrostatic and non-hydrostatic atmospheric simulations and it was chosen as the new US global weather prediction model, indeed, it replaced the spectral transform Global Forecast System (GFS) in June, 2019 (https://www.noaa.gov/media-release/noaa-upgrades-us-global-weather-forecast-model, last access: 19th of March, 2024). Additionally, the FV3 dynamical core is employed in the GEOS Chem model (Martin et al., 2022) from Harvard University, in NASA's next-generation Mars Climate Model (Wilson et al., 2022), and also in the SHiELD model from GFDL (L. Harris et al., 2020).

However, a well-known problem that occurs on cubed-sphere models that use low-order numerical methods is the grid imprinting visible due to the coordinate system discontinuity, especially at larger scales, leading to the emergence of a wavenumber 4 pattern. This was reported in the paper of Rančić et al. (2017), where the authors employ a finite-difference numerical scheme on the Uniform Jacobian cubed-sphere using a Arakawa B-grid. The unpublished report from Whitaker (2015) shows grid imprinting in other models, including the FV3. More recently, Mouallem et al. (2023) has shown some idealized simulations using FV3 where grid imprinting appears in many simulations.

Generally speaking, grid imprinting is the presence of artificial behaviors on the numerical solution that is associated with the grid employed. It is important to stress out that other quasi-uniform grids may also suffer from grid imprinting. For instance, a popular mimetic method, known as TRiSK, was proposed in the literature by Thuburn et al. (2009) and Ringler et al. (2010) using finite difference and finite volume schemes. This scheme is designed for general orthogonal grids, such as the Voronoi and icosahedral grids, and ensures mass and total energy conservation. This method has been employed in the dynamical core of the Model for Prediction Across Scales (MPAS) from National Center for Atmospheric Research (NCAR) (W. Skamarock et al., 2012), which intended to work on general Voronoi grids, including locally refined Voronoi grids. However, the TRiSK scheme is a low-order scheme and also suffers from grid imprinting, *i.e.*, geometric properties of the grid, such as cell alignment, interfere with the method accuracy (P. Peixoto, 2016; P. Peixoto & Barros, 2013; Weller, 2012).

Despite being chosen as the new US global weather prediction model, there is a lack of numerical studies of the FV3 discretizations in the literature, especially regarding the grid imprinting problem and its mimetic properties. Numerical results for the advection equation on the cubed-sphere using the FV3 dynamical core was presented in Putman and Lin (2007) and some shallow-water simulations were presented in L. M. Harris and Lin (2013), considering cubed-spheres with local refinement through grid nesting. From the work L. M. Harris and Lin (2013) we can notice that the FV3 dynamical lack convergence on the maximum norm for the shallow-water model considering the classical balanced geostrophic flow test case from D. Williamson et al. (1992). The authors attribute these errors to the abrupt change in the grid resolution near the nested grid, but no quantitative results are shown considering the quasi-uniform grid. Many other papers available in the literature use the complete FV3 model in three-dimensional frameworks which make it harder to perform a numerical analysis study due not only to its computational cost but also due to the complexity of three-dimensional atmospheric models. There are no detailed works published in intermediate two-dimensional frameworks, using, for instance, the shallow-water equations on the sphere. Even though the advection equation on the sphere plays a key role in the dynamical core development, since it models the transport of scalar fields on the sphere, important features captured by the shallow-water equations on the sphere, such as the Coriolis effect, inertia-gravity waves, geostrophic adjustment, Rossby waves, among others, are not captured by a simple advection model. Hence, shallow-water equations provide an excellent benchmark to assess dynamical cores in general, since it is only two-dimensional but is a complex enough geophysical model for atmosphere dynamics.

1.3 Outline and contributions

This thesis is outlined as follows.

- Chapter 2 is dedicated to review the Piecewise Parabolic Method (PPM) for the one-dimensional advection equation. In this Chapter, we demonstrate how the temporal component of this PPM can be expressed as a departure point calculation. Subsequently, we enhance the departure point calculation from first order (which is utilized in FV3) to second order. This enhancement results in a significant improvement, particularly in non-constant wind simulations. The additional cost is only due to linear interpolation, which has little impact on the overall performance.
- Chapter 3 reviews the dimension splitting method, which allow us to use one-dimensional methods, such as the PPM, to solve the two-dimensional advection equation.
- Chapter 4 introduces the cubed-sphere grid and shows how we can compute stencils near to the cubed-edges using 1D Lagrange interpolation through the duogrid technique.
- Chapter 5 extends the ideas of Chapter 3 to the cubed-sphere grid. The dimension-splitting method on each cubed-sphere panel works as in the plane, with the addition of metric terms, due to non-orthogonality of the grid, and interpolation between panels to obtain ghost cells values needed for stencil computations.

We give some future work perspectives in Chapter 7.

Chapter 2

One-dimensional finite-volume methods

The aim of this Chapter is to provide a detailed description of one-dimensional (1D) finite-volume (FV) schemes within a Semi-Lagrangian (SL) framework, specifically applied to the 1D advection equation. These schemes are also known as flux-form Semi-Lagrangian schemes, and they allow for time steps beyond the Courant-Friedrichs-Lewy (CFL) condition while preserving the total mass. FV-SL schemes have been explored in the literature since the work of LeVeque (1985), which extended the finite-volume schemes from Godunov (1959) to accommodate larger time steps. This approach has been further investigated in the literature (c.f, e.g. . Leonard et al. (1996) and Lin and Rood (1996)). We are going to focus on the linear advection equation because in FV3, the horizontal dynamics are solved by using flux advection operators to compute the fluid density, absolute vorticity, and the kinetic energy (L. Harris et al., 2021; L. M. Harris & Lin, 2013; Lin & Rood, 1997; Putman, 2007). The boundary conditions are assumed to be periodic for simplicity.

To introduce the FV-SL schemes, we begin by discretizing the spatial and temporal domains into uniform grids. Subsequently, the FV-SL schemes involve three steps. The first step involves computing the departure points of the spatial grid edges. The second step, known as reconstruction, utilizes the grid cell average values to determine a piecewise function within each cell. This piecewise function approximates the values of the advected quantity and ensures the preservation of its local mass within each grid cell. The third step involves updating the fluxes at the grid edges by integrating the reconstruction function over a domain that extends from the departure point of the grid edge to the grid edge itself.

The first step of FV-SL schemes can be accomplished by integrating an ordinary differential equation (ODE) backward in time. The second step is performed using the Piecewise-Parabolic Method (PPM) proposed by Colella and Woodward (1984). As the name suggests, PPM employs piecewise-parabolic functions. The third and final step is computed easily, as the reconstruction functions consist of parabolas that preserve the local mass.

It is worth noting that the reconstruction function can be constructed using functions other than parabolas. In fact, PPM can be seen as an extension of the Piecewise-Linear method proposed by Van Leer (1977), which, in turn, was inspired by the Piecewise-Constant method introduced by Godunov (1959). Additionally, other schemes inspired by PPM have been proposed in the literature utilizing higher-order polynomials, such as quartic polynomials (White & Adcroft, 2008). For a comprehensive review of general piecewise-polynomial reconstruction, we recommend referring to the technical report by Engwirda and Kelley (2016), Lauritzen et al. (2011), and the references therein.

The PPM approach has become popular in the literature for gas dynamics simulations, astrophysical phenomena modeling (Woodward, 1986), and later on atmospheric simulations (Carpenter et al., 1990). Indeed, PPM has been implemented in the FV3 dynamical core on its latitude-longitude grid (Lin, 2004) and cubed-sphere (Putman & Lin, 2007) versions. Although many other shapes for the basis functions and higher-order schemes are available in the literature, L. Harris et al. (2021) points out that the PPM scheme suits the needs of FV3 well. It is a flexible method that can be modified to ensure low diffusivity or shape preservation, for example. Additionally, a finite-volume numerical method usually requires monotonicity constraints, which, according to Godunov's order barrier theorem (Wesseling, 2001), limit the order of convergence to at most 1. Therefore, a higher-order scheme needs to strike a well-balanced trade-off between increasing computational cost and potential benefits.

This Chapter begins with a basic review of one-dimensional advection equation in the integral form in Section 2.1. In Section 2.2, we establish the framework for general one-dimensional finite-volume Semi-Lagrangian schemes. Section 2.3 presents methods for computing the departure point. The PPM reconstruction is described in Section 2.4, while Subsection 2.4.2 introduces a different approach to ensure the monotonicity of parabolas. Section 2.5 focuses on the description and investigation of the PPM flux computation. Section 2.6 presents numerical results using the PPM scheme for the advection equation. Finally, Section 2.7 presents some concluding remarks. The application of PPM to solve two-dimensional problems will be addressed in Chapter 3.

2.1 One-dimensional advection equation in integral form

2.1.1 Notation

Before introducing the FV-SL schemes, let us establish some notation by introducing the concepts of a Δx-grid, a Δt-temporal grid, and the $(\Delta x, \Delta t, \lambda)$-discretization, as well as the concept of grid function/winds. In this Chapter, we will use the notation $\Omega = [a, b]$ to represent the interval under consideration, and ν to represent a non-negative integer indicating the number of ghost cell layers in each boundary. We also use the notations $\mathbb{R}^N_\nu := \mathbb{R}^{N+2\nu}$ and $\mathbb{R}^{N+1}_\nu := \mathbb{R}^{N+1+2\nu}$.

Definition 2.1 (Δx-grid)**.** *For a given interval Ω and a positive real number Δx such that $\Delta x = (b-a)/N$ for some positive integer N, we say that $\Omega_{\Delta x} = \{X_i\}_{i=-\nu+1}^{N+\nu}$ is a Δx-grid for Ω*

if

$$X_i = [x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}] = [a + (i-1)\Delta x, a + i\Delta x],$$

and $\Delta x = x_{i+\frac{1}{2}} - x_{i-\frac{1}{2}}$. *Each* X_i *is referred to as a control volume or cell, and* $x_{i-\frac{1}{2}}$ *and* $x_{i+\frac{1}{2}}$ *are the edges of the control volume* X_i. *The cell centroid is defined by*

$$x_i = \frac{1}{2}(x_{i+\frac{1}{2}} + x_{i-\frac{1}{2}}), \quad \forall i = -\nu + 1, \dots, N + \nu,$$

and Δx *is the cell length.*

Remark 2.1. *If* $1 \le i \le N$, *we refer to* i *as an interior index; otherwise,* i *is considered a ghost cell index and we say the* X_i *is a ghost cell.*

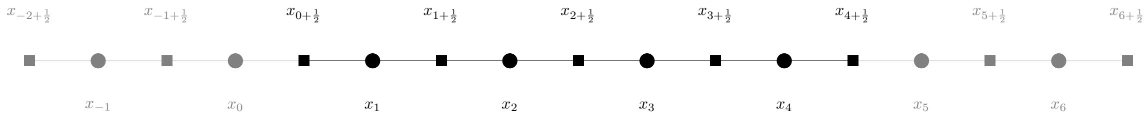


Figure 2.1: *Illustration of a* Δx*-grid with* $N = 4$ *cells in its interior (in black) and* $\nu = 2$ *ghost cell layers (in gray). The edges are denoted by squares and the cell centroids are denoted using circles.*

Definition 2.2 (Δt-temporal grid)**.** *For a given interval* $[0, T]$ *and a positive real number* Δt *such that* $\Delta t = T/N_T$ *for some positive integer* N_T, *we say that* $T_{\Delta T} = \{T_n\}_{n=0}^{N_T}$ *a* Δt*-temporal grid for* $[0, T]$ *if*

$$T_n = [t^n, t^{n+1}], \quad t^n = n\Delta t, \quad \Delta t = \frac{T}{N_T}, \quad \forall n = 0, \dots, N_T.$$

In this context, we also define $t^{n+\frac{1}{2}} = \frac{t^n + t^{n+1}}{2}$.

Definition 2.3 ($(\Delta x, \Delta t, \lambda)$-discretization)**.** *Given* $\Omega \times [0, T]$ *and positive real numbers* Δx *and* Δt, *we say that* $(\Omega_{\Delta x}, T_{\Delta t})$ *is a* $(\Delta x, \Delta t, \lambda)$*-discretization of* $\Omega \times [0, T]$ *if* $\Omega_{\Delta x}$ *is a* Δx*-grid for* Ω, $T_{\Delta t}$ *is a* Δt*-temporal grid for* $[0, T]$, *and* $\frac{\Delta t}{\Delta x} = \lambda$.

Remark 2.2. *Whenever we refer to a* Δx*-grid, a* Δt*-temporal grid, or a* $(\Delta x, \Delta t, \lambda)$*-discretization,* X_i, N, t^n, *and* N_T *are assumed to be implicitly defined.*

Next, we introduce the definitions of grid functions at cell centroids and edges.

Definition 2.4 (Δx-grid function)**.** *For a* Δx*-grid, we say that* Q *is a* Δx*-grid function if* $Q = (Q_{-\nu+1}, \dots, Q_{N+\nu}) \in \mathbb{R}^N_\nu$.

Definition 2.5 (Δx-grid wind)**.** *For a* Δx*-grid, we say that* u *is a* Δx*-grid wind if* $u = (u_{-\nu+\frac{1}{2}}, \dots, u_{N+\nu+\frac{1}{2}}) \in \mathbb{R}^{N+1}_\nu$.

The definition of a Δx-grid wind is based on the Arakawa grids (Arakawa & Lamb, 1977). Considering functions $q, u : \Omega \times [0, T] \to \mathbb{R}$ and a $(\Delta x, \Delta t, \lambda)$-discretization of $\Omega \times [0, T]$, we introduce the grid functions $q^n \in \mathbb{R}^N_\nu$ and $u^n \in \mathbb{R}^{N+1}_\nu$. Here, $q^n_i = q(x_i, t^n)$ and $u^n_{i+\frac{1}{2}} = u(x_{i+\frac{1}{2}}, t^n)$. These grid functions represent the discrete values of q and u at the cell centroids and edges, respectively, for each time level t^n (Figure 2.2).

In this Chapter, our focus lies on periodic grid functions. We define a Δx-grid function Q as periodic if it satisfies the following conditions:

$$
\begin{aligned}
Q_i &= Q_{N+i}, \quad i = -\nu + 1, \dots, 0, \\
Q_i &= Q_{i-N}, \quad i = N + 1, \dots, N + \nu.
\end{aligned}
$$

Similarly, we define a Δx-grid wind as periodic if it meets the following requirements:

$$
\begin{aligned}
u_{i-\frac{1}{2}} &= u_{N+i+\frac{1}{2}}, \quad i = -\nu, \dots, -1, \\
u_{i+\frac{1}{2}} &= u_{i+\frac{1}{2}-N}, \quad i = N + 1, \dots, N + \nu.
\end{aligned}
$$

We use the notation $\mathbb{P}_\nu^N$ and $\mathbb{P}_\nu^{N+1}$ to represent the spaces of periodic Δx-grid functions and winds, respectively.

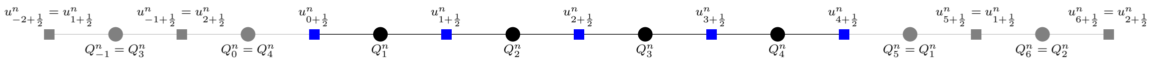


Figure 2.2: *Illustration of Δx-grid function Q (black circles) and a Δx-grid wind u (blue squares) and its ghost cell values (in gray) assuming periodicity.*

Given $Q \in \mathbb{P}_\nu^N$, we define the p-norm as

$$
\|Q\|_{p,\Delta x} = \begin{cases} \left(\sum_{i=1}^{N} |Q_i|^p \right)^{\frac{1}{p}} & \text{if } 1 \leq p < \infty, \\ \max_{i=1,\dots,N} |Q_i| & \text{otherwise}, \end{cases} \tag{2.1}
$$

which is indeed a norm for periodic grid functions. Using a similar notation as in Engwirda and Kelley (2016), we define the stencil and a grid function evaluated on a stencil as follows.

Definition 2.6 (Stencil)**.** *For a Δx-grid, and each $i = 0, \dots, N$, we define a stencil as a set of the form $\mathcal{S}_{i+\frac{1}{2}} = \{i - r + 1, \dots, i - 1, i, i + 1, \dots, i + s\} \subset \{-\nu + 1, \dots, N + \nu\}$.*

Definition 2.7 (Grid function restricted to a stencil)**.** *For a Δx-grid, a stencil $\mathcal{S}_{i+\frac{1}{2}}$, and a Δx-grid function Q, we define $Q(\mathcal{S}_{i+\frac{1}{2}}) = (Q_k)_{k \in \mathcal{S}_{i+\frac{1}{2}}}$.*

These definitions provide the necessary notation for describing grid functions and their evaluations on stencils. To achieve a more compact notation in some situations, we introduce the centered difference notation:

$$
\delta_x g(x_i, t) = g(x_{i+\frac{1}{2}}, t) - g(x_{i-\frac{1}{2}}, t), \tag{2.2}
$$

for any function $g : \Omega \times [0, T] \to \mathbb{R}$. Additionally, we introduce the average value of q in the i-th control volume at time t, denoted as $Q_i(t)$, defined by:

$$
Q_i(t) = \frac{1}{\Delta x} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} q(x, t)\, dx. \tag{2.3}
$$

Moreover, we define the Δx-grid function of average values as $Q(t) = (Q_i(t))_{i=-\nu+1}^{N+\nu}$. Here,

$Q_i(t)$ represents the average value of q in the i-th control volume at time t.

For the consideration of periodic boundary conditions, we can define spaces of periodic functions over the interval Ω as follows:

$$\mathcal{S}_P(\Omega) = \{q : \mathbb{R} \times [0, +\infty[\to \mathbb{R} \ : \ q(x + b - a, t) = q(x, t), \quad \forall x \in \mathbb{R}, \quad t \geq 0\}.$$

Similarly, the space of k-times periodically differentiable functions $C_P^k(\Omega)$ can be defined as:

$$C_P^k(\Omega) = \mathcal{S}_P(\Omega) \cap C^k(\mathbb{R} \times [0, \infty[),$$

where $C^k(\mathbb{R} \times [0, +\infty[)$ denotes the space of functions that are k times continuously differentiable in both the spatial and temporal variables. In summary, $\mathcal{S}_P(\Omega)$ represents the space of periodic functions, and $C_P^k(\Omega)$ represents the space of k-times periodically differentiable functions over the interval Ω subject to periodic boundary conditions.

2.1.2 The 1D advection equation

In this Section, we will derive the integral form of the 1D advection equation with periodic boundary conditions over the interval Ω. What is going to be presented here follows LeVeque (1990, 2002) closely. The advection equation with periodic boundary conditions, is the given by the following PDE:

$$\begin{cases} [\partial_t q + \partial_x (uq)](x, t) = 0, \quad \forall (x, t) \in \mathbb{R} \times]0, +\infty[, \\ q(a, t) = q(b, t), \quad \forall t \geq 0, \\ q_0(x) = q(x, 0), \quad \forall x \in \Omega. \end{cases} \tag{2.4}$$

Here, $q \in C_P^1(\Omega)$ represents the advected quantity, and $u \in C_P^1(\Omega)$ represents the velocity or wind. We will focus on Equation (2.4) over the domain $D = \Omega \times [0, T]$, where $T > 0$ is a finite time. A strong or classical solution to the advection equation is defined as a function $q \in C_P^1(\Omega)$ and satisfies Equation (2.4). In order to deduce the integral form of Equation (2.4), we consider $[x_1, x_2] \times [t_1, t_2] \subset D$. Integrating Equation (2.5) over $[x_1, x_2]$, we obtain:

$$\frac{d}{dt} \int_{x_1}^{x_2} q(x, t)\, dx = -((uq)(x_2, t) - (uq)(x_1, t)), \tag{2.5}$$

and integrating Equation (2.5) over $[t_1, t_2]$, we get

$$\int_{x_1}^{x_2} q(x, t_2)\, dx = \int_{x_1}^{x_2} q(x, t_1) - \left(\int_{t_1}^{t_2} (uq)(x_2, t)\, dt - \int_{t_1}^{t_2} (uq)(x_1, t)\, dt \right). \tag{2.6}$$

The presented problem, Problem 2.1, aims to find a solution, called weak solution, to the advection equation in its integral form, considering the given initial condition (IC) q_0 and velocity function u.

Problem 2.1. *Given an IC q_0 and a velocity function u we would like to find a weak solution*

q of the advection equation in the integral form:

$$\int_{x_1}^{x_2} q(x,t_2)\,dx = \int_{x_1}^{x_2} q(x,t_1)\,dx + \int_{t_1}^{t_2} (uq)(x_1,t)\,dt - \int_{t_1}^{t_2} (uq)(x_2,t)\,dt,$$

$\forall [x_1,x_2]\times[t_1,t_2] \subset \Omega\times[0,T]$, *and* $q(x,0)=q_0(x)$, $\forall x\in\Omega$, $q(a,t)=q(b,t)$, $\forall t\in[0,T]$.

We point out that, for Problem 2.1, the total mass in Ω at time t defined by:

$$M_{[a,b]}(t) = \int_a^b q(x,t)\,dx,$$

remains constant over time, i.e.,

$$M_{[a,b]}(t) = M_{[a,b]}(0), \quad \forall t\in[0,T].$$

This conservation of total mass property is highly desirable for numerical schemes aiming to approximate general conservation law solutions accurately.

Applying the steps from Equation (2.4) to Equation (2.6) in reverse order, one can verify that if q is a weak solution and $q\in C_P^1(\Omega)$, then it satisfies Equation (2.4). Therefore, Equation (2.4) and Problem (2.1) are equivalent when $q\in C_P^1(\Omega)$. However, Problem (2.1) can be formulated for functions that are not C^1 and have discontinuities. In fact, Problem (2.1) only requires that q and uq are locally integrable.

It is worth noting that Equation (2.6) holds for all x_1, x_2, t_1, and t_2 such that $[x_1,x_2]\times[t_1,t_2]\subset D$. Therefore, let us consider a $(\Delta x,\Delta t,\lambda)$-discretization of D and rewrite Equation (2.6) in terms of this discretization. By replacing t_1, t_2, x_1, and x_2 with $t^n, t^{n+1}, x_{i-\frac{1}{2}}$, and $x_{i+\frac{1}{2}}$, respectively, in Equation (2.6), we obtain:

$$Q_i(t^{n+1}) = Q_i(t^n) - \frac{1}{\Delta x}\left(\int_{t^n}^{t^{n+1}} (uq)(x_{i+\frac{1}{2}},t)\,dt - \int_{t^n}^{t^{n+1}} (uq)(x_{i-\frac{1}{2}},t)\,dt\right), \quad \forall i=1,\dots,N, \quad \forall n=0,\dots,N_T-1. \tag{2.7}$$

To achieve a more compact notation, we use the centered difference notation and then Equation (2.7) can be rewritten as:

$$Q_i(t^{n+1}) = Q_i(t^n) - \frac{1}{\Delta x}\delta_x\left(\int_{t^n}^{t^{n+1}} (uq)(x_i,t)\,dt\right), \quad \forall i=1,\dots,N, \quad \forall n=0,\dots,N_T-1. \tag{2.8}$$

Now we can define a discretized version of Problem 2.1 as Problem 2.2.

Problem 2.2. *Let us consider the framework of Problem 2.1 and a $(\Delta x,\Delta t,\lambda)$-discretization of $\Omega\times[0,T]$. Since we are operating within the framework of Problem 2.1, the following relationship holds:*

$$Q_i(t^{n+1}) = Q_i(t^n) - \lambda\delta_x\left(\frac{1}{\Delta t}\int_{t^n}^{t^{n+1}} (uq)(x_i,t)\,dt\right), \quad \forall i=1,\dots,N, \quad \forall n=0,\dots,N_T-1, \tag{2.9}$$

where $Q_i(t) = \frac{1}{\Delta x}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} q(x,t)\,dx$. Our objective now is to determine the values $Q_i(t^n)$, $\forall i = 1,\dots,N$, $\forall n = 0,\dots,N_T - 1$, given the initial values $Q_i(0)$, $\forall i = 1,\dots N$. In other words, we aim to find the average values of q in each control volume X_i at the specified time instances.

It is important to note that no approximations have been made in problems (2.1) and (2.2). In Equation (2.9), we divided and multiplied by Δt to interpret $\frac{1}{\Delta t}\int_{t^n}^{t^{n+1}} (uq)(x_{i\pm\frac{1}{2}}, t)\,dt$ as a time-averaged flux. This interpretation is useful for deriving finite-volume schemes.

In Problem 2.2, we need 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, we can express the temporal integral 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 $s \in [t^n, t^{n+1}]$, we consider the following Cauchy problem backward in time:

$$\begin{cases} \partial_t x^d_{i+\frac{1}{2}}(t,s) = u(x^d_{i+\frac{1}{2}}(t,s), t), & t \in [t^n, s] \\ x^d_{i+\frac{1}{2}}(s,s) = x_{i+\frac{1}{2}}. \end{cases} \tag{2.10}$$

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 Figure 2.3 we illustrate the departure point idea.

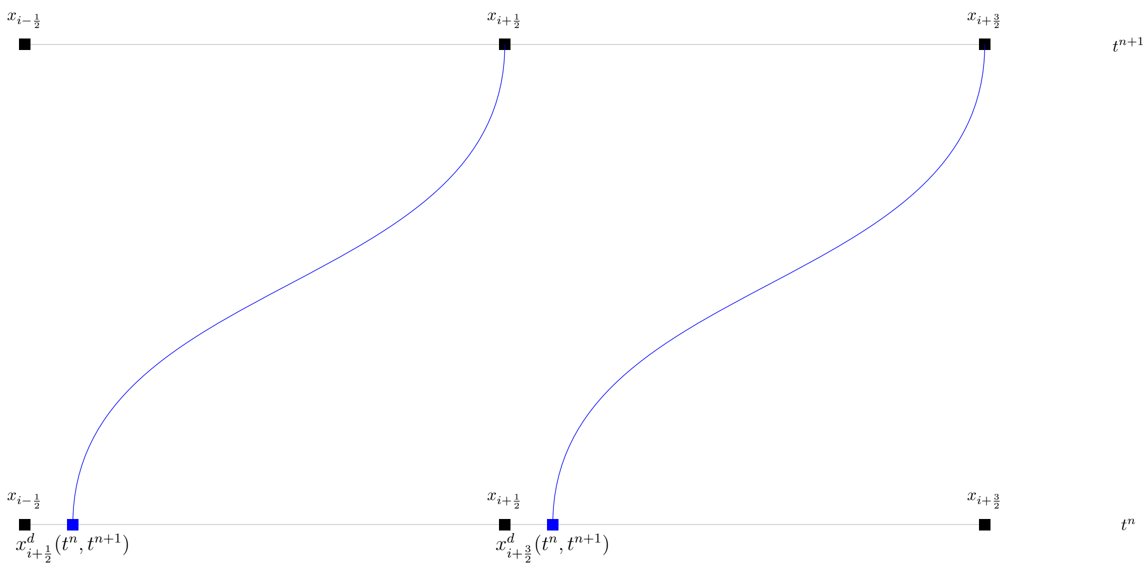


Figure 2.3: *Illustration of the departure point of the cell edges from time t^{n+1} to t^n.*

Integrating Equation (2.10) over the interval $[t, s]$, we get:

$$x^d_{i+\frac{1}{2}}(t,s) = x_{i+\frac{1}{2}} - \int_t^s u(x^d_{i+\frac{1}{2}}(\theta, s), \theta)\,d\theta. \tag{2.11}$$

In the following Proposition, we show how the time-averaged flux is related to a spatial integral over a interval depending on departure points.

Proposition 2.1. *Assume the framework of Problem 2.2. If q and u are C^1 functions, then:*

$$\int_{t^n}^{t^{n+1}} (uq)(x_{i+\frac{1}{2}}, s)\, ds = \int_{x^d_{i+\frac{1}{2}}(t^n,t^{n+1})}^{x_{i+\frac{1}{2}}} q(x, t^n)\, dx \tag{2.12}$$

Proof. Using the Leibniz rule for integration (Theorem A.3 with $f(s,\theta) = u\big(x^d_{i+\frac{1}{2}}(\theta, s), \theta\big)$), in Equation (2.11), it follows that:

$$\begin{aligned} \partial_s x^d_{i+\frac{1}{2}}(t,s) &= -\left(u(x_{i+\frac{1}{2}}, s) + \int_t^s \partial_s u\big(x^d_{i+\frac{1}{2}}(\theta,s),\theta\big)\, d\theta \right) \\ &= -u(x_{i+\frac{1}{2}}, s) - \int_t^s \partial_x u\big(x^d_{i+\frac{1}{2}}(\theta,s),\theta\big)\partial_s x^d_{i+\frac{1}{2}}(\theta,s)\, d\theta. \end{aligned} \tag{2.13}$$

Taking the derivative with respect to t of Equation (2.13), we have:

$$\partial_t\partial_s x^d_{i+\frac{1}{2}}(t,s) = \partial_x u\big(x^d_{i+\frac{1}{2}}(t,s),t\big)\partial_s x^d_{i+\frac{1}{2}}(t,s). \tag{2.14}$$

Using standard ODE's techniques, we get that $x^d_{i+\frac{1}{2}}$ that solves Equations (2.13) and (2.14) is given by:

$$\partial_s x^d_{i+\frac{1}{2}}(t,s) = -\exp\left(\int_t^s \partial_x u\big(x^d_{i+\frac{1}{2}}(\theta,s),\theta\big)\, d\theta \right) u(x_{i+\frac{1}{2}}, s). \tag{2.15}$$

Computing q on the trajectory give by $x^d_{i+\frac{1}{2}}(t,s)$ and taking its time derivative, we obtain:

$$\begin{aligned} \frac{d}{dt} q\big(x^d_{i+\frac{1}{2}}(t,s),t\big) &= \partial_t q\big(x^d_{i+\frac{1}{2}}(t,s),t\big) + (u\partial_x q)\big(x^d_{i+\frac{1}{2}}(t,s),t\big) \\ &= -\partial_x u\big(x^d_{i+\frac{1}{2}}(t,s),t\big) q\big(x^d_{i+\frac{1}{2}}(t,s),t\big), \end{aligned} \tag{2.16}$$

where we used that q satisfies the linear advection equation on its differential (2.4) form and that $x^d_{i+\frac{1}{2}}(t,s)$ solves Equation (2.10). Using again standard ODE techniques, we get that q that solves Equation (2.16) is given by:

$$q\big(x^d_{i+\frac{1}{2}}(t,s),t\big) = \exp\left(-\int_t^s \partial_x u\big(x^d_{i+\frac{1}{2}}(\theta,s),\theta\big)\, d\theta \right) q(x_{i+\frac{1}{2}}, s). \tag{2.17}$$

Notice that if u does not depend on x, then q is constant along the trajectory $x^d_{i+\frac{1}{2}}(t,s)$.

Let us consider the mapping $s \in [t^n, t^{n+1}] \to x^d_{i+\frac{1}{2}}(t^n, s)$. Integrating q over all departure points at time t^n from $x_{i+\frac{1}{2}}$ at time s, we have

$$\int_{x^d_{i+\frac{1}{2}}(t^n,t^n)=x_{i+\frac{1}{2}}}^{x^d_{i+\frac{1}{2}}(t^n,t^{n+1})} q(x,t^n)\, dx = \int_{t^n}^{t^{n+1}} q\big(x^d_{i+\frac{1}{2}}(t^n,s),t^n\big)\partial_s x^d_{i+\frac{1}{2}}(t^n,s)\, ds, \tag{2.18}$$

where we are just using the variable change integration formula. Then, it follows from

Equations (2.15) and (2.17) with $t = t^n$ that:

$$\int_{x_{i+\frac{1}{2}}}^{x^d_{i+\frac{1}{2}}(t^n,t^{n+1})} q(x,t^n)\,dx = -\int_{t^n}^{t^{n+1}} (uq)(x_{i+\frac{1}{2}},s)\,ds, \tag{2.19}$$

which is the desired formula. □

With the aid of Proposition 2.1, we can rewrite Problem 2.2 in terms of the departure point, avoiding the need for knowledge about q over the entire interval $[t^n, t^{n+1}]$. This is described in Problem 2.3:

Problem 2.3. *Assume the framework of Problem 2.1 and a $(\Delta x, \Delta t, \lambda)$-discretization of $\Omega \times [0,T]$. Since we are in the framework of Problem 2.1, it follows that:*

$$Q_i(t^{n+1}) = Q_i(t^n) - \lambda\left(\frac{1}{\Delta t}\int_{x^d_{i+\frac{1}{2}}(t^n,t^{n+1})}^{x_{i+\frac{1}{2}}} q(x,t^n)\,dx - \frac{1}{\Delta t}\int_{x^d_{i-\frac{1}{2}}(t^n,t^{n+1})}^{x_{i-\frac{1}{2}}} q(x,t^n)\,dx\right), \tag{2.20}$$
$$\forall i = 1,\dots,N, \quad \forall n = 0,\dots,N_T - 1,$$

where $Q_i(t) = \frac{1}{\Delta x}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} q(x,t)\,dx$. Our problem now consists of finding the values $Q_i(t^n)$, $\forall i = 1,\dots,N$, $\forall n = 0,\dots,N_T-1$, given the initial values $Q_i(0)$, $\forall i = 1,\dots N$. In other words, we would like to find the average values of q in each control volume X_i at the considered time instants.

At each time step t^n, we compute the values of $Q_i(t^{n+1})$ based on $Q_i(t^n)$ and the integrals of $q(x,t^n)$ over specific intervals. These intervals are defined by the departure points $x^d_{i+\frac{1}{2}}(t^n,t^{n+1})$ and $x^d_{i-\frac{1}{2}}(t^n,t^{n+1})$. To perform the computations, we need to determine the departure points from the edges of all control volumes and calculate the required integrals. This idea serves as the motivation for defining finite-volume Semi-Lagrangian schemes. These schemes involve estimating the departure points and reconstructing the function q at time t^n using its average values $Q_i(t^n)$, which enables us to compute the necessary integrals.

2.2 The finite-volume Semi-Lagrangian approach

Finally, we define the 1D FV-SL scheme problem as follows in Problem 2.3.

Problem 2.4 (1D FV-SL scheme)**.** *Assume the framework defined in Problem 2.3. The finite-volume Semi-Lagrangian approach of Problem 2.3 consists of finding a scheme of the form:*

$$Q_i^{n+1} = Q_i^n - \lambda(F^n_{i+\frac{1}{2}} - F^n_{i-\frac{1}{2}}), \quad \forall i = 1,\dots,N, \quad \forall n = 0,\dots,N_T-1, \tag{2.21}$$

where $Q^n \in \mathbb{P}^N_\nu$ is intended to be an approximation of $Q(t^n) \in \mathbb{P}^N_\nu$ in some sense. We define $Q_i^0 = Q_i(0)$ or $Q_i^0 = q_i^0$. The terms $F^n_{i\pm\frac{1}{2}}$ are known as numerical flux and are given by

$$F^n_{i\pm\frac{1}{2}} = \frac{1}{\Delta t}\int_{\tilde{x}^n_{i\pm\frac{1}{2}}}^{x_{i\pm\frac{1}{2}}} \tilde{q}(x;Q^n)\,dx, \tag{2.22}$$

where $\tilde{x}^n_{i\pm\frac{1}{2}}$ is an estimate of the departure point $x^d_{i-\frac{1}{2}}(t^n, t^{n+1})$, and $\tilde{q}$ is a reconstruction function for q built with the values Q^n. Thus, $F^n_{i\pm\frac{1}{2}}$ approximates $\frac{1}{\Delta t}\int_{x^d_{i\pm\frac{1}{2}}(t^n,t^{n+1})}^{x_{i\pm\frac{1}{2}}} q(x,t^n)\,dx$.

For a 1D FV-SL the discrete total mass at the time-step n is given by

$$M^n = \Delta x \sum_{i=1}^{N} Q_i^n. \tag{2.23}$$

Therefore, the discrete total mass is constant for a 1D-FV scheme, which follows from a straightforward computation:

$$M^{n+1} = \Delta x \sum_{i=1}^{N} Q_i^{n+1} = M^n - \Delta t \sum_{i=1}^{N} (F^n_{i+\frac{1}{2}} - F^n_{i-\frac{1}{2}}) = M^n - \Delta t (F^n_{N+\frac{1}{2}} - F^n_{\frac{1}{2}}) = M^n,$$

where we are using that $F^n_{N+\frac{1}{2}} = F^n_{\frac{1}{2}}$, since we are assuming periodic boundary conditions.

We would like to highlight an important relationship between the average values of q and its values at the cell centroids. In Problem 2.4, we mentioned that the IC can be represented as q_i^0 instead of $Q_i(0)$. Moreover, when analyzing the convergence of a FV-SL scheme, it is useful to compare Q_i^n with q_i^n since computing $Q_i(t^n)$ requires evaluating an analytical integral, which can be challenging in certain cases. In Proposition 2.2, we provide a simple proof that q_i^n approximates $Q_i(t^n)$ with second-order error when q is twice continuously differentiable.

Proposition 2.2. *If $q \in C^2_P(\Omega)$, then $Q_i(t^n) - q_i^n = C_1 \Delta x^2$, where $C_1 = \frac{1}{24}\frac{\partial^2 q}{\partial x^2}(\eta, t^n)$, $\eta \in X_i$.*

Proof. Just apply Theorem A.4 for the function $q(x,t^n)$. □

Hence, 1D FV-SL schemes may be conceptualized as schemes that update the centroid values. The Problem of the convergence of 1D FV-SL schemes is addressed in Section A.3. Now we are going to address the problem of the departure point estimation and the reconstruction problem.

2.3 Departure point computation

Before presenting estimates for the departure point, let us recall the definition of the CFL number.

Definition 2.8. *For Problem 2.4, the CFL number at an edge $x_{i+\frac{1}{2}}$ and at a time level t^n is defined by*

$$c^n_{i+\frac{1}{2}} = \frac{\Delta t}{\Delta x} u^n_{i+\frac{1}{2}}. \tag{2.24}$$

The CFL number is the maximum of the values $c^n_{i+\frac{1}{2}}$. The CFL number at edges and at time levels $n + \frac{1}{2}$ is defined in the same manner. The problem of estimating the departure point is very common in Semi-Lagrangian schemes, which are quite popular in atmospheric modeling. For a review of departure point calculation methods, we refer to Tumolo (2011, Chapter 3) and the references therein. There are different approaches to compute the departure point, such as integrating the ODE from Equation (2.1) using different time integrators (D. Durran, 2011) backward in time. The Runge-Kutta methods are a possible choice to compute the departure point (*cf.* *e.g.* Guo et al. (2014), Lu et al. (2022)).

Equation (2.11) enables us to compute or estimate the departure point. For instance, if u is constant, the departure point at time t^n for the point $x_{i+\frac{1}{2}}$ at time t^{n+1} is given by:

$$x^d_{i+\frac{1}{2}}(t^n, t^{n+1}) = x_{i+\frac{1}{2}} - u\Delta t. \tag{2.25}$$

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

$$\tilde{x}^n_{i+\frac{1}{2}} = x_{i+\frac{1}{2}} - \tilde{u}^n_{i+\frac{1}{2}}\Delta t, \tag{2.26}$$

where $\tilde{u}^n_{i+\frac{1}{2}}$ represents the time-averaged wind and approximates:

$$\frac{1}{\Delta t}\int_{t^n}^{t^{n+1}} u(x^d_{i+\frac{1}{2}}(\theta, t^{n+1}), \theta)\, d\theta. \tag{2.27}$$

The departure point $\tilde{x}^n_{i+\frac{1}{2}}$ is said to be p-order accurate if:

$$x^d_{i+\frac{1}{2}}(t^n, t^{n+1}) - \tilde{x}^n_{i+\frac{1}{2}} = \mathcal{O}(\Delta t^p). \tag{2.28}$$

2.3.1 DP1 scheme

One possible way of estimating the time-averaged wind is by using:

$$\tilde{u}^n_{i+\frac{1}{2}} = u^{n+\frac{1}{2}}_{i+\frac{1}{2}}, \tag{2.29}$$

as in FV3 papers (Lin & Rood, 1996; Putman & Lin, 2007). In this case, the time-averaged CFL is given by:

$$\tilde{c}^n_{i+\frac{1}{2}} = c^{n+\frac{1}{2}}_{i+\frac{1}{2}}, \tag{2.30}$$

For simplicity, in this Chapter, we shall assume that the wind is known for all time instants needed. This scheme will be referred to as **DP1**. In FV3, the wind is at time level $n + \frac{1}{2}$ is obtained by solving the horizontal dynamics on a C-grid as an intermediate step (Lin, 2004; Lin & Rood, 1997). Our objective now is to determine the value of p in Equation (2.28) in the following proposition. It is useful to introduce the concept of a material derivative beforehand:

$$\frac{Dh}{Dt} = \frac{\partial h}{\partial t} + u\frac{\partial h}{\partial x},$$

where h is a function belonging to C^1.

Proposition 2.3. *If $u \in C^1$ and the time-averaged wind is computed using Equation* (2.29), *then the departure point from Equation (2.26) satisfies:*

$$x^d_{i+\frac{1}{2}}(t^n, t^{n+1}) - \tilde{x}^n_{i+\frac{1}{2}} = \mathcal{O}(\Delta t^2), \tag{2.31}$$

for a constant C that depends on u.

Proof. Using the midpoint rule (Theorem A.4) for the function $f(t) = u\big(x^d_{i+\frac{1}{2}}(t, t^{n+1}), t\big)$ in Equation (2.11), we obtain:

$$x^d_{i+\frac{1}{2}}(t^n, t^{n+1}) = x_{i+\frac{1}{2}} - u\big(x^d_{i+\frac{1}{2}}(t^{n+\frac{1}{2}}, t^{n+1}), t^{n+\frac{1}{2}}\big)\Delta t - \frac{1}{24}\frac{D^2u}{Dt^2}\big(x^d_{i+\frac{1}{2}}(\theta_1, t^{n+1}), \theta_1\big)\Delta t^2, \tag{2.32}$$

for $\theta_1 \in [t^n, t^{n+1}]$. Now observe that, from the intermediate value theorem for integrals and Equation (2.11), we have

$$x^d_{i+\frac{1}{2}}(t^{n+\frac{1}{2}}, t^{n+1}) = x_{i+\frac{1}{2}} - \frac{\Delta t}{2} u\big(x^d_{i+\frac{1}{2}}(\theta_2, t^{n+1}), \theta_2\big)$$

for $\theta_2 \in [t^{n+\frac{1}{2}}, t^{n+1}]$. Combining this with a Taylor's expansion of $u\big(x^d_{i+\frac{1}{2}}(t, t^{n+1}), t^{n+\frac{1}{2}}\big)$ for $t = t^{n+\frac{1}{2}}$, we have:

$$u\big(x^d_{i+\frac{1}{2}}(t^{n+\frac{1}{2}}, t^{n+1}), t^{n+\frac{1}{2}}\big) = u^{n+\frac{1}{2}}_{i+\frac{1}{2}} - \left(u\frac{\partial u}{\partial x}\right)\big(x_{i+\frac{1}{2}}(\theta_3, t^{n+1}), t^{n+\frac{1}{2}}\big)\big)u\big(x^d_{i+\frac{1}{2}}(\theta_2, t^{n+1}), \theta_2\big)\frac{\Delta t^2}{2}, \tag{2.33}$$

for $\theta_3 \in [t^n, t^{n+1}]$. Substituting Equation (2.33) into Equation (2.32), we obtain the desired estimate. □

2.3.2 DP2 scheme

In this work, we shall consider a second-order Runge-Kutta method to compute the departure point, which we express in terms of $\tilde{u}^n_{i+\frac{1}{2}}$ using the following equations (D. R. Durran, 2010):

$$\begin{aligned} \tilde{x}^{n+\frac{1}{2}}_{i+\frac{1}{2}} &= x_{i+\frac{1}{2}} - u^n_{i+\frac{1}{2}}\frac{\Delta t}{2} = x_{i+\frac{1}{2}} - c^n_{i+\frac{1}{2}}\frac{\Delta x}{2}, \\ \tilde{u}^n_{i+\frac{1}{2}} &= u\left(\tilde{x}^{n+\frac{1}{2}}_{i+\frac{1}{2}}, t^n + \frac{\Delta t}{2}\right). \end{aligned} \tag{2.34}$$

Notice that this scheme requires values of u at points that are not grid points, both in space. We overcome this using linear interpolation in space:

$$\tilde{u}^n_{i+\frac{1}{2}} = \begin{cases} \big(1 - \alpha^n_{i+\frac{1}{2}}\big)u^{n+\frac{1}{2}}_{i+\frac{1}{2}-k} + \alpha^n_{i+\frac{1}{2}}u^{n+\frac{1}{2}}_{i-\frac{1}{2}-k} & \text{if } u^n_{i+\frac{1}{2}} \geq 0, \\ \alpha^n_{i+\frac{1}{2}}u^{n+\frac{1}{2}}_{i+\frac{3}{2}-k} + \big(1 - \alpha^n_{i+\frac{1}{2}}\big)u^{n+\frac{1}{2}}_{i+\frac{1}{2}-k} & \text{if } u^n_{i+\frac{1}{2}} < 0, \end{cases} \tag{2.35}$$

where $\frac{c^n_{i+\frac{1}{2}}}{2} = \alpha^n_{i+\frac{1}{2}} + k$, $k = \lfloor \frac{c^n_{i+\frac{1}{2}}}{2} \rfloor$, $\alpha^n_{i+\frac{1}{2}} \in [0, 1[$, and $\lfloor \cdot \rfloor$ is the floor function. This scheme leads to a third-order error in the departure point estimate (see *e.g.* D. R. Durran (2010, Section 7.1.2)). This scheme shall be referred to as **DP2**. Notice that for this scheme, we need ghost values for the velocity, depending on how large the CFL number is. In particular, if the CFL number is less than 2, then $k = 0$ and we need the ghost values $u^n_{-1+\frac{1}{2}}$ and $u^n_{N+\frac{3}{2}}$. In this case, it useful to work with the time-averaged CFL number:

$$\tilde{c}^n_{i+\frac{1}{2}} = \begin{cases} \left(1 - \frac{c^n_{i+\frac{1}{2}}}{2}\right) c^{n+\frac{1}{2}}_{i+\frac{1}{2}} + \frac{c^n_{i+\frac{1}{2}}}{2} c^{n+\frac{1}{2}}_{i-\frac{1}{2}} & \text{if } c^n_{i+\frac{1}{2}} \geq 0, \\ \frac{c^n_{i+\frac{1}{2}}}{2} c^{n+\frac{1}{2}}_{i+\frac{3}{2}} + \left(1 - \frac{c^n_{i+\frac{1}{2}}}{2}\right) c^{n+\frac{1}{2}}_{i+\frac{1}{2}} & \text{if } c^n_{i+\frac{1}{2}} < 0. \end{cases} \tag{2.36}$$

2.4 Reconstruction: the Piecewise-Parabolic Method

In this Section, we will review the Piecewise-Parabolic Method (PPM). The analysis of its accuracy will be presented in Section A.6. PPM was originally proposed by Colella and Woodward (1984) for gas dynamic simulations, and its applicability to atmospheric simulations has been demonstrated by Carpenter et al. (1990). This method is based on utilizing parabolas to reconstruct the function using its average values, ensuring both mass conservation and monotonicity. PPM is an extension of the Piecewise-Linear Method introduced by Van Leer (1977), and it is implemented in the FV3 model using the dimension splitting method developed by Lin and Rood (1996).

Let's consider a function q defined in $\Omega = [a, b]$ and a Δx-grid covering Ω. We assume that we are given the average values $Q_i = \frac{1}{\Delta x} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} q(x)\, dx$ for each control volume X_i, where $i = 1, \dots, N$. In this context, it is convenient to define the Δx-grid function $Q \in \mathbb{P}^N_\nu$ with the entries given by Q_i. To facilitate the discussion, we introduce the indicator function $\chi_i(x)$ for each control volume X_i, defined as:

$$\chi_i(x) = \begin{cases} 1 & \text{if } x \in X_i, \\ 0 & \text{otherwise.} \end{cases}$$

Drawing inspiration from Stoer and Bulirsch (2002, Chapter 1), we consider a family of functions $\Phi(\xi; \mu)$ defined for $\xi \in [0, 1]$, depending on a parameter $\mu = (\mu_0, \mu_1, \dots, \mu_d) \in \mathbb{R}^{d+1}$. The reconstruction problem involves finding a piecewise function:

$$\tilde{q}(x; Q) = \sum_{i=1}^{N} \chi_i(x) q_i(x; Q), \tag{2.37}$$

where $q_i(x; Q) = \Phi\big(\frac{x - x_{i-\frac{1}{2}}}{\Delta x}; \alpha_i\big)$ and $\alpha_i = (\alpha_{i0}, \alpha_{i1}, \dots \alpha_{id}) \in \mathbb{R}^{d+1}$. It is required that:

$$\frac{1}{\Delta x} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \tilde{q}(x; Q)\, dx = \frac{1}{\Delta x} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} q_i(x; Q)\, dx = \int_0^1 \Phi(\xi; \alpha_i)\, d\xi = Q_i,$$

which means that $q_i(x; Q)$ preserves the mass within each control volume X_i.

Notice that, given $q_i(x; Q) = \Phi\big(\frac{x - x_{i-\frac{1}{2}}}{\Delta x}; \alpha_i\big)$, it is reasonable to expect that $\Phi(0; \alpha_i)$ approximates $q_i(x_{i-\frac{1}{2}})$ and $\Phi(1;\alpha_i)$ approximates $q_i(x_{i+\frac{1}{2}})$. Additionally, if both q and Φ are sufficiently differentiable, $\Phi^{(l)}(0;\alpha_i)$ should approximate $(\Delta x)^l q^{(l)}(x_{i-\frac{1}{2}})$ and $\Phi^{(l)}(1;\alpha_i)$ should approximate $(\Delta x)^l q^{(l)}(x_{i+\frac{1}{2}})$, provided these derivatives exist.

One approach to estimating these values at the edges $x_{i+\frac{1}{2}}$ using the average values Q is by employing a reconstruction method based on primitive functions (LeVeque, 2002, Chapter 17). It is worth noting that if we define:

$$Q(x) = \int_a^x q(\xi)\, d\xi, \tag{2.38}$$

we have $Q^{(l)}(x) = q^{(l-1)}(x)$. Specifically, $Q^{(l)}(x_{i+\frac{1}{2}}) = q^{(l-1)}(x_{i+\frac{1}{2}})$ and $Q(x_{i+\frac{1}{2}}) = \Delta x \sum_{k=1}^{i} Q_k$, for all $i = 0, \dots, N$. Therefore, we can employ finite-difference schemes to estimate $q^{(l-1)}(x_{i+\frac{1}{2}})$ using the Δx-grid function Q, given that it is assumed to be known.

Let us assume that the l-th derivative of Q at $x_{i+\frac{1}{2}}$ is approximated using a stencil $\mathcal{S}^{(l)}_{i+\frac{1}{2}}$ and weights $\beta^{(l)}_{k,i}$, where $k \in \mathcal{S}^{(l)}_{i+\frac{1}{2}}$. When d is odd, we can seek a parameter $\alpha_i \in \mathbb{R}^{d+1}$ that ensures mass conservation and approximates q and its derivatives at the edges by solving the following system:

$$\begin{cases} \int_0^1 \Phi(\xi;\alpha_i)\, d\xi & = Q_i, \\ \Phi^{(l)}(0;\alpha_i) & = (\Delta x)^l \sum_{k\in\mathcal{S}^{(l)}_{i-\frac{1}{2}}} \beta^{(l)}_{k,i} Q_k, \quad \text{for } l = 0,\dots,d-1. \end{cases} \tag{2.39}$$

If d is even, similarly we look for a parameter $\alpha_i \in \mathbb{R}^{d+1}$ that solves:

$$\begin{cases} \int_0^1 \Phi(\xi;\alpha_i)\, d\xi & = Q_i, \\ \Phi^{(l)}(0;\alpha_i) & = (\Delta x)^l \sum_{k\in\mathcal{S}^{(l)}_{i-\frac{1}{2}}} \beta^{(l)}_{k,i} Q_k, \quad \text{for } l = 0,\dots,\frac{d}{2}-1, \\ \Phi^{(l)}(1;\alpha_i) & = (\Delta x)^l \sum_{k\in\mathcal{S}^{(l)}_{i+\frac{1}{2}}} \beta^{(l)}_{k,i} Q_k, \quad \text{for } l = 0,\dots,\frac{d}{2}-1. \end{cases} \tag{2.40}$$

The reconstruction problem becomes linear when $\Phi(\xi;\mu)$ can be expressed as:

$$\Phi(\xi;\mu) = \sum_{k=0}^{d} \mu_k \Phi_k(\xi),$$

where Φ_k are functions defined on $[0,1]$. In this case, Equation (2.39) and Equation (2.40) form $(d+1)\times(d+1)$ linear systems. It is common to assume that the Φ_k's are linearly independent. Therefore, we have described a method that allows us to reconstruct a function from its average values, preserving its mass in each control volume, and approximating q at the edges. This method works for functions Φ_k as long as they are sufficiently differentiable. For example, choosing $d = 0$ and $\Phi_0(\xi) = 1$ gives us piecewise constant functions, as used in Godunov (1959). If we choose $d = 1$, $\Phi_0(\xi) = 1$, and $\Phi_1(\xi) = \xi$, we obtain a piecewise linear reconstruction, similar to Van Leer (1977). For polynomial reconstruction schemes, we refer to Engwirda and Kelley (2016) and the references therein.

Hereafter, we are going the focus on the piecewise parabolic method from Colella and Woodward (1984) that uses $d = 2$, $\Phi_0(\xi) = 1$, $\Phi_1(\xi) = \xi$, $\Phi_1(\xi) = (1-\xi)\xi$. In order to follow the notation from Colella and Woodward (1984), we write $\alpha_{0i} = q_{L,i}$, $\alpha_{1i} = \Delta q_i$ and $\alpha_{2i} = q_{6,i}$. Therefore, each q_i may be expressed as:

$$q_i(x;Q) = q_{L,i} + \Delta q_i z_i(x) + q_{6,i} z_i(x)(1 - z_i(x)), \quad \text{where } z_i(x) = \frac{x - x_{i-\frac{1}{2}}}{\Delta x}, \quad x \in X_i, \tag{2.41}$$

where the values $q_{L,i}$, Δq_i and $q_{6,i}$ will be specified latter. Note that each z_i is just a normalization function that maps X_i onto $[0,1]$. It is easy to see that $\lim_{x\to x^+_{i-\frac{1}{2}}} q_i(x;Q) = q_{L,i}$. If we define $q_{R,i} = \lim_{x\to x^-_{i+\frac{1}{2}}} q_i(x;Q)$, then we have:

$$\Delta q_i = q_{R,i} - q_{L,i}. \tag{2.42}$$

The average value of q_i is given by:

$$\frac{1}{\Delta x}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} q_i(x;Q)\,dx = \frac{(q_{L,i} + q_{R,i})}{2} + \frac{q_{6,i}}{6}. \tag{2.43}$$

Under the hypothesis of mass conservation, we have:

$$q_{6,i} = 6\left(Q_i - \frac{(q_{L,i} + q_{R,i})}{2}\right). \tag{2.44}$$

Therefore, we have found the parameters Δq_i and $q_{6,i}$ as functions of the parameters $q_{L,i}$ and $q_{R,i}$, such that the parabola q_i from (2.37) guarantees mass conservation. To completely determine the parabola q_i, we need to set the values $q_{L,i}$ and $q_{R,i}$, which, as we have seen, represent the limits of q_i when x tends to the left and right boundaries of X_i, respectively. Hence, it is natural to seek for $q_{L,i}$ as an approximation of $q(x_{i-\frac{1}{2}})$ and $q_{R,i}$ as an approximation of $q(x_{i+\frac{1}{2}})$. As we mentioned before in after introducing Equation (2.38), this is achieved using finite-differences.

2.4.1 hord0

This Subsection is dedicated to present the unlimited approximation of $q(x_{i-\frac{1}{2}})$ presented in Colella and Woodward (1984). An explicit expression for the approximation of $q(x_{i-\frac{1}{2}})$, denoted by $q_{i+\frac{1}{2}}$, is given by (Colella & Woodward, 1984):

$$q_{i+\frac{1}{2}} = \frac{1}{2}\left(Q_{i+1} + Q_i\right) - \frac{1}{6}\left(\delta Q_{i+1} - \delta Q_i\right), \tag{2.45}$$

where δQ_i is the average slope in the i-th control-volume:

$$\delta Q_i = \frac{1}{2}\left(Q_{i+1} - Q_{i-1}\right). \tag{2.46}$$

We notice that Formula (2.46) may be rewritten more explicitly as:

$$q_{i+\frac{1}{2}} = \frac{7}{12}\left(Q_{i+1} + Q_i\right) - \frac{1}{12}\left(Q_{i+2} + Q_{i-1}\right). \tag{2.47}$$

The Formula (2.47) is fourth-order accurate if q is at least C^4 (Colella & Woodward, 1984). Indeed, we prove this later in Proposition A.1. The expression for the values of $q_{R,i}$ and $q_{L,i}$ are given by:

$$q_{R,i} = q_{i+\frac{1}{2}} \tag{2.48}$$

$$q_{L,i} = q_{i-\frac{1}{2}}. \tag{2.49}$$

During this work, we refer to this PPM scheme as **hord0**. This name is justified because in FV3, the 1D advection solver input is named “hord”.

2.4.2 hord8

This Subsection is dedicated to presenting a possible way of ensuring the creation of new extrema values in the PPM reconstruction. We are going to present an alternative scheme from Lin (2004), which was an attempt to reduce the diffusion of the original scheme Colella and Woodward (1984) and is currently employed in the FV3 dynamical core (L. Harris et al., 2021).

Similarly to Colella and Woodward (1984), Lin (2004) reduces numerical oscillations in the parabolas by defining the average slope as

$$\delta_m Q_i = \max(|\delta Q_i|, 2\delta Q_{\min,i}, 2\delta Q_{\max,i}) \cdot \operatorname{sgn}(\delta Q_i) \tag{2.50}$$

where $\delta Q_i = \frac{Q_{i+1}-Q_{i-1}}{2}$, $\delta Q_{\min,i} = Q_i - \min(Q_{i+1}, Q_i, Q_{i-1})$ $\delta Q_{\max,i} = \max(Q_{i+1}, Q_i, Q_{i-1}) - Q_i$. We then initially compute an analogous version of Equation (2.45) as:

$$q_{i+\frac{1}{2}} = \frac{1}{2}\left(Q_{i+1} + Q_i\right) - \frac{1}{6}\left(\delta_m Q_{i+1} - \delta_m Q_i\right). \tag{2.51}$$

The values $q_{R,i}$ and $q_{L,i}$ are then computed using Equations (2.48) and (2.49), respectively. The monotonicity is achieved by the following scheme:

$$q_{L,i} \leftarrow Q_i - \max(|\delta_m Q_i|, |q_{L,i} - Q_i|) \cdot \operatorname{sgn}(\delta_m Q_i), \tag{2.52}$$

$$q_{R,i} \leftarrow Q_i - \max(|\delta_m Q_i|, |q_{R,i} - Q_i|) \cdot \operatorname{sgn}(\delta_m Q_i). \tag{2.53}$$

This scheme may be further improved to reduce the diffusion even more, as described by Lin (2004), but we are not going to assess this approach here. This scheme is referred to as **hord8** because, in FV3, the paramenter “hord” is set equal to 8 to use this scheme. At last, we point out that many other PPM reconstruction schemes are available in the literature and in FV3 (L. Harris et al., 2021; Lin et al., 2017), but for simplicity, we are just going to consider the schemes hord0 and hord8.

2.5 Flux

Let us consider the framework outlined in Problem 2.4. Assuming that $Q^n \in \mathbb{P}_\nu^N$ is known, our objective is to compute the values Q^{n+1}. To accomplish this, we utilize a scheme similar to the one presented in Problem 2.4, taking into account the presence of a reconstruction function $\tilde{q}(x;Q^n)$ as discussed in Section 2.4, and an initial departure point estimation $\tilde{x}^n_{i+\frac{1}{2}} = x_{i+\frac{1}{2}} - \tilde{u}^n_{i+\frac{1}{2}}\Delta t$ for a time-averaged wind $\tilde{u}^n_{i+\frac{1}{2}}$ as explained in Section 2.3. The numerical flux function $F^n_{i+\frac{1}{2}}$ is then suggested in Problem 2.4:

$$F^n_{i+\frac{1}{2}}[Q^n, \tilde{u}^n] = \frac{1}{\Delta t}\int_{x_{i+\frac{1}{2}}-\tilde{u}^n_{i+\frac{1}{2}}\Delta t}^{x_{i+\frac{1}{2}}} \tilde{q}(x;Q^n)\,dx. \tag{2.54}$$

Notice that if we define the averaged CFL number,

$$\tilde{c}^n_{i+\frac{1}{2}} = \tilde{u}^n_{i+\frac{1}{2}}\frac{\Delta t}{\Delta x},$$

where $\tilde{c}^n_{i+\frac{1}{2}} = k+\alpha^n_{i+\frac{1}{2}}$, $k = \lfloor \tilde{c}^n_{i+\frac{1}{2}} \rfloor$, $\alpha^n_{i+\frac{1}{2}} \in [0,1[$, we can express the numerical flux as (Y. Chen et al., 2017; Lin & Rood, 1996):

$$F^n_{i+\frac{1}{2}}[Q^n, \tilde{u}^n] = \frac{1}{\Delta t}\begin{cases} \Delta x\sum_{l=0}^{k-1} Q_{i-l} + \int_{x_{i-k+\frac{1}{2}}-\alpha^n_{i+\frac{1}{2}}\Delta x}^{x_{i-k+\frac{1}{2}}} \tilde{q}(x;Q^n)\,dx, & \text{if } \tilde{u}^n_{i+\frac{1}{2}} \geq 0, \\ \Delta x\sum_{l=0}^{k-1} Q_{i-l} - \int_{x_{i-k+\frac{1}{2}}}^{x_{i-k+\frac{1}{2}}-\alpha^n_{i+\frac{1}{2}}\Delta x} \tilde{q}(x;Q^n)\,dx, & \text{if } \tilde{u}^n_{i+\frac{1}{2}} < 0. \end{cases} \tag{2.55}$$

where we used that $\tilde{q}$ preserves the local mass.

We will provide explicit expressions for the integrals in Equation (2.55) when using the PPM method. For each control volume edge, denoted by $i = 0, \ldots, N$, and $y > 0$, we define the following averages of the Piecewise-Parabolic approximation, as defined in Equation (2.37) for Q^n (Colella & Woodward, 1984):

$$F_{L,i+\frac{1}{2}}[Q^n, y] = \frac{1}{y}\int_{x_{i+\frac{1}{2}}-y}^{x_{i+\frac{1}{2}}} \tilde{q}(x;Q^n)\,dx, \tag{2.56}$$

and

$$F_{R,i+\frac{1}{2}}[Q^n, y] = \frac{1}{y}\int_{x_{i+\frac{1}{2}}}^{x_{i+\frac{1}{2}}+y} \tilde{q}(x;Q^n)\,dx. \tag{2.57}$$

If $y \leq \Delta x$, then both of the above integral domains are constrained to a single control volume. Thus, it follows from a straightforward computation using Equation (2.41) that:

$$F_{L,i+\frac{1}{2}}[Q^n, y] = \frac{1}{y}\int_{x_{i+\frac{1}{2}}-y}^{x_{i+\frac{1}{2}}} q_i(x;Q^n)\,dx = q_{R,i} + \frac{(q_{6,i} - \Delta q_i)}{2\Delta x}y - \frac{q_{6,i}}{3\Delta x^2}y^2, \tag{2.58}$$

and

$$F_{R,i+\frac{1}{2}}[Q^n, y] = \frac{1}{y}\int_{x_{i+\frac{1}{2}}}^{x_{i+\frac{1}{2}}+y} q_{i+1}(x;Q^n)\,dx = q_{L,i+1} + \frac{(q_{6,i+1} + \Delta q_{i+1})}{2\Delta x}y - \frac{q_{6,i+1}}{3\Delta x^2}y^2. \tag{2.59}$$

The numerical flux function for PPM is then defined by:

$$\mathfrak{F}^{PPM}_{i+\frac{1}{2}}[Q^n, \tilde{u}^n] = \begin{cases} F_{L,i+\frac{1}{2}}[Q^n, \alpha^n_{i+\frac{1}{2}}\Delta x] & \text{if } \tilde{u}^n_{i+\frac{1}{2}} \geq 0, \\ F_{R,i+\frac{1}{2}}[Q^n, -\alpha^n_{i+\frac{1}{2}}\Delta x] & \text{if } \tilde{u}^n_{i+\frac{1}{2}} < 0, \end{cases} \tag{2.60}$$

and

$$F^n_{i+\frac{1}{2}}[Q^n, \tilde{u}^n] = \frac{1}{\Delta t}\left(\Delta x \sum_{l=0}^{k-1} Q_{i-l} + \Delta x \alpha^n_{i+\frac{1}{2}} \mathfrak{F}^{PPM}_{i+\frac{1}{2}}[Q^n, \tilde{u}^n]\right). \tag{2.61}$$

In particular, if the CFL number is less than one, then:

$$\mathfrak{F}^{PPM}_{i+\frac{1}{2}}[Q^n, \tilde{c}^n] = \begin{cases} q_{R,i} + \left(\frac{q_{6,i}-\Delta q_i}{2}\right)\tilde{c}^n_{i+\frac{1}{2}} - \frac{q_{6,i}}{3}(\tilde{c}^n_{i+\frac{1}{2}})^2, & \text{if } \tilde{c}^n_{i+\frac{1}{2}} \geq 0, \\ q_{L,i+1} + \left(\frac{q_{6,i+1}+\Delta q_{i+1}}{2}\right)\tilde{c}^n_{i+\frac{1}{2}} - \frac{q_{6,i+1}}{3}(\tilde{c}^n_{i+\frac{1}{2}})^2, & \text{if } \tilde{c}^n_{i+\frac{1}{2}} < 0, \end{cases} \tag{2.62}$$

and

$$F^n_{i+\frac{1}{2}}[Q^n, \tilde{c}^n] = \tilde{u}^n_{i+\frac{1}{2}} \mathfrak{F}^{PPM}_{i+\frac{1}{2}}[Q^n, \tilde{c}^n], \tag{2.63}$$

where we are expressing the flux in terms of the time-averaged CFL number $\tilde{c}^n$. Notice that this flux is upwind based, that is, it always computes the flux using the parabola in the upwind direction. Finally, for both **hord0** and **hord8** schemes, $F^n_{i+\frac{1}{2}}$ uses the stencil $\mathcal{S}_{i+\frac{1}{2}} = \{i-3, i-2, i-1, i, i+1, i+2, i+3\}$, and therefore we need $\nu = 3$ layers of ghost cells.

In FV3, the 1D flux is computed based on the perturbation values (L. Harris et al., 2021) given by:

$$b_{L,i} = q_{L,i} - Q^n_i, \tag{2.64}$$
$$b_{R,i} = q_{R,i} - Q^n_i. \tag{2.65}$$

Then, Equation (2.62) becomes:

$$\mathfrak{F}^{PPM}_{i+\frac{1}{2}}[Q^n, \tilde{c}^n] = \begin{cases} Q^n_i + (1-\tilde{c}^n_{i+\frac{1}{2}})\big(b_{R,i} - \tilde{c}^n_{i+\frac{1}{2}}(b_{L,i} + b_{R,i})\big), & \text{if } \tilde{c}^n_{i+\frac{1}{2}} \geq 0, \\ Q^n_{i+1} + (1+\tilde{c}^n_{i+\frac{1}{2}})\big(b_{L,i+1} + \tilde{c}^n_{i+\frac{1}{2}}(b_{L,i+1} + b_{R,i+1})\big), & \text{if } \tilde{c}^n_{i+\frac{1}{2}} < 0, \end{cases} \tag{2.66}$$

which is the formula implemented in FV3. Finally, the average value update is implemented in FV3 as

$$Q^{n+1}_i = Q^n_i - \left(\tilde{c}^n_{i+\frac{1}{2}} \mathfrak{F}^{PPM}_{i+\frac{1}{2}}[Q^n, \tilde{c}^n] - \tilde{c}^n_{i-\frac{1}{2}} \mathfrak{F}^{PPM}_{i-\frac{1}{2}}[Q^n, \tilde{c}^n]\right), \tag{2.67}$$

for $i = 1, \cdots, N$. Thefore, at each time-step, we need to:

1. Compute $\tilde{c}^n_{i+\frac{1}{2}}$ (for $i = 0, \cdots, N$) using the schemes DP1 or DP2;
2. Compute $q_{L,i}$ and $q_{R,i}$ (for $i = 1, \cdots, N$) using hord0 or hord8;
3. Evalute the pertubation values (for $i = 1, \cdots, N$) using Equations (2.64) and (2.65);
4. Evaluate the fluxes $\mathfrak{F}^{PPM}_{i+\frac{1}{2}}$ (for $i = 0, \cdots, N$) using Equation (2.66);
5. Update the Q^{n+1} using Equation (2.67).

2.6 Numerical experiments

This Section is dedicated to presenting the numerical results of the PPM and its variations discussed here. We will consider the reconstruction schemes **hord0** (Subsection

2.4.1) and **hord8** (Subsection 2.4.2), as well as the departure point schemes **DP1** (Subsection 2.3.1) and **DP2** (Subsection 2.3.2). The code used in this Section can be found in Appendix B.

For all the simulations presented here, we will consider the spatial domain $[-\frac{L}{2}, \frac{L}{2}]$, and the time interval $[0, T]$, where $L = \frac{\pi}{2}R$, $R = 6.371 \times 10^6$ meters is the Earth's radius and $T = 1036800$ seconds, equivalent to 12 days. The spatial domain spans approximately 10^4 kilometers, which corresponds to approximately the length of a cubed-sphere panel, as shall be seen in Chapter 4. The relative change at time step n in the mass is computed as:

$$\frac{|M^n - M^0|}{|M^0|},$$

where M^n is given by Equation (2.23). For all the simulations, the mass is preserved with machine precision. Furthermore, we compute the initial average values $Q_i(0)$ using the initial values of q_i^0 at the control volume centroids for all simulations, which is second-order accurate by Proposition 2.2. In the error calculation, only when q_0 is given by Equation (2.70), we replace $Q_i(t^n)$ by its centroid value $q_i(t^n)$, which again gives a second-order approximation by Proposition 2.2.

2.6.1 Square wave with constant wind advection

As a first numerical experiment, we consider a discontinuous IC given by:

$$q_0(x) = \begin{cases} 1 & \text{if } x \in [-0.1L, 0.1L], \\ 0.1 & \text{otherwise.} \end{cases} \tag{2.68}$$

for the linear advection equation with constant velocity, which we adopt as $u = \frac{L}{T}$.

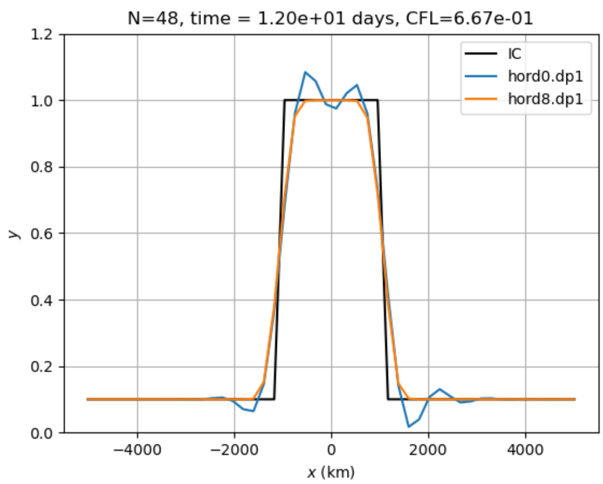


Figure 2.4: *Linear advection experiment using the IC given by Equation (2.68) (black curve) with constant velocity. These figures show the advected profile after 12 days (one time period). Reconstruction schemes employed: hord0 (blue curve) and hord8 (orange curve).*

It is easy to check that the exact solution of Problem 2.1 is given by $q_0(x - ut)$ and that the solution returns to its initial position after 12 days. We will employ a time step of 14400 seconds and set $N = 48$, resulting in a CFL number approximately equal to 0.67. The departure schemes **DP1** and **DP2** compute the departure point exactly in this case, so we will only use the **DP1** scheme.

In Figure 2.4, we present the obtained results. It is evident that the monotonic scheme hord8 exhibit a significant advantage. This scheme effectively prevent the strong oscillations observed in the hord0 scheme, as well as the generation of new extrema, which aligns with our expectations.

2.6.2 Flow deformation with divergent wind

As a second experiment, we shall investigate the how the PPM schemes behave when the velocity is variable. This cases is useful to assess the departure point schemes, which shall not be exact as in the previous test. We are going to consider the velocity

$$u(x,t) = u_0 \cos\left(\frac{\pi t}{T}\right)\cos^2\left(\pi\left(\frac{x}{L}-\frac{t}{T}\right)\right) + u_1. \tag{2.69}$$

We adopt the parameters $T = 12$ days and $u_0 = u_1 = \frac{L}{T}$. Following the approach in Trefethen (2000), we initialize the periodic Gaussian profile defined as:

$$q(x) = 0.1 + 0.9\exp\left(-10\sin^2\left(\frac{\pi x}{L}\right)\right), \quad x \in \left[-\frac{L}{2}, \frac{L}{2}\right]. \tag{2.70}$$

The velocity function given by Equation (2.69) is based on the deformational flow test case in Nair and Lauritzen (2010), where we add a constant wind u_1 to prevent error cancellations. As the velocity is variable, we utilize the departure point schemes DP1 and DP2. In this case, the solution exhibits a period of 12 days, meaning that the profile deforms and returns to its initial shape and position after 12 days, allowing us to compute the error. Indeed, in Figure 2.5, we show how the solution behaves using a high-resolution ($N = 768$), the hord8 scheme and the DP1 departure point scheme.

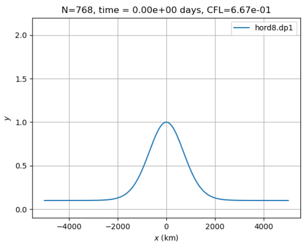
To investigate the error convergence, we employ $(\Delta x^{(k)}, \Delta t^{(k)}, \lambda)$-discretizations with $\Delta x^{(k)} = \frac{L}{N^{(k)}}$, $N^{(k)} = 48 \times 2^k$, $\Delta t^{(k)} = \frac{7200}{2^k}$, for $k = 0, \ldots, 4$. To measure the accuracy, we consider the relative error in the maximum norm as follows:

$$E_k = \frac{\|Q^{N_T} - Q^0\|_{\infty,\Delta x}}{\|Q^0\|_{\infty,\Delta x}}.$$

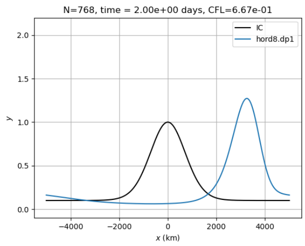
The convergence rate is defined by

$$CR_k = \frac{\ln\left(\frac{E_k}{E_{k-1}}\right)}{\ln 2}, \quad \text{for} \quad k = 1, \ldots 4.$$

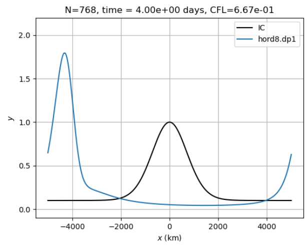
The difference between the DP1 and DP2 schemes becomes clear when observing the relative error in Figure 2.6. In the L_∞ norm (Figure 2.6a), for hord0, the DP1 scheme results in a first-order error in the departure point, which dominates the total error. This



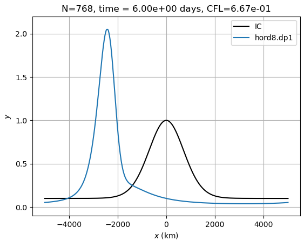
(a) $t = 0$.



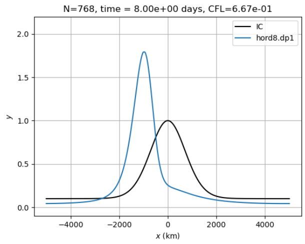
(b) $t = 2$ *days.*



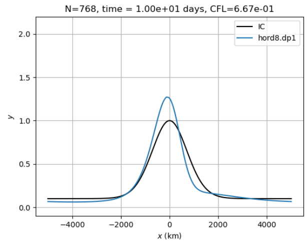
(c) $t = 4$ *days.*



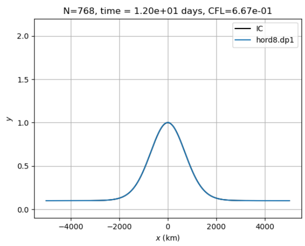
(d) $t = 6$ *days.*



(e) $t = 8$ *days.*

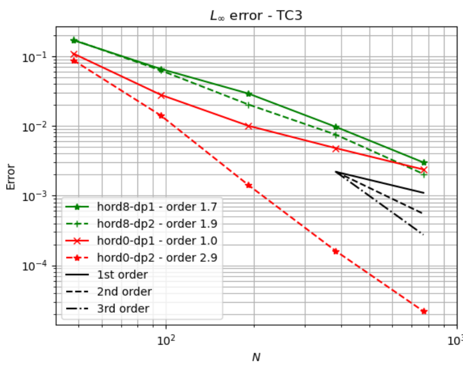


(f) $t = 10$ *days.*

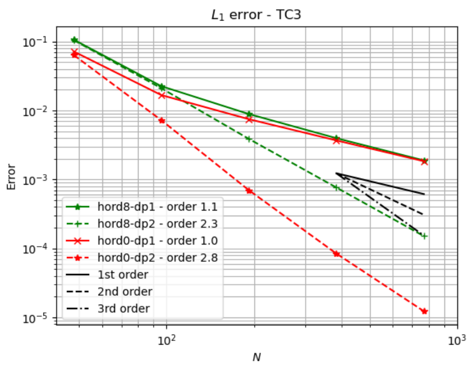


(g) $t = 12$ *days.*

Figure 2.5: *Linear advection experiment using the velocity from Equation (2.70), a CFL number equal to* 0.67*,* $N = 768$ *cells, and the IC is given by Equation (2.70) (2.5a). These figures show the advected profile at 2 (2.5b), 4 (2.5c), 6 (2.5d), 8 (2.5e), 10 (2.5f), and 12 (2.5g) days. We are using the hord8 scheme with the DP1 departure point scheme.*



(a) L_∞ *error.*



(b) L_1 *error.*

Figure 2.6: *Relative error for hord0 (red lines) and hord8 (green lines) schemes in* L_∞ *(Figure 2.6a) and* L_1 *norms (Figure 2.6b). Results using DP1 scheme uses solid lines and DP2 results uses dashed lines. The IC given by Equation (2.70) and the variable velocity given by Equation (2.69).*

observation is in agreement with the discussion in Section 2.3. On the other hand, when employing the DP2 scheme, we can achieve third-order accuracy for hord0. For hord8, the DP2 slightly reduces the L_∞ error.

However, in the L_1 norm, as shown in Figure 2.6b, for both hord0 and hord8, we observe that DP1 results in a 1st order accuracy, while DP2 results in schemes with an order greater than 2. This experiment illustrates the impact of departure point calculation errors on the overall error and the benefit of using DP2.

2.7 Concluding remarks

In this Chapter, we provided a general overview of 1D FV-SL schemes for the advection equation. We discussed the three essential tasks involved in these schemes. The first task is the reconstruction of a function from its average values. We employed the PPM method introduced by Colella and Woodward (1984) (hord0) and its monotonic variant such as the one from Lin (2004) (hord8). The second task involves computing the departure point of the control volume edges. For this purpose, we utilized the first-order departure point calculation using a time-centered wind in an approach known as DP1. Additionally, we explored a second-order approach by employing a two-stages Runge-Kutta scheme to integrate the departure point ODE. Lastly, the third task entails computing the flux, which involves integrating the reconstructed function over a domain determined by the departure point.

The difference between the departure point schemes became apparent when we performed a test with variable velocity. The simulation using the DP1 scheme with hord0 resulted in a final first-order error, despite the scheme having third-order accuracy in space. However, the DP2 scheme with hord0 preserved third-order accuracy despite being only second-order accurate. We expect that, in general, combining PPM with the DP2 scheme should result in at least second-order accuracy. The DP2 scheme also showed to lead to a more accurate result when combined with hord8, especially in the L_1 norm. Clearly, the DP2 scheme is more computationally expensive since it requires linear interpolation of the velocity field, but this additional cost is minimal.

Chapter 3

Two-dimensional finite-volume methods

In Chapter 2, we addressed the problem of solving the one-dimensional linear advection equation using the finite-volume method based on PPM. In this Chapter, our focus shifts to solving the two-dimensional linear advection equation using the finite-volume method. This step is crucial in our work since, as we will explore in Chapter 5, solving the linear advection equation on the cubed-sphere relies on solving two-dimensional linear advection equations at each cube face, with interpolation between adjacent panels, which are described in Chapter 4.

A natural approach to develop a finite-volume method for the two-dimensional linear advection equation would involve extending PPM to two dimensions. Indeed, Rančić (1992) proposed a piecewise bi-parabolic extension of PPM using a semi-Lagrangian temporal discretization. Further, this type of method can be extended to the cubed-sphere (Lauritzen et al., 2010). However, this method suffers from a significant drawback—its computationally expensive nature. As a popular alternative, dimension-splitting methods are often used, which replace the two-dimensional problem with a sequence of one-dimensional problems. For example, we can solve the two-dimensional linear advection equation by solving a series of one-dimensional linear advection equations using the PPM from Chapter 2. Moreover, in principle, we can employ any numerical method that solves the one-dimensional linear advection equation.

A comparison between two-dimensional and dimension-splitting semi-Lagrangian schemes on a plane was investigated by Y. Chen et al. (2017), utilizing the PPM as the one-dimensional solver and distorted two-dimensional grids. Their main conclusion was that dimension-splitting schemes are more sensitive to grid distortions, but they are computationally cheaper and more accurate than two-dimensional methods, particularly when dealing with large CFL numbers.

The primary objective of this Chapter is to provide a comprehensive explanation of the dimension splitting method proposed by Lin and Rood (1996). This method is currently utilized in the FV3 dynamical core and is applied to the two-dimensional linear advection equation using the one-dimensional finite-volume schemes described in Chapter 2. To begin, similar to Chapter 2, we start this Chapter with a review of the integral form of the two-dimensional advection equation in Section 3.1. Following this, in Section 3.2, we establish the framework for general two-dimensional finite-volume schemes. Subsequently, the dimension splitting method is presented in Section 3.3, where we delve into its intricacies. Finally, we showcase numerical experiments in Section 3.4 to illustrate the practical application of the dimension splitting approach.

3.1 Two-dimensional advection equation in integral form

3.1.1 Notation

This Section is dedicated to extending the notation of Section 2.1.1. Based on definitions 2.1 and 2.3, we introduce the concepts of a $(\Delta x, \Delta y)$-grid and $(\Delta x, \Delta y, \Delta t, \lambda)$ discretization. Throughout this Chapter, we will use the notation $\Omega = [a,b] \times [c,d]$ and ν to represent a non-negative integer indicating the number of ghost cell layers in each boundary. We also use the notations $\mathbb{R}_\nu^{N\times M} := \mathbb{R}^{(N+2\nu)\times(M+2\nu)}$ and $\mathbb{R}_\nu^{(N+1)\times M} := \mathbb{R}^{(N+1+2\nu)\times(M+2\nu)}$, $\mathbb{R}_\nu^{N\times(M+1)} := \mathbb{R}^{(N+2\nu)\times(M+1+2\nu)}$.

Definition 3.1 ($(\Delta x, \Delta y)$-grid)**.** *Given Ω and positive real numbers Δx and Δy such that $\Delta x = (b-a)/N$, $\Delta y = (d-c)/M$, for positive integers N and M, we say that $\Omega_{\Delta x,\Delta y} = (\Omega_{ij})_{i=-\nu+1,\dots,N+\nu}^{j=-\nu+1,\dots,M+\nu}$ is a $(\Delta x, \Delta y)$-grid for Ω if*

$$\Omega_{ij} = [x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}] \times [y_{j-\frac{1}{2}}, y_{j+\frac{1}{2}}] = [a+(i-1)\Delta x, a+i\Delta x] \times [c+(j-1)\Delta x, c+j\Delta x],$$

$\Delta x = x_{i+\frac{1}{2}} - x_{i-\frac{1}{2}}$, $\Delta y = y_{j+\frac{1}{2}} - y_{j-\frac{1}{2}}$. Each Ω_{ij} is called control volume or cell. The cell centroids (x_i, y_j) are defined by

$$x_i = \frac{1}{2}(x_{i+\frac{1}{2}} + x_{i-\frac{1}{2}}), \quad y_j = \frac{1}{2}(y_{j+\frac{1}{2}} + y_{j-\frac{1}{2}}).$$

Remark 3.1. *If $1 \le i \le N$,$1 \le j \le M$, we refer to (i,j) as an interior index; otherwise, (i,j) is considered a ghost cell index and we say the Ω_{ij} is a ghost cell.*

Definition 3.2 ($(\Delta x, \Delta y, \Delta t, \lambda)$-discretization)**.** *Given $\Omega \times [0,T]$, and positive real numbers Δx Δy and Δt, we say that $(\Omega_{\Delta x,\Delta y}, T_{\Delta t})$ is a $(\Delta x, \Delta y, \Delta t, \lambda)-$discretization of $\Omega \times [0,T]$ if $\Omega_{\Delta x,\Delta y}$ is a $(\Delta x, \Delta y)$ grid for Ω and $T_{\Delta t}$ is a Δt-temporal grid for $[0,T]$, $\frac{\Delta t}{\Delta x} = \lambda$ and $\frac{\Delta t}{\Delta y} = \lambda$.*

Remark 3.2. *Whenever we mention a $(\Delta x, \Delta y)-$grid, or a $(\Delta x, \Delta y, \Delta t, \lambda)$-discretization, then Ω_{ij}, N and M are implicitly defined.*

Next, we introduce the definitions of grid functions at cell centroids and C-grid functions.

Definition 3.3 ($(\Delta x, \Delta y)$-grid function)**.** *For a $(\Delta x, \Delta y)$-grid, we say that $Q = (Q_{ij})_{i=-\nu+1,\dots,N+\nu}^{j=-\nu+1,\dots,M+\nu} \in \mathbb{R}_\nu^{N\times M}$ is a $(\Delta x, \Delta y)$-grid function.*

Definition 3.4 ($(\Delta x, \Delta y)$-C grid wind)**.** *For a $(\Delta x, \Delta y)$-grid, we say that (u, v) is a $(\Delta x, \Delta y)$-C grid wind if $u = (u_{i+\frac{1}{2},j})_{i=-\nu,\dots,N+\nu}^{j=-\nu+1,\dots,M+\nu} \in \mathbb{R}_\nu^{(N+1)\times M}$, $v = (v_{i,j+\frac{1}{2}})_{i=-\nu+1,\dots,N+\nu}^{j=-\nu,\dots,M+\nu} \in \mathbb{R}_\nu^{N\times(M+1)}$.*

Considering a function $q : \Omega \times [0,T] \to \mathbb{R}$, a vector field $\boldsymbol{u} : \Omega \times [0,T] \to \mathbb{R}$, $\boldsymbol{u} = (u, v)$, a $(\Delta x, \Delta y, \Delta t, \lambda)$-discretization of $\Omega \times [0,T]$, we introduce the grid functions $q^n \in \mathbb{R}_v^{N\times M}$, $u^n \in \mathbb{R}_v^{(N+1)\times M}$, $v^n \in \mathbb{R}_v^{N\times(M+1)}$. Here, $q_{ij}^n = q(x_i, y_j, t^n)$, $u^n_{i+\frac{1}{2},j} = u(x_{i+\frac{1}{2}}, y_j, t^n)$, $v^n_{i,j+\frac{1}{2}} = u(x_i, y_{j+\frac{1}{2}}, t^n)$. These grid functions represent the discrete values of q and $\boldsymbol{u}$ at the cell centroids and edges, respectively, for each time level t^n (Figure 2.2). We shall also use the notations $q^n_{i+\frac{1}{2},j} = q(x_{i+\frac{1}{2}}, y_j, t^n)$ and $q^n_{i,j+\frac{1}{2}} = q(x_i, y_{j+\frac{1}{2}}, t^n)$.

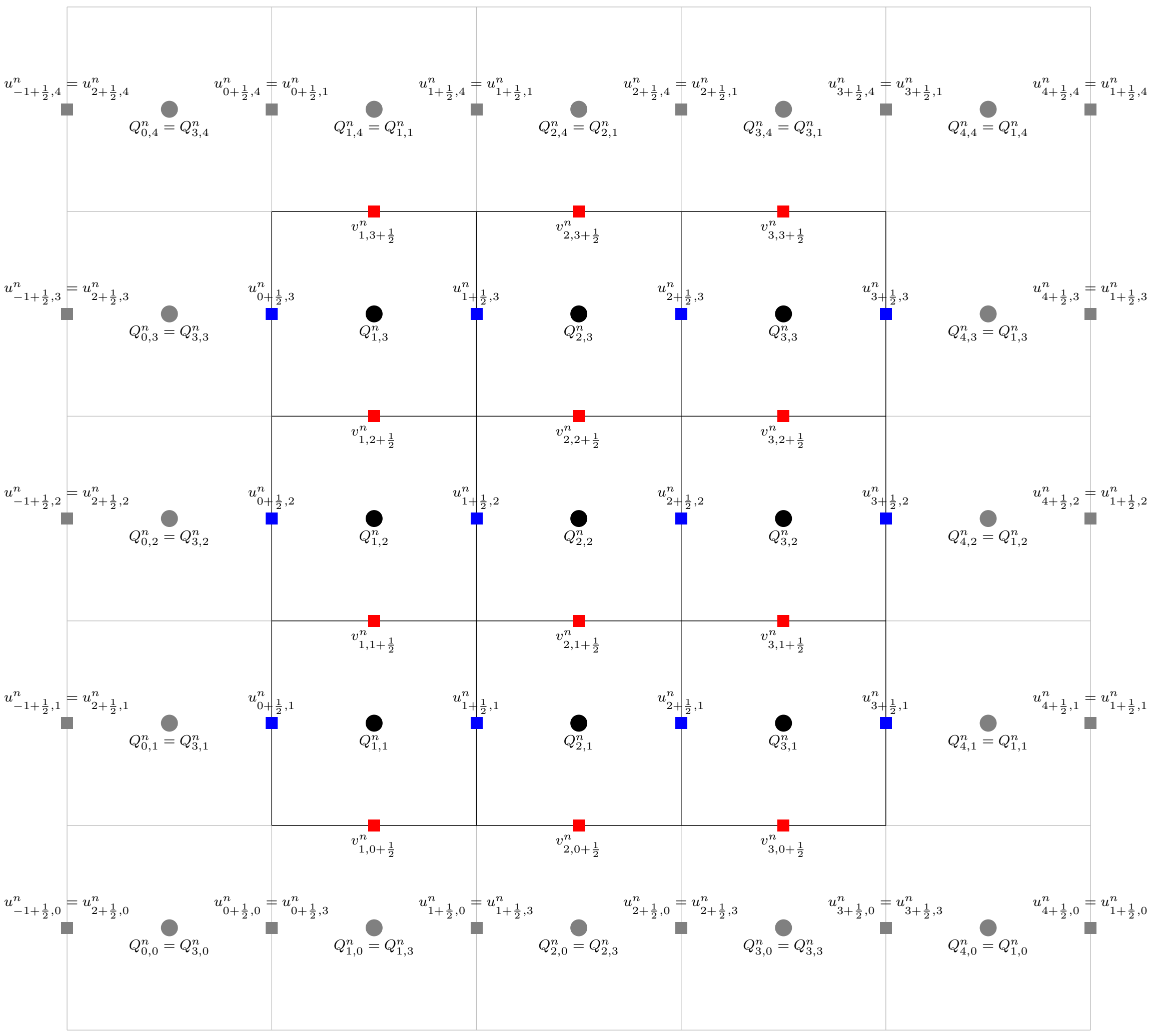


Figure 3.1: *Illustration of $(\Delta x, \Delta y)$-grid function Q (black circles) and a $(\Delta x \Delta y)$-C grid wind u (blue squares) and v (red squares) and its ghost cell values (in gray) assuming biperiodicity.*

We denote by $\nabla \cdot (q\boldsymbol{u})$ the divergence operator:

$$\nabla \cdot (q\boldsymbol{u})(x, y, t) = [\partial_x(uq) + \partial_y(vq)](x, y, t). \tag{3.1}$$

We recall that we say the $\boldsymbol{u}$ is **non-divergent** if $\nabla \cdot \boldsymbol{u} = 0$. We define the $(\Delta x, \Delta y)$-grid function δ^n as the exact divergence of $q\boldsymbol{u}$ at the cell centers, namely

$$\delta^n_{ij} = \nabla \cdot (\boldsymbol{u}q)(x_i, y_j, t^n). \tag{3.2}$$

In this Chapter, our focus also lies on periodic grid functions. We define a $(\Delta x, \Delta y)$-grid function Q as periodic if it satisfies the following conditions:

$$
\begin{aligned}
Q_{i,j} &= Q_{N+i,j}, &\quad i &= -\nu+1,\dots,0, &\quad j &= -\nu+1,\dots,M+\nu,\\
Q_{i,j} &= Q_{i-N,j}, & i &= N+1,\dots,N+\nu, & j &= -\nu+1,\dots,M+\nu,\\
Q_{i,j} &= Q_{i,M+j}, & j &= -\nu+1,\dots,0, & i &= -\nu+1,\dots,N+\nu,\\
Q_{i,j} &= Q_{i,j-M}, & j &= M+1,\dots,M+\nu, & i &= -\nu+1,\dots,N+\nu.
\end{aligned}
$$

We use the notation $\mathbb{P}_\nu^{N\times M}$ represent the spaces of periodic $(\Delta x, \Delta y)$-grid functions. Similarly, we define a $(\Delta x, \Delta y)$-grid wind (u, v) as periodic if it meets the following requirements:

$$
\begin{aligned}
u_{i-\frac{1}{2},j} &= u_{N+i+\frac{1}{2},j}, &\quad i &= -\nu,\dots,-1, &\quad j &= -\nu+1,\dots,M+\nu,\\
u_{i+\frac{1}{2},j} &= u_{i+\frac{1}{2}-N,j}, & i &= N+1,\dots,N+\nu, & j &= -\nu+1,\dots,M+\nu,\\
u_{i+\frac{1}{2},j} &= u_{i+\frac{1}{2},M+j}, & i &= -\nu,\dots,N+1+\nu, & j &= -\nu+1,\dots,0,\\
u_{i+\frac{1}{2},j} &= u_{i+\frac{1}{2},j-M}, & i &= -\nu,\dots,N+1+\nu, & j &= M+1,\dots,M+\nu,\\
v_{i,j-\frac{1}{2}} &= v_{i,M+j+\frac{1}{2}}, & j &= -\nu,\dots,-1, & i &= -\nu+1,\dots,N+\nu,\\
v_{i,j+\frac{1}{2}} &= v_{i,j+\frac{1}{2}-M}, & j &= M+1,\dots,M+\nu, & i &= -\nu+1,\dots,N+\nu,\\
v_{i,j+\frac{1}{2}} &= v_{N+i,j+\frac{1}{2}}, & j &= -\nu,\dots,M+1+\nu, & i &= -\nu+1,\dots,0,\\
v_{i,j+\frac{1}{2}} &= c_{i-N,j+\frac{1}{2}}, & j &= -\nu,\dots,N+1+\nu, & i &= N+1,\dots,N+\nu.
\end{aligned}
$$

In this case, we use the notation $u \in \mathbb{P}_\nu^{(N+1)\times M}$, $v \in \mathbb{P}_\nu^{N\times(M+1)}$.

For a grid function Q we also use the notations:

$$
\begin{aligned}
Q_{\times,j} &:= (Q_{-\nu+1,j},\dots,Q_{N+\nu,j}) \in \mathbb{R}_\nu^N,\\
Q_{i,\times} &:= (Q_{i,-\nu+1},\dots,Q_{i,M+\nu}) \in \mathbb{R}_\nu^M.
\end{aligned}
$$

Given $Q = (Q_{ij}) \in \mathbb{P}_{\nu,P}^{N\times M}$, we define the p-norm by

$$
\|Q\|_{p,\Delta x\times\Delta y} = \begin{cases} \left(\sum_{i=1}^N \sum_{j=1}^M |Q_{ij}|^p \right)^{\frac{1}{p}} & \text{if } 1 \le p < \infty,\\ \max_{i=1,\dots,N,j=1,\dots,M} |Q_{ij}| & \text{otherwise .} \end{cases} \tag{3.3}
$$

We also introduce the centered difference notation:

$$
\delta_x h(x_i, y, t) = h(x_{i+\frac{1}{2}}, y, t) - h(x_{i-\frac{1}{2}}, y, t), \tag{3.4}
$$

$$
\delta_y h(x, y_j, t) = h(x, y_{j+\frac{1}{2}}, t) - h(x, y_{j-\frac{1}{2}}, t), \tag{3.5}
$$

for any function $h : \Omega \times [0, T] \to \mathbb{R}$. Additionally, we introduce the average value of q in the control volume Ω_{ij} at time t, denoted as $Q_{ij}(t)$, defined by:

$$Q_{ij}(t) = \frac{1}{\Delta x \Delta y} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \int_{y_{j-\frac{1}{2}}}^{y_{j-\frac{1}{2}}} q(x, y, t)\, dx. \tag{3.6}$$

Moreover, we define the $(\Delta x, \Delta y)$-grid function of average values as $Q(t) = (Q_{ij}(t))_{i=-\nu+1,\ldots,N+\nu}^{j=-\nu+1,\ldots,M+\nu}$.

For the consideration of periodic boundary conditions, we can define spaces of periodic functions over the interval Ω as follows:

$$\mathcal{S}_P(\Omega) = \{q : \mathbb{R}^2 \times [0, +\infty[\to \mathbb{R} : q(x + b - a, y + d - c, t) = q(x, y, t), \quad \forall x, y \in \mathbb{R}, \quad t \geq 0\}.$$

Similarly, the space of k-times periodically differentiable functions $C_P^k(\Omega)$ can be defined as:

$$C_P^k(\Omega) = \mathcal{S}_P(\Omega) \cap C^k(\mathbb{R}^2 \times [0, \infty[),$$

where $C^k(\mathbb{R}^2 \times [0, +\infty[)$ denotes the space of functions that are k times continuously differentiable in both the spatial and temporal variables. In summary, $\mathcal{S}_P(\Omega)$ represents the space of periodic functions, and $C_P^k(\Omega)$ represents the space of k-times periodically differentiable functions over Ω subject to periodic boundary conditions.

3.1.2 The 2D advection equation

Let us consider a velocity field given by $\boldsymbol{u} = (u, v)$, where u is the velocity in x-direction and v is the velocity in x and y direction and $u, v \in C_P^1(\Omega)$. The two-dimensional advection equation in its differential form in a domain Ω associated to the velocity field or wind $\boldsymbol{u}$ and assuming biperiodic boundary conditions is given by:

$$\begin{cases} [\partial_t q + \partial_x(uq) + \partial_y(vq)](x, y, t) = 0, \quad \forall (x, y, t) \in \mathbb{R}^2 \times]0, +\infty[, \\ q(a, y, t) = q(b, y, t), \quad \forall y \in [c, d], \quad \forall t \geq 0, \\ q(x, c, t) = q(x, d, t), \quad \forall x \in [a, b], \quad \forall t \geq 0, \\ q_0(x) = q(x, y, 0), \quad \forall (x, y) \in \Omega. \end{cases} \tag{3.7}$$

A classical or strong solution to the two-dimensional advection equation is a $C_P^1(\Omega)$ function q satisfying Equation (3.7). As we did in Section 2.1, our goal is to deduce an integral form of Equation (3.7). Thus, let us consider $[x_1, x_2] \times [y_1, y_2] \subset \Omega$ and $[t_1, t_2] \subset [0, +\infty[$. Integrating Equation (3.7) over $[x_1, x_2] \times [y_1, y_2]$ yields:

$$\begin{aligned} \frac{d}{dt}\left(\int_{x_1}^{x_2} \int_{y_1}^{y_2} q(x, y, t)\, dx\, dy \right) = &- \int_{y_1}^{y_2} \Big((uq)(x_2, y, t) - (uq)(x_1, y, t) \Big)\, dy \\ &- \int_{x_1}^{x_2} \Big((vq)(x, y_2, t) - (vq)(x, y_1, t) \Big)\, dx. \end{aligned} \tag{3.8}$$

Integrating Equation (3.8) over the time interval $[t_1, t_2]$, we have:

$$\begin{aligned} \int_{x_1}^{x_2}\int_{y_1}^{y_2} q(x,y,t_{n+1})\,dx\,dy = &\int_{x_1}^{x_2}\int_{y_1}^{y_2} q(x,y,t_n)\,dx\,dy \\ &- \int_{t_1}^{t_2}\int_{y_1}^{y_2}\Big((uq)(x_2,y,t)-(uq)(x_1,y,t)\Big)\,dy\,dt \\ &- \int_{t_1}^{t_2}\int_{x_1}^{x_2}\Big((vq)(x,y_2,t)-(vq)(x,y_1,t)\Big)\,dx\,dt. \end{aligned} \tag{3.9}$$

Equation (3.9) is the integral form of Equation (3.7). We say that q is a weak solution to the advection equation (3.7) if q satisfies the integral form (3.9), $\forall [x_1, x_2] \times [y_1, y_2] \subset \Omega^o$ and $\forall [t_1, t_2] \subset [0, +\infty[$. We summarize the weak version of Equation (3.7) in Problem (3.1).

Problem 3.1. *Given an initial condition q_0 and a velocity function $\boldsymbol{u} = (u, v)$ we would like to find a weak solution q of the two-dimensional advection equation in its integral form:*

$$\begin{aligned} \int_{x_1}^{x_2}\int_{y_1}^{y_2} q(x,y,t)\,dx\,dy = &\int_{x_1}^{x_2}\int_{y_1}^{y_2} q(x,y,t)\,dx\,dy \\ &- \int_{t_1}^{t_2}\int_{y_1}^{y_2}\Big((uq)(x_2,y,t)-(uq)(x_1,y,t)\Big)\,dy\,dt \\ &- \int_{t_1}^{t_2}\int_{x_1}^{x_2}\Big((vq)(x,y_2,t)-(vq)(x,y_1,t)\Big)\,dx\,dt. \end{aligned}$$

$\forall [x_1, x_2] \times [y_1, y_2] \times [t_1, t_2] \subset \Omega \times [0, T]$, and $q(x, y, 0) = q_0(x, y)$, $\forall (x, y) \in \Omega$, $q(a, y, t) = q(b, y, t)$, $\forall y \in [c, d]$, $\forall t \geq 0$, $q(x, c, t) = q(x, d, t)$, $\forall x \in [a, b]$, $\forall t \geq 0$.

Similarly to Section 2.1, Equation (3.7) and Problem (3.1) are equivalent when $q, \boldsymbol{u} \in C_P^1(\Omega)$. For Problem 3.1, the total mass in Ω is defined by:

$$M_\Omega(t) = \int_\Omega q(x,y,t)\,dx\,dy, \quad \forall t \in [0,T], \tag{3.10}$$

and is conserved within time:

$$M_\Omega(t) = M_\Omega(0), \quad \forall t \in [0,T]. \tag{3.11}$$

Considering a $(\Delta x, \Delta y, \Delta t, \lambda)$ discretization of $D = \Omega \times [0, T]$ and substituting t_1, t_2, x_1, x_2, y_1 and y_2 by $t_n, t_{n+1}, x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}, y_{j-\frac{1}{2}}, y_{j+\frac{1}{2}}$, respectively, in Equation (3.9), we obtain:

$$\begin{aligned} Q_{ij}(t_{n+1}) = Q_{ij}(t_n) &- \frac{\Delta t}{\Delta x \Delta y}\delta_x\left(\frac{1}{\Delta t}\int_{t_1}^{t_2}\int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}} (uq)(x_i,y,t)\,dy\,dt\right) \\ &- \frac{\Delta t}{\Delta x \Delta y}\delta_y\left(\frac{1}{\Delta t}\int_{t_1}^{t_2}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} (vq)(x,y_j,t)\,dx\,dt\right), \end{aligned} \tag{3.12}$$

where we are using the centered finite-difference notation. Now we can define a discretized version of Problem 3.1 as Problem 3.2.

Problem 3.2. *Assume the framework of Problem 3.1 and consider a* $(\Delta x, \Delta y, \Delta t, \lambda)$-

discretization of $\Omega \times [0,T]$. Since we are in the framework of Problem 3.1, it follows that:

$$Q_{ij}(t_{n+1}) = Q_{ij}(t_n) - \lambda\delta_x\left(\frac{1}{\Delta t\Delta y}\int_{t^n}^{t^{n+1}}\int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}}(uq)(x_i,y,t)\,dy\,dt\right) - \lambda\delta_y\left(\frac{1}{\Delta t\Delta x}\int_{t^n}^{t^{n+1}}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}}(vq)(x,y_j,t)\,dx\,dt\right),$$

where $Q_{ij}(t) = \frac{1}{\Delta x\Delta y}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}}\int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}} q(x,y,t)\,dx\,dy$. Our problem now consists of finding the values $Q_{ij}(t_n)$, $\forall i = 1,\dots,N$, $\forall j = 1,\dots,M$, $\forall n = 0,\dots,N_T-1$, given the initial values $Q_{ij}(0)$, $\forall i = 1,\dots N$, $\forall j = 1,\dots,M$. In other words, we aim to find the average values of q in each control volume Ω_{ij} at the specified time instances.

It is important to note that no approximations have been made in Problems (3.1) and (3.2).

3.2 The finite-volume approach

Finally, we define the 2D-FV scheme problem as follows in Problem 3.3.

Problem 3.3 (2D-FV scheme)**.** *Assume the framework defined in Problem 3.2. The finite-volume approach of Problem 3.1 consists of a finding a scheme of the form:*

$$Q_{ij}^{n+1} = Q_{ij}^n - \lambda\delta_i F_{ij}^n - \lambda\delta_j G_{ij}^n, \qquad (3.13)$$
$$\forall i = 1,\dots,N, \quad \forall j = 1,\dots,M, \quad \forall n = 0,\dots,N_T-1,$$

where $\delta_i F_{ij}^n = F_{i+\frac{1}{2},j}^n - F_{i-\frac{1}{2},j}^n$, $\delta_j G_{ij}^n = G_{i,j+\frac{1}{2}}^n - G_{i,j-\frac{1}{2}}^n$ and $Q^n \in \mathbb{P}_v^{N\times M}$ is intended to be an approximation of $Q(t_n) \in \mathbb{P}_v^{N\times M}$ in some sense. We define $Q_{ij}^0 = Q_{ij}(0)$ or $Q_{ij}^0 = q_{ij}^0$.

The term $F_{i+\frac{1}{2},j}^n$ is known as numerical flux in the x direction and it approximates $\frac{1}{\Delta t\Delta y}\int_{t_n}^{t_{n+1}}\int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}}(uq)(x_{i+\frac{1}{2}},y,t)\,dy\,dt$, $\forall i = 0,1,\dots,N$, and $G_{i,j+\frac{1}{2}}^n$ is known as numerical flux in the y direction and it approximates $\frac{1}{\Delta t\Delta x}\int_{t_n}^{t_{n+1}}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}}(vq)(x,y_{j+\frac{1}{2}},t)\,dx\,dt$, $\forall j = 0,1,\dots,M$, or, in other words, they estimate the time-averaged fluxes at the control volume Ω_{ij} boundaries.

Remark 3.3. *For Problem 3.3, we define the CFL number in the x and y direction by $\max\{|u_{i+\frac{1}{2},j}^n|\}\frac{\Delta t}{\Delta x}$ and $\max\{|v_{i,j+\frac{1}{2}}^n|\}\frac{\Delta t}{\Delta y}$, respectively. The CFL number is maximum between these numbers and we say that the CFL condition is satisfied if the CFL number is less than one.*

For a 2D-FV the discrete total mass at the time-step n is given by

$$M^n = \Delta x\Delta y\sum_{i=1}^{N}\sum_{j=1}^{M} Q_{ij}^n.$$

Therefore, the discrete total mass is constant for a 2D-FV scheme, which follows from a straightforward computation:

$$M^{n+1} = \Delta x \sum_{i=1}^{N}\sum_{j=1}^{M} Q_{ij}^{n+1} = M^n - \Delta t \sum_{i=1}^{N}\sum_{j=1}^{M}(F^n_{i+\frac{1}{2},j} - F^n_{i-\frac{1}{2},j}) - \Delta t \sum_{i=1}^{N}\sum_{j=1}^{M}(G^n_{i,j+\frac{1}{2}} - G^n_{i,j-\frac{1}{2}})$$
$$= M^n - \Delta t \sum_{j=1}^{M}(F^n_{N+\frac{1}{2},j} - F^n_{\frac{1}{2},j}) - \Delta t \sum_{i=1}^{N}(G^n_{i,M+\frac{1}{2}} - G^n_{i,\frac{1}{2}}) = M^n,$$

where we are using that $F^n_{N+\frac{1}{2},j} = F^n_{\frac{1}{2},j}$, $G^n_{i,M+\frac{1}{2}} = G^n_{i,\frac{1}{2}}$ since we are assuming bi-periodic boundary conditions.

As we mentioned in Problem 3.3, the initial condition may be assumed as q^0_{ij} or $Q_{ij}(0)$. For two-dimensional simulations, we are going to assume q^0_{ij} as initial data to avoid the computation of integrals. Furthermore, the errors will be calculated using the values q^n_{ij} instead of $Q_{ij}(t_n)$. Similarly to Proposition 2.2, we have that the centroid value approximates the average value with second order, as Proposition 3.1 shows.

Proposition 3.1. *If $q \in C^2$, then $|Q_{ij}(t^n) - q^n_{ij}| = C_1\Delta x^2 + C_2\Delta x\Delta y + C_3\Delta y^2$, where C_1, C_2 and C_3 are constants.*

Proof. Just apply Theorem A.5 for the function $q(x, y, t^n)$. □

In order to check the consistency of 2D-FV, it is useful to use the notion of discrete divergence.

Definition 3.5 (Discrete divergence)**.** *For Problem 3.3, we define the discrete divergence as a $(\Delta x, \Delta y)$-grid function $\mathbb{D}^n(Q^n, u^n, v^n) \in \mathbb{P}_\nu^{N\times M}$ given by:*

$$\mathbb{D}^n_{ij}(Q^n, u^n, v^n) = \frac{1}{\Delta t}\left(\frac{\delta_i F^n_{ij}}{\Delta x} + \frac{\delta_j G^n_{ij}}{\Delta y}\right), \quad i = 1, \ldots, N, \quad j = 1, \ldots, M. \tag{3.14}$$

With the aid of the discrete divergence, we may rewrite Equation (3.13) as:

$$Q^{n+1} = Q^n - \Delta t \mathbb{D}^n(Q^n, u^n, v^n), \tag{3.15}$$

Notice that if we replace Q^n by the exact solution $Q(t^n)$ in Equation (3.15), we have

$$Q(t^{n+1}) = Q(t^n) - \Delta t \mathbb{D}^n(Q(t^n), u^n, v^n) - \Delta t \tau^n, \tag{3.16}$$

where $\tau^n \in \mathbb{P}_\nu^{N\times M}$ is the local truncation error (LTE). Rearranging the terms of Equation (3.16), we obtain:

$$\tau^n = \frac{Q(t^{n+1}) - Q(t^n)}{\Delta t} - \mathbb{D}^n(Q(t^n), u^n, v^n). \tag{3.17}$$

We define the consistency of the 2D-FV scheme as follows.

Definition 3.6 (Consistency)**.** *Let us consider the framework of Problem 3.3. A 2D-FV scheme is said to be consist in the p-norm if for any sequence of $(\Delta x^{(k)}, \Delta y^{(k)}, \Delta t^{(k)}, \lambda)$-discretizations,*

$k \in \mathbb{N}$, *with* $\lim_{k\to\infty} \Delta x^{(k)} = \lim_{k\to\infty} \Delta y^{(k)} = \lim_{k\to\infty} \Delta t^{(k)} = 0$, *we have:*

$$\lim_{k\to\infty} \left[\max_{1\le n\le N_T^{(k)}} \|\tau^n\|_{p,\Delta x^{(k)}\times\Delta y^{(k)}} \right] = 0,$$

and it is said to be consistent with order d *in the* $p-$*norm if*

$$\max_{1\le n\le N_T^{(k)}} \|\tau^n\|_{p,\Delta x^{(k)}\times\Delta y^{(k)}} = \mathcal{O}(\Delta x^d).$$

The relationship between consistency and convergence is explained in Section A.4. If q satisfies Equation (3.7), it can be observed that consistency is equivalent to the following:

$$\max_{1\le n\le N_T^{(k)}} \|\delta^n - \mathbb{D}^n(Q^n, u^n, v^n)\|_{p,\Delta x^{(k)}\times\Delta y^{(k)}} = \mathcal{O}(\Delta x^d),$$

where $\delta^n \in \mathbb{P}_\nu^{N\times M}$ is defined in Equation (3.2). Therefore, we can determine whether a 2D-FV scheme is consistent by comparing the discrete divergence to the exact divergence.

3.3 Dimension splitting

This Section aims to demonstrate how a 2D-FV scheme, such as the one presented in Problem 3.3, can be constructed using 1D-FV schemes through a technique known as dimension splitting. Before introducing the dimension splitting scheme proposed by Lin and Rood (1996), it is helpful to examine general operator splitting schemes, as the dimension splitting technique is a specific instance of operator splitting methods.

For a given time interval $[0,T]$, we utilize a Δt-temporal grid. Let us consider the abstract Cauchy problem.

$$\begin{cases} \frac{dq}{dt}(t) & = Aq(t), \quad t \in [t^n, t^{n+1}], \\ q(t^n) & = q_n, \end{cases}$$

for $n = 0, \dots, N_T - 1$, where $q(t) \in \mathcal{B}$ for some Banach space $\mathcal{B}$, and $A : \mathcal{B} \to \mathcal{B}$ is a linear operator following the framework of Richtmyer and Morton (1968, Chapter 3). We are interested in finding $q(t^{n+1})$ given q_n. Assuming that $A = A_1 + A_2$ for two linear operators $A_1, A_2 : \mathcal{B} \to \mathcal{B}$, we consider the following abstract Cauchy sub-problems:

$$\begin{cases} \frac{dq^1}{dt}(t) & = A_1 q(t), \quad t \in [t^n, t^{n+1}], \\ q^1(t^n) & = q_n, \end{cases}$$

and

$$\begin{cases} \frac{dq^{21}}{dt}(t) & = A_2 q(t), \quad t \in [t^n, t^{n+1}], \\ q^{21}(t^n) & = q^1(t^{n+1}). \end{cases}$$

Then we can approximate $q(t_0 + \Delta t)$ as $q^{21}(t^n + \Delta t)$ with an error of $\mathcal{O}(\Delta t)$ if A_1 and A_2

do not commute. Otherwise, this method is exact. This approach is known as Lie-Trotter splitting. It's worth noting that the Lie-Trotter splitting can also be performed in reverse order when solving the sub-problems:

$$\begin{cases} \frac{dq^2}{dt}(t) & = A_2 q(t), \quad t \in [t^n, t^{n+1}], \\ q^2(t^n) & = q_n, \end{cases}$$

and

$$\begin{cases} \frac{dq^{21}}{dt}(t) & = A_1 q(t), \quad t \in [t^n, t^{n+1}], \\ q^{12}(t^n) & = q^1(t^{n+1}), \end{cases}$$

and again we estimate $q(t^{n+1})$ by $q^{12}(t^{n+1})$ with error $\mathcal{O}(\Delta t)$. As noted by Strang (1968), we can consider the following equation to approximate $q(t^{n+1})$ using a second-order ($\mathcal{O}(\Delta t^2)$) symmetric scheme:

$$q^*(t^{n+1}) = \frac{q^{21}(t^{n+1}) + q^{12}(t^{n+1})}{2}, \tag{3.18}$$

This scheme is referred to as the average Lie-Trotter splitting (Holden et al., 2010). The process of averaging two Lie-Trotter splittings is a specific case of methods known as weighted sequential splitting methods in the literature. Furthermore, this scheme averaging process can be extended to achieve higher-order schemes (Jia & Li, 2011). For an analysis of the accuracy of weighted sequential splitting methods, we recommend referring to Csomós et al. (2005).

It is worth noting that one of the most commonly used second-order splitting schemes in the literature is the Strang splitting (Strang, 1968). This scheme requires solving three sub-problems per time-step, with one of them at time $t_n + \frac{\Delta t}{2}$. In contrast, the average Lie-Trotter splitting requires solving four sub-problems per time-step. Consequently, the Strang splitting is computationally more efficient. However, as we will observe in this Chapter, when applied to the linear advection equation, the average Lie-Trotter splitting allows for a modification that eliminates a splitting error arising from considering a constant scalar field and non-divergent velocity (Lin & Rood, 1996).

3.3.1 Lie-Trotter splitting using PPM

To move towards the scheme from Lin and Rood (1996), let us consider Problem 3.1 in its differential form (Equation (3.7)). We are going to consider $N + 2\nu$ one-dimensional advection equations in the x-direction:

$$[\partial_t q^x + \partial_x(uq^x)](x, y_j, t) = 0,$$

for $j = -\nu + 1, \ldots, M + \nu$, and the $N + 2\nu$ one-dimensional advection equations in the y-direction

$$[\partial_t q^y + \partial_y(vq^y)](x_i, y, t) = 0,$$

for, $i = -\nu + 1, \ldots, N + \nu$.

We shall assume that these problems are solved using a 1D-FV scheme as in Problem 2.4

with the PPM numerical flux functions $\mathfrak{F}^{PPM,x}_{i+\frac{1}{2},j}[Q^n_{\times,j},\tilde{c}^{x,n}]$ and $\mathfrak{F}^{PPM,y}_{i,j+\frac{1}{2}}[Q^n_{i,\times},\tilde{c}^{y,n}]$, respectively, where $\tilde{c}^{x,n}_{i+\frac{1}{2},j}$ is the time-averaged CFL used in the departure point estimation in the x direction and $\tilde{c}^{y,n}_{i,j+\frac{1}{2}}$ is the time-averaged CFL used in the departure point estimation in the y direction, assuming that the CFL number is less than one (see Equation (2.66)). The time-averaged CFL numbers are computed using the schemes **DP1** (Subsection 2.3.1) and **DP2** (Subsection 2.3.2), applied separately in the x and y directions.

The values $q^x_{L,ij}$, $q^x_{R,ij}$, $q^y_{L,ij}$, and $q^y_{R,ij}$, which approximate values of q, namely $q_{i-\frac{1}{2},j}$, $q_{i+\frac{1}{2},j}$, $q_{i,j-\frac{1}{2}}$, $q_{i,j+\frac{1}{2}}$, respectively, are computed using one of the schemes **hord0** and **hord8** as described in Sections 2.4.1 and 2.4.2, again applied separately in the x and y directions. These approximations are expected to be second-order accurate because the given average values are computed on the 2D control volume Ω_{ij} instead of the 1D control volumes X_i or Y_j.

As in Section 2.5, in Equations (2.64) and (2.64), we define the perturbation values in the x direction as:

$$b^x_{L,i,j} = q^x_{L,i,j} - Q^n_{ij}, \tag{3.19}$$
$$b^x_{R,i,j} = q^x_{R,i,j} - Q^n_{ij}, \tag{3.20}$$

and the perturbation values in the y direction as:

$$b^y_{L,i,j} = q^y_{L,i,j} - Q^n_{ij}, \tag{3.21}$$
$$b^y_{R,i,j} = q^y_{R,i,j} - Q^n_{ij}. \tag{3.22}$$

Then, we may express the 1D fluxes in x direction as in Equation (2.66), namely:

$$\mathfrak{F}^{PPM,x}_{i+\frac{1}{2},j}[Q^n_{\times,j},\tilde{c}^{x,n}] = \begin{cases} Q^n_{ij} + (1-\tilde{c}^{x,n}_{i+\frac{1}{2},j})\big(b^x_{R,i,j} - \tilde{c}^{x,n}_{i+\frac{1}{2},j}(b^x_{L,i,j}+b^x_{R,i,j})\big), & \text{if } \tilde{c}^{x,n}_{i+\frac{1}{2},j} \geq 0, \\ Q^n_{i+1,j} + (1+\tilde{c}^{x,n}_{i+\frac{1}{2},j})\big(b^x_{L,i+1,j} + \tilde{c}^{x,n}_{i+\frac{1}{2},j}(b^x_{L,i+1,j}+b^x_{R,i+1,j})\big), & \text{if } \tilde{c}^{x,n}_{i+\frac{1}{2},j} < 0, \end{cases} \tag{3.23}$$

for $i = 0,\dots,N$, $j = -\nu+1,\dots,M+\nu$, and the 1D fluxes in y direction reads

$$\mathfrak{F}^{PPM,y}_{i,j+\frac{1}{2}}[Q^n_{i,\times},\tilde{c}^{y,n}] = \begin{cases} Q^n_{ij} + (1-\tilde{c}^{y,n}_{i,j+\frac{1}{2}})\big(b^y_{R,i,j} - \tilde{c}^{y,n}_{i,j+\frac{1}{2}}(b^y_{L,i,j}+b^y_{R,i,j})\big), & \text{if } \tilde{c}^{y,n}_{i,j+\frac{1}{2}} \geq 0, \\ Q^n_{i,j+1} + (1+\tilde{c}^{y,n}_{i,j+\frac{1}{2}})\big(b^y_{L,i,j+1} + \tilde{c}^{y,n}_{i,j+\frac{1}{2}}(b^y_{L,i,j+1}+b^y_{R,i,j+1})\big), & \text{if } \tilde{c}^{y,n}_{i,j+\frac{1}{2}} < 0, \end{cases} \tag{3.24}$$

for $i = -\nu+1,\dots,N+\nu$, $j = 0,\dots,M$. For both hord0 and hord8 schemes, we set $\nu = 3$.

We introduce the auxiliary grid functions $\mathbf{F}$ and $\mathbf{G}$, both belonging to $\mathbb{R}^{N\times M}_\nu$, given by:

$$\mathbf{F}_{ij}[Q^n,\tilde{c}^{x,n}] = -\frac{1}{|\Omega_{ij}|}\left(\mathcal{A}^x_{i+\frac{1}{2},j}\mathfrak{F}^{PPM,x}_{i+\frac{1}{2},j}[Q^n_{\times,j},\tilde{c}^{x,n}] - \mathcal{A}^x_{i-\frac{1}{2},j}\mathfrak{F}^{PPM,x}_{i-\frac{1}{2},j}[Q^n_{\times,j},\tilde{c}^{x,n}]\right),$$

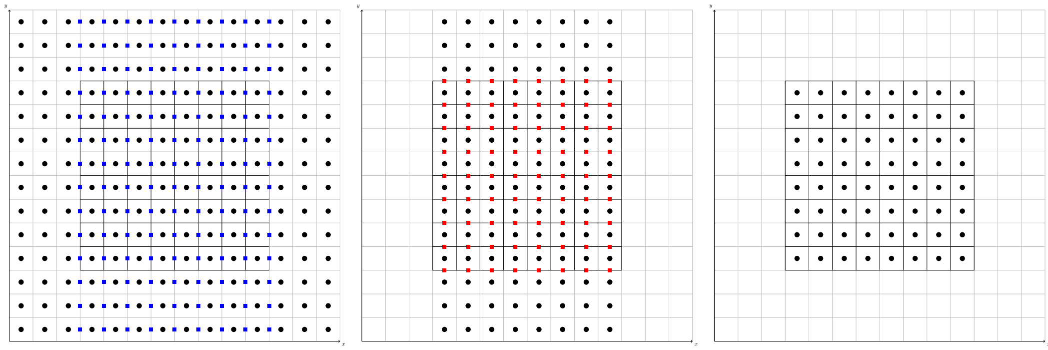
for $i = 1,\dots,N$, $j = -\nu+1,\dots,M+\nu$, and

$$\mathbf{G}_{ij}[Q^n,\tilde{c}^{y,n}] = -\frac{1}{|\Omega_{ij}|}\left(\mathcal{A}^y_{i,j+\frac{1}{2}}\mathfrak{F}^{PPM,y}_{i,j+\frac{1}{2}}[Q^n_{i,\times},\tilde{c}^{y,n}] - \mathcal{A}^y_{i,j-\frac{1}{2}}\mathfrak{F}^{PPM,y}_{i,j-\frac{1}{2}}[Q^n_{i,\times},\tilde{c}^{y,n}]\right),$$

for $i = -\nu + 1, \dots, N + \nu$ $j = 1, \dots, M$. We are using the notations $|\Omega_{ij}| = \Delta x \Delta y$ to represent the area of the control volume and

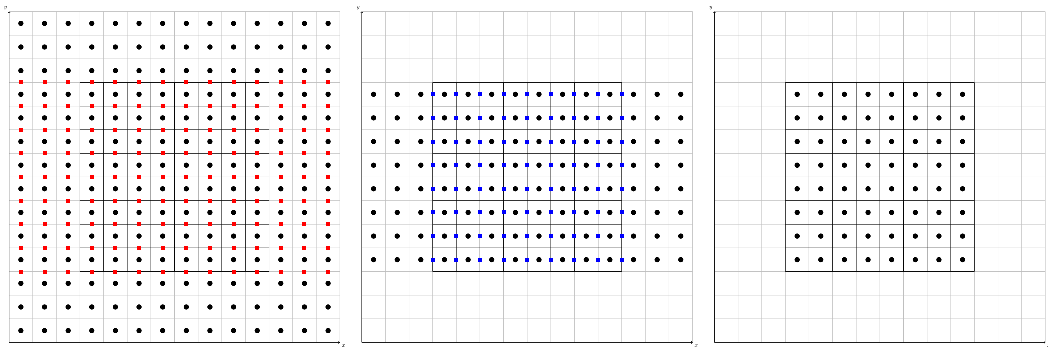
$$\mathcal{A}^{x}_{i+\frac{1}{2},j} = \tilde{c}^{x,n}_{i+\frac{1}{2},j} \Delta x \Delta y,$$
$$\mathcal{A}^{y}_{i,j+\frac{1}{2}} = \tilde{c}^{y,n}_{i,j+\frac{1}{2}} \Delta x \Delta y.$$

This notation shall be useful when we consider these schemes on the cubed-sphere in Chapter 5. Hence, the operators $\mathbf{F}$ and $\mathbf{G}$ represent the numerical updates added to the average values at time level n to obtain their values at time level $n+1$ when solving the advection equation in the x and y directions, respectively.



(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.*

Figure 3.2: *Illustration of the Lie-Trotter splitting applied in the x direction (operator $\mathbf{F}$) and then in the y direction (operator $\mathbf{G}$). 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 DP1 departure point scheme. If the DP2 scheme is used, an additional layer of wind ghost values should be added at each boundary in (a) and (b).*



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

Figure 3.3: *Similar to Figure 3.2 but considering the Lie-Trotter splitting in reverse order.*

The Lie-Trotter splitting is obtained by solving the advection in the x direction

$$Q^{x,n}_{ij} = Q^n_{ij} + \mathbf{F}_{ij}[Q^n, \tilde{c}^{x,n}],$$

for $j = \nu+1, \dots, M+\nu$, $i = 1, \dots, N$ (Figure 3.2b), and then we advect in the y direction with initial data $Q^{x,n}$

$$Q^{yx,n}_{ij} = Q^{x,n}_{ij} + \mathbf{G}_{ij}[Q^{x,n}, \tilde{c}^{y,n}],$$

for $j = 1, \dots, M$, $i = 1, \dots, N$ (Figure 3.2c). To get the average Lie-Trotter splitting we repeat the process in the reverse order by solving the advection equation in the y direction

$$Q^{y,n}_{ij} = Q^{n}_{ij} + \mathbf{G}_{ij}[Q^{n}, \tilde{c}^{y,n}],$$

for $i = -\nu+1, \dots, N+\nu$, $j = 1, \dots, M$ (Figure 3.3b), and then we advect in the x-direction with initial data $Q^{y,n+1}$

$$Q^{xy,n}_{ij} = Q^{y,n}_{ij} + \mathbf{F}_{ij}[Q^{y,n}, \tilde{c}^{x,n}],$$

for $i = 1, \dots, N$, $j = 1, \dots, M$ (Figure 3.3c) and thus we have the average Lie-Trotter solution:

$$\begin{aligned} Q^{n+1} = \frac{(Q^{xy,n} + Q^{yx,n})}{2} &= Q^n + \frac{1}{2}\mathbf{F}[Q^n, \tilde{c}^{x,n}] + \frac{1}{2}\mathbf{G}[Q^n, \tilde{c}^{y,n}] \\ &+ \frac{1}{2}\mathbf{F}\left[Q^n + \mathbf{G}[Q^n, \tilde{c}^{y,n}], \tilde{c}^{x,n}\right] + \frac{1}{2}\mathbf{G}\left[Q^n + \mathbf{F}[Q^n, \tilde{c}^{x,n}], \tilde{c}^{y,n}\right]. \end{aligned} \tag{3.25}$$

This scheme shall be referred to in this work as the average Lie-Trotter (**LT**) scheme. Finally, we point out that this scheme could be built using any other 1D numerical flux function, but we focus on PPM since this is what is used in FV3.

3.3.2 Elimination of splitting error for a constant scalar field and non-divergent wind

Let us, for an instant, assume that **F** and **G** are linear in their first input. This implies that Equation (3.25) may be rewritten as:

$$\begin{aligned} Q^{n+1} &= Q^n + \mathbf{F}[Q^n, \tilde{c}^{x,n}] + \mathbf{G}[Q^n, \tilde{c}^{y,n}] \\ &+ \frac{1}{2}\mathbf{F}\left[\mathbf{G}[Q^n, \tilde{c}^{y,n}], \tilde{c}^{x,n}\right] + \frac{1}{2}\mathbf{G}\left[\mathbf{F}[Q^n, \tilde{c}^{x,n}], \tilde{c}^{y,n}\right]. \end{aligned} \tag{3.26}$$

The numerical flux functions defined in Chapter 2 are indeed linear in the input Q if there are no monotonic constraints, that is, when we use hord0, implying that **F** and **G** are both linear in this case. We are going to consider Equation (3.26) even when there are monotonic constraints, to analyse the scheme when $\boldsymbol{u}$ is non-divergent ($\nabla \cdot \boldsymbol{u} = 0$) and the scalar field is equal to a constant $\overline{q}$. Then the solution remains constant. Since the wind is non-divergent, it follows from the Helmholtz decomposition theorem that there exists a stream function $\psi \in C^2$ such that

$$\begin{aligned}u(x,y,t) &= -\partial_y \psi(x,y,t),\\ v(x,y,t) &= \partial_x \psi(x,y,t).\end{aligned}$$

Then, we may compute the wind using centered-finite differences

$$\begin{aligned}u^n_{i+\frac{1}{2},j} &= -\left(\frac{\psi^n_{i+\frac{1}{2},j+\frac{1}{2}} - \psi^n_{i+\frac{1}{2},j-\frac{1}{2}}}{\Delta y}\right),\\ v^n_{i,j+\frac{1}{2}} &= \frac{\psi^n_{i+\frac{1}{2},j+\frac{1}{2}} - \psi^n_{i-\frac{1}{2},j+\frac{1}{2}}}{\Delta x},\end{aligned}$$

and thus the following discrete divergence free condition holds

$$\frac{\delta_i u^n_{ij}}{\Delta x} + \frac{\delta_j v^n_{ij}}{\Delta y} = 0. \tag{3.27}$$

Notice that this identity holds for the time-averaged winds if we assume that that $\tilde{u}^n$ and $\tilde{v}^n$ are computed using DP1. If we use DP2, this identity is no longer valid. Now, using the fact that the scalar field is supposed to be constant, we have:

$$\begin{aligned}\mathbf{F}_{ij}[\overline{q}, \tilde{c}^{x,n}] &= -\overline{q}\lambda\delta_i \tilde{u}^n_{ij},\\ \mathbf{G}_{ij}[\overline{q}, \tilde{c}^{y,n}] &= -\overline{q}\lambda\delta_j \tilde{v}^n_{ij},\end{aligned}$$

recalling that $\lambda = \frac{\Delta t}{\Delta x} = \frac{\Delta t}{\Delta y}$. Applying $\mathbf{G}$ and $\mathbf{F}$ again, we have:

$$\begin{aligned}\mathbf{G}_{ij}\left[\mathbf{F}[\overline{q}, \tilde{c}^{y,n}], \tilde{c}^{x,n}\right] &= \overline{q}\lambda^2\left(\tilde{v}^n_{i,j+\frac{1}{2}}\mathfrak{F}^{PPM,y}_{i,j+\frac{1}{2}}[\delta_i\tilde{u}^n_{ij}, \tilde{c}^{y,n}] - \tilde{v}^n_{i,j-\frac{1}{2}}\mathfrak{F}^{PPM,y}_{i,j-\frac{1}{2}}[\delta_i\tilde{u}^n_{ij}, \tilde{c}^{y,n}]\right)\\ &= \overline{q}\lambda^2\delta_i\left(\tilde{v}^n_{ij}\mathfrak{F}^{PPM,y}_{ij}[\delta_i\tilde{u}^n_{ij}, \tilde{c}^{y,n}]\right),\end{aligned}$$

$$\begin{aligned}\mathbf{F}_{ij}\left[\mathbf{G}[\overline{q}, \tilde{c}^{x,n}], \tilde{c}^{y,n}\right] &= \overline{q}\lambda^2\left(\tilde{u}^n_{i,j+\frac{1}{2}}\mathfrak{F}^{PPM,x}_{i,j+\frac{1}{2}}[\delta_j\tilde{v}^n_{ij}, \tilde{c}^{x,n}] - \tilde{u}^n_{i,j-\frac{1}{2}}\mathfrak{F}^{PPM,x}_{i,j-\frac{1}{2}}[\delta_j\tilde{v}^n_{ij}, \tilde{c}^{x,n}]\right)\\ &= \overline{q}\lambda^2\delta_j\left(\tilde{u}^n_{ij}\mathfrak{F}^{PPM,x}_{ij}[\delta_j\tilde{v}^n_{ij}, \tilde{c}^{x,n}]\right).\end{aligned}$$

However, if we compute the updated solution using Equation (3.26) we have that the error is given by

$$\begin{aligned}Q^{n+1}_{ij} - \overline{q} &= -\Delta t\left(\frac{\delta_i\tilde{u}^n_{ij}}{\Delta x} + \frac{\delta_j\tilde{v}^n_{ij}}{\Delta y}\right) - \frac{\overline{q}}{2}\lambda^2\left(\delta_j\left(\tilde{u}^n_{ij}\mathfrak{F}^{PPM,x}_{ij}[\delta_j\tilde{v}^n_{ij}, \tilde{c}^{x,n}]\right) + \delta_i\left(\tilde{v}^n_{ij}\mathfrak{F}^{PPM,y}_{ij}[\delta_i\tilde{u}^n_{ij}, \tilde{c}^{y,n}]\right)\right)\\ &= -\frac{\overline{q}}{2}\lambda^2\left(\delta_j\left(\tilde{u}^n_{ij}\mathfrak{F}^{PPM,x}_{ij}[\delta_j\tilde{v}^n_{ij}, \tilde{c}^{x,n}]\right) + \delta_i\left(\tilde{v}^n_{ij}\mathfrak{F}^{PPM,y}_{ij}[\delta_i\tilde{u}^n_{ij}, \tilde{c}^{y,n}]\right)\right),\end{aligned} \tag{3.28}$$

To eliminate this error, Lin and Rood (1996) proposed modifying the Equation (3.25) to

$$Q^{n+1} = Q^n + \frac{1}{2}\mathbf{F}[Q^n, \tilde{c}^{x,n}] + \frac{1}{2}\mathbf{G}[Q^n, \tilde{c}^{y,n}] + \frac{1}{2}\mathbf{F}\left[Q^n + \mathbf{g}[Q^n, \tilde{c}^{y,n}], \tilde{c}^{x,n}\right] + \frac{1}{2}\mathbf{G}\left[Q^n + \mathbf{f}[Q^n, \tilde{c}^{x,n}], \tilde{c}^{y,n}\right], \tag{3.29}$$

where $\mathbf{f}$ and $\mathbf{g}$ are called inner advective operators. In this work, we shall consider the inner advective operator proposed by Putman and Lin (2007) (hereafter referred to as **PL**). The PL scheme is currently used in the FV3 dynamical core. Also, notice that the LT scheme is equivalent to the PL scheme but uses $\mathbf{f} = \mathbf{F}$ and $\mathbf{g} = \mathbf{G}$. All the expressions for each inner advective operator mentioned are shown in Table 3.1. It is easy to see that the PL operator

Scheme	$\mathbf{f}_{ij}(Q^n, \tilde{c}^{x,n})$	$\mathbf{g}_{ij}(Q^n, \tilde{c}^{y,n})$
LT	$\mathbf{F}_{ij}(Q^n, \tilde{c}^{x,n})$	$\mathbf{G}_{ij}(Q^n, \tilde{c}^{y,n})$
PL	$-Q^n_{ij} + \frac{Q^n_{ij}+\mathbf{F}_{ij}(Q^n,\tilde{c}^{x,n})}{1-\frac{1}{\lvert\Omega_{ij}\rvert}\left(\mathcal{A}^x_{i+\frac{1}{2},j}-\mathcal{A}^x_{i-\frac{1}{2},j}\right)}$	$-Q^n_{ij} + \frac{Q^n_{ij}+\mathbf{G}_{ij}(Q^n,\tilde{c}^{y,n})}{1-\frac{1}{\lvert\Omega_{ij}\rvert}\left(\mathcal{A}^y_{i,j+\frac{1}{2}}-\mathcal{A}^y_{i,j-\frac{1}{2}}\right)}$

Table 3.1: *Expression of the inner advective operators considered in this work. LT stands for the average Lie-Trotter scheme, while PL stands for the scheme from Putman and Lin (2007).*

eliminates the term multiplied by λ^2 that appeared in Equation (3.28) when we apply these operators to a constant grid function Q^n and a non-divergent velocity field in Equation (3.28). Therefore, these inner advective operators eliminate the splitting error for a constant field and a non-divergent velocity field, making this scheme exact in this case if we use DP1 to compute the departure points. If we use DP2 with PL splitting, Equation (3.28) will introduce a first-order error since the discrete divergence-free condition (Equation (3.27)) for the time-averaged winds of DP2 does not hold in this case. We shall see this in the numerical experiments (Section 3.4). We point out that although the LT scheme has an error in Equation (3.28), it is a second-order error, since this scheme is generally second-order accurate (Holden et al., 2010), provided that the 1D flux is second-order, which shall be the case if we use the DP2 scheme as discussed in Chapter 2.

3.4 Numerical experiments

To assess the dimension-splitting schemes LT and PL introduced previously, we are going to consider the linear advection equation on the spatial domain $[-\frac{L}{2}, \frac{L}{2}] \times [-\frac{L}{2}, \frac{L}{2}]$ and in the time interval $[0, T]$, with biperiodic boundary conditions, where $L = \frac{\pi}{2}R$. Here, $R = 6.371 \times 10^6$ meters, representing the Earth's radius, and $T = 1036800$ seconds, equivalent to 12 days. The spatial domain spans approximately 10^4 kilometers in both directions, which correspond to approximately the lengths of a cubed-sphere panel, as shall be seen in Chapter 4.

For the 1D schemes, we will consider the FV-SL schemes **hord0** (Subsection 2.4.1) and **hord8** (Subsection 2.4.2), each tested with both departure point schemes **DP1** (Subsection 2.3.1) and **DP2** (Subsection 2.3.2). We employ $(\Delta x^{(k)}, \Delta y^{(k)}, \lambda)$-discretizations with $\Delta x^{(k)} =$

$\Delta y^{(k)} = \frac{L}{N^{(k)}}$, $N^{(k)} = 48 \times 2^k$, $k = 0, \ldots, 4$. We introduce the relative error in the maximum norm:

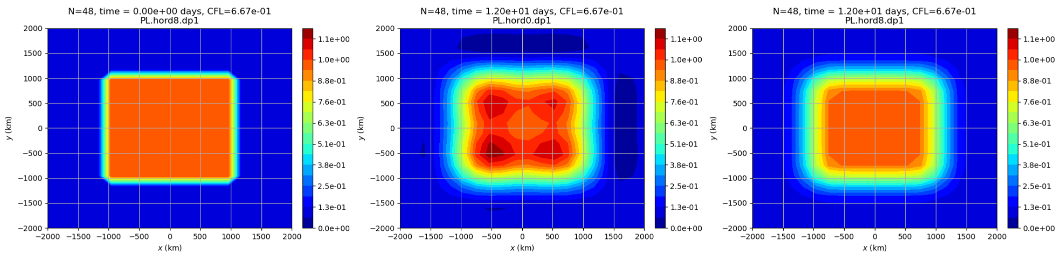
$$E_k = \frac{\|Q^n - Q^0\|_{\infty,\Delta x \times \Delta y}}{\|Q^0\|_{\infty,\Delta x \times \Delta y}}.$$

The convergence rate, as defined in Section 2.6, and the preservation of total mass variation with machine precision are considered in all experiments presented here. It is worth noting that in error computation, we employ centroid values instead of exact average values to avoid the computation of analytical integrals. This approximation, as discussed in Proposition 3.1, introduces a second-order error.

3.4.1 Square wave with constant wind advection

For the initial test, a constant velocity $\boldsymbol{u} = \left(\frac{L}{T}, \frac{L}{T}\right)$ is considered. The IC is a rectangular profile (refer to Figure 3.4a) given by:

$$q_0(x, y) = \begin{cases} 1 & \text{if } (x, y) \in [-0.1L, 0.1L] \times [-0.1L, 0.1L], \\ 0.1 & \text{otherwise.} \end{cases} \tag{3.30}$$



(a) *IC* **(b)** *PL-hord0* **(c)** *PL-hord8.*

Figure 3.4: *Linear advection experiment using a constant velocity* $\boldsymbol{u} = \left(\frac{L}{T}, \frac{L}{T}\right)$, *a CFL number set to* 0.67, *and a grid resolution of* $N = M = 48$. *The initial condition is given by Equation* (3.30). *We run this test with the PL splitting combined with the schemes hord0 (b) and hord8 (c). The figures display the advected profile after 12 days (one time period). The initial condition is depicted in (a).*

We will employ a time step of 14400 seconds and set $N = M = 48$, resulting in a CFL number approximately equal to 0.67. The exact solution of Problem 3.1 in this scenario is $q_0((x, y) - \boldsymbol{u}t)$. Due to the constant velocity field, all splitting schemes introduced in Section 3.3 are equivalent. Therefore, we only consider the PL splitting. Additionally, it is evident that the Lie-Trotter splitting is exact in this case (see, for example, LeVeque, 1990, p. 202-203), meaning no splitting error is introduced. For the 1D schemes, we utilize DP1 to compute the departure point, as this scheme is exact when the velocity is constant.

The conclusions drawn from this test closely resemble those of the first 1D test discussed in Section 2.6.1. This similarity arises because no splitting error is introduced when the velocity remains constant. Figure 3.4c illustrates that PL splitting maintains monotonicity, particularly noticeable when using the monotonic 1D scheme hord8.

3.4.2 Flow deformation with nondivergent wind

For a first variable velocity testing, we consider two Gaussian hills given by:

$$\begin{aligned} q_0(x,y) = 0.1 + 0.9 \exp\left(-10\sin^2\left(\pi\left(\frac{x}{L}-0.1\right)\right)\right)\exp\left(-10\sin^2\left(\pi\frac{y}{L}\right)\right)+ \\ \exp\left(-10\sin^2\left(\pi\left(\frac{x}{L}+0.1\right)\right)\right)\exp\left(-10\sin^2\left(\pi\frac{y}{L}\right)\right), \end{aligned} \tag{3.31}$$

defined in $[-\frac{L}{2}, \frac{L}{2}] \times [-\frac{L}{2}, \frac{L}{2}]$, whose graph is shown in Figure 3.5.

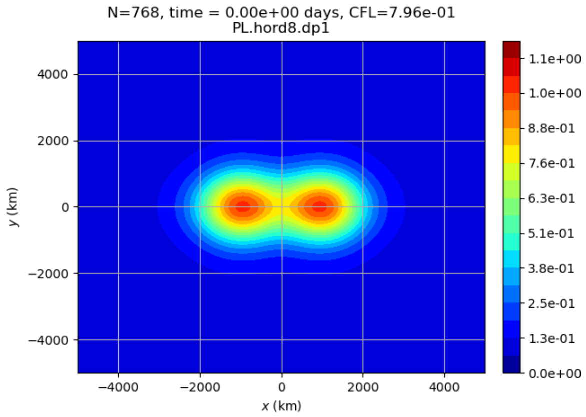


Figure 3.5: *Two Gaussian hills IC (Equation (3.31)).*

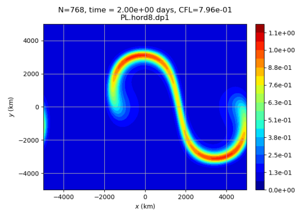
We consider the Cartesian version of the deformational flow test case on the sphere from Nair and Lauritzen (2010) proposed by Y. Chen et al. (2017). The velocity is given by:

$$\begin{cases} u(x,y,t) &= -c\frac{L}{T}\sin^2(\alpha_1)\sin\left(\frac{\pi y}{L}\right)\cos\left(\frac{\pi y}{L}\right)\cos\left(\frac{\pi t}{T}\right)+\frac{L}{T}, \\ v(x,y,t) &= -2c\frac{L}{T}\sin(\alpha_1)\cos(\alpha_1)\cos^2\left(\frac{\pi y}{L}\right)\cos\left(\frac{\pi t}{T}\right), \end{cases} \tag{3.32}$$

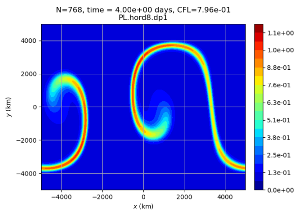
where $\alpha_1 = 2\pi\left(\frac{x}{L} - \frac{t}{T}\right)$, $c = 10$. Y. Chen et al. (2017) uses periodic boundary conditions in the $x-$direction and zero-gradient in the $y-$direction. However, we will employ biperiodic boundary conditions to simplify the problem. This velocity field is divergence-free, and deforms the initial condition. After T time units (12 days in our case), the scalar field returns to its initial position and shape, allowing us to compute the error. Notice that in Equation (3.32), we have added a constant wind $\frac{L}{T}$ in the component u to prevent error cancellation, as discussed by Nair and Lauritzen (2010).

Figure 3.6 illustrates the results obtained using two Gaussian hills and the velocity field from Equation (3.32). We employed a high-resolution grid with $N = 768$, along with the PL-DP1-hord8 scheme, to demonstrate the behavior of the test. The Figure shows the deformation of the scalar field over time, eventually returning to its initial position.

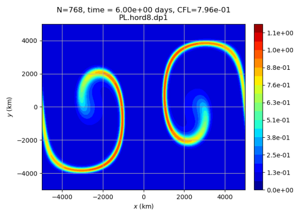
To investigate the error convergence, we employ time steps $\Delta t^{(k)} = \frac{5400}{2^k}$ for $k = 0, \ldots, 4$, and the spatial discretization as described at the beginning of Section 3.4, resulting in a CFL number approximately equal to 0.79.



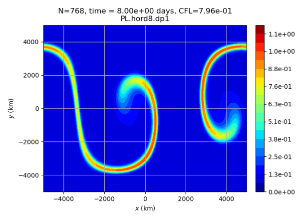
(a) $t = 2$ *days.*



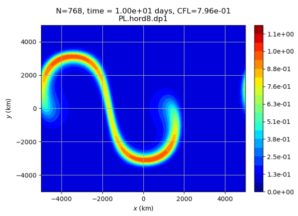
(b) $t = 4$ *days.*



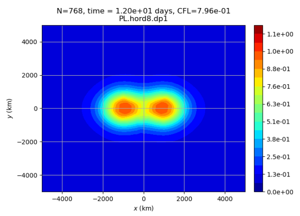
(c) $t = 6$ *days.*



(d) $t = 8$ *days.*



(e) $t = 10$ *days.*



(f) $t = 12$ *days.*

Figure 3.6: *Linear advection experiment using the velocity from Equation* (3.32)*, a CFL number equal to* 0.79*,* $N = 768$ *cells, and the IC is given by Equation* (3.31) *These figures show the advected profile at 2* (3.6a)*, 4* (3.6b)*, 6* (3.6c)*, 8* (3.6d)*, 10* (3.6e)*, and 12* (3.6f) *days. We are using the PL-DP1-hord8 scheme.*

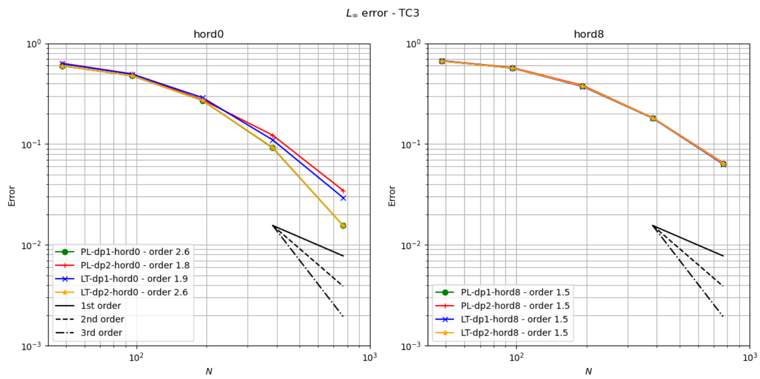


Figure 3.7: L_∞ *error for the two Gaussian hills (Equation 3.31) with the velocity from Equation* (3.32)*. Schemes using hord0 are on the left, and hord8 are on the right. The PL scheme with DP1 is in green, and with DP2 is in red. The LT scheme with DP1 is in blue, and with DP2 is in yellow.*

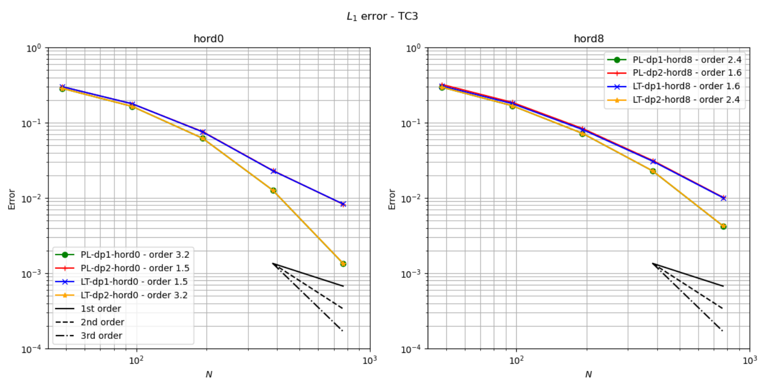


Figure 3.8: *Similar to Figure 3.7 but considering the* L_1 *error.*

We can observe from Figure 3.7 that for hord0, PL-DP1 and LT-DP2 have smaller error and higher convergence order than PL-DP2 and LT-DP1. However, when considering the hord8 scheme, all the schemes have the same error in L_∞ norm. The errors in L_1 norm (Figure 3.8) exhibit a similar behavior; the only difference is that PL-DP1 and LT-DP2 have smaller errors than PL-DP2 and LT-DP1, along with higher convergence order.

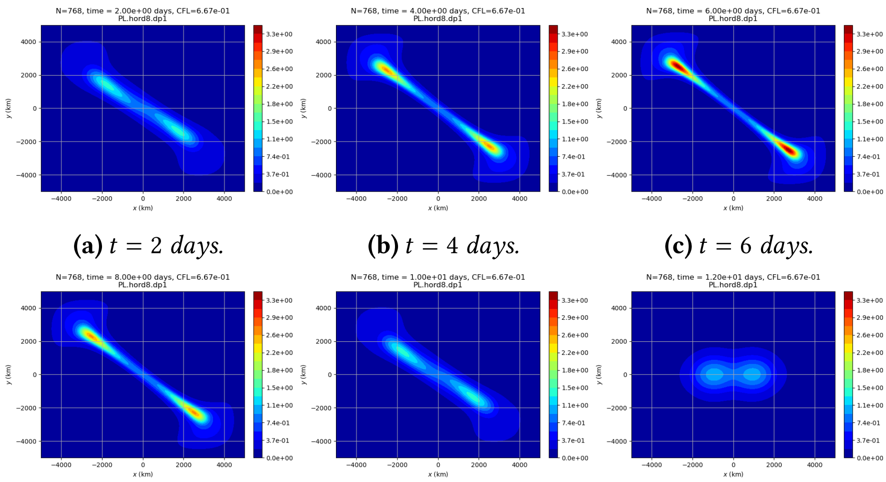
3.4.3 Flow deformation with divergent wind

For a second variable velocity testing, we consider two Gaussian hills given by Equation (3.31) and the following wind:

$$\begin{cases} u(x,y,t) &= -\frac{L}{T}\cos^2\left(\frac{\pi x}{L}\right)\sin\left(\frac{2\pi y}{L}\right)\cos\left(\frac{\pi t}{T}\right), \\ v(x,y,t) &= -\frac{L}{T}\cos^2\left(\frac{\pi y}{L}\right)\sin\left(\frac{2\pi x}{L}\right)\cos\left(\frac{\pi t}{T}\right). \end{cases} \tag{3.33}$$

This test is based on the planar test from Nair and Lauritzen (2010), but we adapt it to make the wind divergent. Figure 3.9 illustrates the results obtained using two Gaussian hills and the velocity field from Equation (3.33), similarly to Figure 3.6. Again, the IC returns to its initial position after 12 days, allowing us to compute the error.

We employ time steps $\Delta t^{(k)} = \frac{14400}{2^k}$ for $k = 0, \ldots, 4$, to analyse the error convergence, along with the spatial discretization as described at the beginning of Section 3.4, resulting in a CFL number approximately equal to 0.67.



(a) $t = 2$ *days.* **(b)** $t = 4$ *days.* **(c)** $t = 6$ *days.*

(d) $t = 8$ *days.* **(e)** $t = 10$ *days.* **(f)** $t = 12$ *days.*

Figure 3.9: *Similar to Figure 3.6 but using the wind from Equation (3.33).*

We can observe from Figure 3.10 that for hord0, PL-DP1 has the bigger error, while LT-DP2 has the smaller error and the highest convergence rate. However, when considering the hord8 scheme, all the schemes have almost the same error in L_∞ norm. Regarding the error in L_1 norm (Figure 3.11), we can see that for hord8, LT-DP2 achieves second-order accuracy, while PL-DP1 achieves only first order. Finally, the schemes PL-DP2 and LT-DP1 have the same errors for both hord0 and hord8, in both L_∞ and L_1 norms.

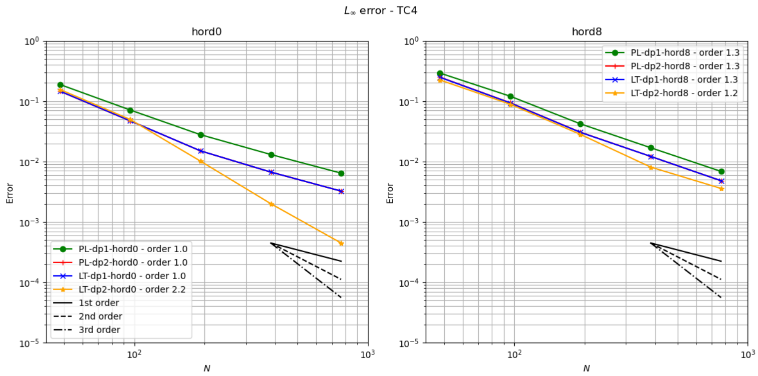


Figure 3.10: *Similar to Figure 3.7 but using the wind from Equation* (3.33).

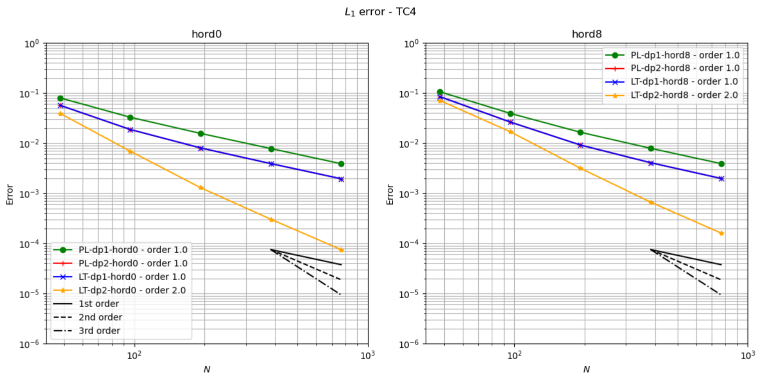


Figure 3.11: *Similar to Figure 3.10 but considering the L_1 error.*

3.5 Concluding remarks

In this Chapter, we introduced the dimension-splitting method, which replaces the solution of the 2D advection equation with the solution of multiple 1D advection equations, resulting in more cost-effective 2D-FV schemes. For our simulations, we adopted the 1D FV-SL scheme based on PPM to solve the 1D equations.

We modified the average of two Lie-Trotter splittings, which is second-order accurate, to ensure the preservation of a constant scalar field with a divergence-free velocity, following the works of Lin and Rood (1996) and Putman and Lin (2007). This modification addresses the limitation of the classical averaging Lie-Trotter splitting and follows the methodology used in FV3.

Based on the simulation with constant velocity, we concluded that all the splitting schemes are equivalent and do not introduce any splitting errors. In fact, the splittings are exact in this case. We observed that all the conclusions from the 1D simulations hold true in the 2D case as well, with mass conservation and monotonicity being preserved when using the monotonic limiters in the 1D subproblems.

In the simulation with variable velocity, we conducted two flow deformation test cases. For the divergence-free test, the schemes PL-DP1 and LT-DP2 showed similar behavior and performed better than PL-DP2 and LT-DP1 in all error metrics analyzed here. However, for the velocity with non-zero divergence, we observed that the scheme PL-DP1 achieved only first-order accuracy and had larger errors than PL-DP2 and LT-DP1. This limitation is because the PL-DP1 method is designed to be accurate for divergence-free winds. This test highlights this limitation because we have divergence. The scheme LT-DP2 showed better error performance, achieving second-order accuracy regardless of the non-divergence-free condition in the wind. LT-DP2 also showed second-order accuracy in the L_1 norm when we employed the monotonic 1D flux, while PL-DP1 achieved first order.

In summary, the scheme PL-DP1, which is currently used in FV3 as the 2D advection solver, showed second-order accuracy for divergence-free winds, with LT-DP2 exhibiting similar behavior. However, for non-divergent free winds, LT-DP2 demonstrated second-order accuracy, while PL-DP1 achieved only first order.

Chapter 4

Cubed-sphere grids

So far, we have described the dimension-splitting technique in Chapter 3 for solving the advection equation on the plane. Our current goal is to apply these schemes to solve the advection equation on the sphere. Consequently, we need to introduce a grid over the sphere. In order to facilitate the extension of dimension-splitting techniques onto the sphere, we require a logical Cartesian coordinate system, at least locally.

We point out that dimension-splitting schemes could be formulated in unstruced grids (see for instance Herzfeld and Engwirda (2023)). A good reason to use a locally Cartesian grid is to avoid problems, such as the lack of convergence of the divergence operator among others, that may arise in some grid cells within those grids (P. Peixoto, 2016; P. Peixoto & Barros, 2013; Weller, 2012). Also, a logical Cartesian coordinate system eases the process of higher-order interpolation, which can be more complicated on a spherical unstructured grid, requiring tangent plane approximations (P. S. Peixoto & Barros, 2014; W. C. Skamarock & Gassmann, 2011).

The scheme proposed by Lin and Rood (1996) was originally implemented on latitude-longitude grids, and the FV dynamical core was elucidated in Lin (2004). The latitude-longitude grids exhibit convergence of meridians at the poles, necessitating the utilization of the Semi-Lagrangian formulation of PPM for larger CFL numbers, as discussed in Section 2.5, to overcome the CFL restriction imposed by the poles. However, this approach needs the processes in a parallel domain decomposition of the latitude-longitude grid to utilize more data at the poles, resulting in non-parallel efficiency. Therefore, Putman and Lin (2007) proposed considering the cubed-sphere (CS, hereafter) instead. The CS grid is more uniform, thus not exhibiting a strong CFL condition anywhere. This eliminates the need for the Semi-Lagrangian formulation of PPM, which is better to parallel efficiency, and led to the development of the FV3 core.

The CS grid was originally proposed by Sadourny (1972) and was reinvestigated by Ronchi et al. (1996) and Rančić et al. (1996). As is usual for Planotic grids, we start with a Platonic solid, in this case, a cube, which is circumscribed in a sphere. We then project its faces onto the sphere. The original CS, called the equidistant CS, was proposed by Sadourny (1972) but resulted in a non-uniform grid. To address this issue, a solution was proposed by introducing angular coordinates, leading to a quasi-uniform grid known as the equiangular CS. The cubed sphere consists of six panels, each one having a local Cartesian coordinate system. As we pointed out before, this makes it easier to extend methods from the plane to the sphere. In fact, Putman and Lin (2007) extends the dimension splitting technique from Lin and Rood (1996), as presented in Chapter 3, to the CS.

There are essentially two major challenges when working with the CS grid:

1. The non-orthogonal grid system: This challenge is primarily related to the appearance of metric terms in the equations. It adds computational cost and often requires conversions between contravariant and covariant components of a velocity field.

2. The discontinuity of the coordinate system at the cube edges: This is perhaps the most problematic challenge. Computing stencils along the cube edges becomes challenging due to the discontinuous nature of the coordinate system.

One possible approach to compute stencils at the edges is to extend the local coordinate of each panel to its neighboring panels, adding ghost cells in the halo region. In the case of the equiangular CS, ghost cell values lie on the same geodesics containing the data from the neighboring panels. This allows for the use of one-dimensional high-order Lagrange interpolation to compute the stencils at the edges. This approach has been extensively used in the literature (X. Chen, 2021; Croisille, 2013; Katta et al., 2015a, 2015b) and was initially introduced by Ronchi et al. (1996). This approach is referred to as **duo-grid**, as named by X. Chen (2021). Alternatively, Putman and Lin (2007) uses extrapolation for the PPM reconstruction values near the cube edges. Another approach that avoids the need for interpolation or extrapolation near the edges is the conformal CS developed by Rančić et al. (1996). While this grid leads to an orthogonal and continuous coordinate system near the edges, it generates grid singularities near the cube corners, similar to the pole problem. An improved and more uniform conformal grid, called the Uniform Jacobian cubed sphere, was later proposed by Rančić et al. (2017). Each approach is likely to generate grid imprinting, and one of the goals of this work is to investigate the amount of grid imprinting produced by different methods.

This Chapter aims to review and investigate the geometrical properties of the CS. We start with a basic review of the CS mappings in Section 4.1. In Section 4.2, we introduce the CS grids and investigate its geometrical properties. Section 4.3 investigates how we can apply 1D Lagrange interpolation using the adjacent panels data to obtain values of a scalar/vector field on ghost cells. In Section 4.3.4, compare the current extrapolation used in FV3 with Lagrange when using the PPM reconstruction to remap the values from centers to edges on the cubed-sphere cells. Final thoughts are presented in Section 4.4.

4.1 Cubed-sphere mappings

4.1.1 Mapping between the cube and sphere

We start this section by introducing the mapping between the cube and the sphere, which will divide the sphere into 6 quadrilaterals, also called panels, and allow us to tessellate the sphere into smaller quadrilaterals for panels. Given $R > 0$, we denote the sphere of radius R centered at the origin of $\mathbb{R}^3$ as:

$$\mathbb{S}^2_R = \{P = (X, Y, Z) \in \mathbb{R}^3 \,:\, X^2 + Y^2 + Z^2 = R^2\}.$$

We consider a parameter the family of maps $\boldsymbol{\Psi}_p \,:\, [-1, 1] \times [-1, 1] \to \mathbb{S}^2_R$, $p = 1, \dots, 6$, where:

$$\boldsymbol{\Psi}_1(x, y) = \frac{R}{\sqrt{1 + x^2 + y^2}}(1, x, y),$$

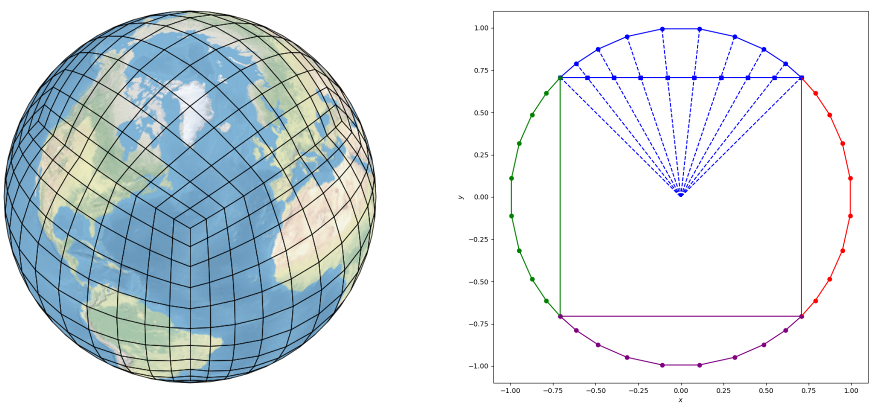
$$\boldsymbol{\Psi}_2(x, y) = \frac{R}{\sqrt{1 + x^2 + y^2}}(-x, 1, y),$$

$$\boldsymbol{\Psi}_3(x, y) = \frac{R}{\sqrt{1 + x^2 + y^2}}(-1, -x, y),$$

$$\boldsymbol{\Psi}_4(x, y) = \frac{R}{\sqrt{1 + x^2 + y^2}}(x, -1, y),$$

$$\boldsymbol{\Psi}_5(x, y) = \frac{R}{\sqrt{1 + x^2 + y^2}}(-y, x, 1),$$

$$\boldsymbol{\Psi}_6(x, y) = \frac{R}{\sqrt{1 + x^2 + y^2}}(y, x, -1).$$



(a) *Gridlines of the cube to the sphere mapping* **(b)** *Cube and sphere mapping for $Z = 0$.*

Figure 4.1: *(a) Illustration of the resulting cube-to-sphere mapping and (b) illustration of the cube-to-sphere projection.*

The set of 6 maps $\{\boldsymbol{\Psi}_p, p = 1, \dots, 6\}$ allow us to cover the sphere (Figure 4.1). Here p

denotes a panel, and they are defined and orientated as Figure 4.2 shows. Then, we can represent a point on the sphere using the cubed-sphere coordinates (x, y, p).

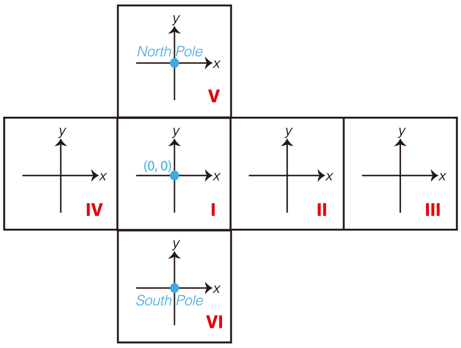


Figure 4.2: *Cubed-sphere panels definition and orientation. Figure taken from Jung et al. (2019).*

The derivative of the maps $\mathbf{\Psi}_p$ are given by:

$$d\mathbf{\Psi}_1(x,y) = \frac{R}{(1+x^2+y^2)^{3/2}} \begin{bmatrix} -x & -y \\ 1+y^2 & -xy \\ -xy & 1+x^2 \end{bmatrix},$$

$$d\mathbf{\Psi}_2(x,y) = \frac{R}{(1+x^2+y^2)^{3/2}} \begin{bmatrix} -(1+y^2) & xy \\ -x & -y \\ -xy & 1+x^2 \end{bmatrix},$$

$$d\mathbf{\Psi}_3(x,y) = \frac{R}{(1+x^2+y^2)^{3/2}} \begin{bmatrix} x & y \\ -(1+y^2) & xy \\ -xy & 1+x^2 \end{bmatrix},$$

$$d\mathbf{\Psi}_4(x,y) = \frac{R}{(1+x^2+y^2)^{3/2}} \begin{bmatrix} 1+y^2 & -xy \\ x & y \\ -xy & 1+x^2 \end{bmatrix},$$

$$d\mathbf{\Psi}_5(x,y) = \frac{R}{(1+x^2+y^2)^{3/2}} \begin{bmatrix} xy & -(1+x^2) \\ 1+y^2 & -xy \\ -x & -y \end{bmatrix},$$

$$d\mathbf{\Psi}_6(x,y) = \frac{R}{(1+x^2+y^2)^{3/2}} \begin{bmatrix} -xy & 1+x^2 \\ 1+y^2 & -xy \\ x & y \end{bmatrix}.$$

With the aid of the derivative, we may define a basis of tangent vectors $\{\partial_x \mathbf{\Psi}, \partial_y \mathbf{\Psi}\}$ on each point on the sphere by:

$$\partial_x \mathbf{\Psi}(x,y,p) = d\mathbf{\Psi}_p(x,y) \cdot \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad \partial_y \mathbf{\Psi}(x,y,p) = d\mathbf{\Psi}_p(x,y) \cdot \begin{bmatrix} 0 \\ 1 \end{bmatrix}.$$

Notice that the matrix

$$G_{\mathbf{\Psi}}(x,y) := [d\mathbf{\Psi}_p(x,y)]^T d\mathbf{\Psi}_p(x,y) = \frac{R^2}{(1+x^2+y^2)^2} \begin{bmatrix} 1+x^2 & -xy \\ -xy & 1+y^2 \end{bmatrix},$$

does not depend on p. This matrix is known as metric tensor. It is easy to see that:

$$G_\Psi(x,y) = \begin{bmatrix} \langle \partial_x \boldsymbol{\Psi}_p, \partial_x \boldsymbol{\Psi}_p \rangle & \langle \partial_x \boldsymbol{\Psi}_p, \partial_y \boldsymbol{\Psi}_p \rangle \\ \langle \partial_x \boldsymbol{\Psi}_p, \partial_y \boldsymbol{\Psi}_p \rangle & \langle \partial_y \boldsymbol{\Psi}_p, \partial_y \boldsymbol{\Psi}_p \rangle \end{bmatrix}, \tag{4.1}$$

where $\langle \cdot, \cdot \rangle$ denotes the standard inner product of $\mathbb{R}^3$, and that $G_\Psi(x,y)$ is positive-definite, $\forall (x,y) \in [-1,1] \times [-1,1]$. The Jacobian of the metric tensor $G_\Psi(x,y)$, denoted by $\sqrt{\mathfrak{g}_\Psi}$ and called metric term, is then given by:

$$\sqrt{\mathfrak{g}_\Psi}(x,y) := \sqrt{|\det G_\Psi(x,y)|} = \frac{R^2}{(1+x^2+y^2)^{3/2}},$$

Now let us assume that we have a function $\beta : [-\alpha, \alpha] \to [-1,1]$, for some positive $\alpha > 0$, supposed to be bijective and C^1 with inverse C^1 as well. That is, β is a change of coordinates. Let us consider $\boldsymbol{\Phi}_p : [-\alpha,\alpha] \times [-\alpha,\alpha] \to \mathbb{S}^2_R$, given by

$$\boldsymbol{\Phi}_p(x,y) := \boldsymbol{\Psi}_p\big(\beta(x), \beta(y)\big).$$

It follows from the chain rule that:

$$d\boldsymbol{\Phi}_p(x,y) = d\boldsymbol{\Psi}_p\big(\beta(x),\beta(y)\big) \cdot \operatorname{diag}\big(\beta'(x), \beta'(y)\big),$$

where $\operatorname{diag}(\beta'(x), \beta'(y))$ is a diagonal 2×2 matrix with diagonal entries given by $\beta'(x)$ and $\beta'(y)$. We also have that tangent vector basis $\{\partial_x \boldsymbol{\Phi}_p, \partial_y \boldsymbol{\Phi}_p\}$ satisfying

$$\begin{aligned} \partial_x \boldsymbol{\Phi}_p(x,y) &= \beta'(x) \cdot \partial_x \boldsymbol{\Psi}_p\big(\beta(x),\beta(y)\big), \\ \partial_y \boldsymbol{\Phi}_p(x,y) &= \beta'(y) \cdot \partial_y \boldsymbol{\Psi}_p\big(\beta(x),\beta(y)\big). \end{aligned}$$

The metric tensor of $\boldsymbol{\Phi}_p$ is defined as G_Ψ in Equation (4.1):

$$G(x,y) = \begin{bmatrix} \langle \partial_x \boldsymbol{\Phi}_p, \partial_x \boldsymbol{\Phi}_p \rangle & \langle \partial_x \boldsymbol{\Phi}_p, \partial_y \boldsymbol{\Phi}_p \rangle \\ \langle \partial_x \boldsymbol{\Phi}_p, \partial_y \boldsymbol{\Phi}_p \rangle & \langle \partial_y \boldsymbol{\Phi}_p, \partial_y \boldsymbol{\Phi}_p \rangle \end{bmatrix}.$$

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

$$\begin{aligned} \sqrt{\mathfrak{g}}(x,y) &= \beta'(x)\beta'(y) \sqrt{\mathfrak{g}_\Psi}\big(\beta(x),\beta(y)\big) \\ &= \beta'(x)\beta'(y) \frac{R^2}{\big(1+\beta(x)^2+\beta(y)^2\big)^{3/2}}, \end{aligned}$$

which may also be expressed as

$$\sqrt{\mathfrak{g}}(x,y) = \|\partial_x \boldsymbol{\Phi}_p\| \|\partial_y \boldsymbol{\Phi}_p\| \sin\alpha(x,y,p), \tag{4.2}$$

where α is the angle between $\partial_x \boldsymbol{\Phi}_p$ and $\partial_y \boldsymbol{\Phi}_p$ that satisfies

$$\cos\alpha(x,y,p) = \frac{\langle \partial_x \boldsymbol{\Phi}_p, \partial_x \boldsymbol{\Phi}_p \rangle}{\|\partial_x \boldsymbol{\Phi}_p\| \|\partial_y \boldsymbol{\Phi}_p\|}.$$

4.2 Cubed-sphere grids

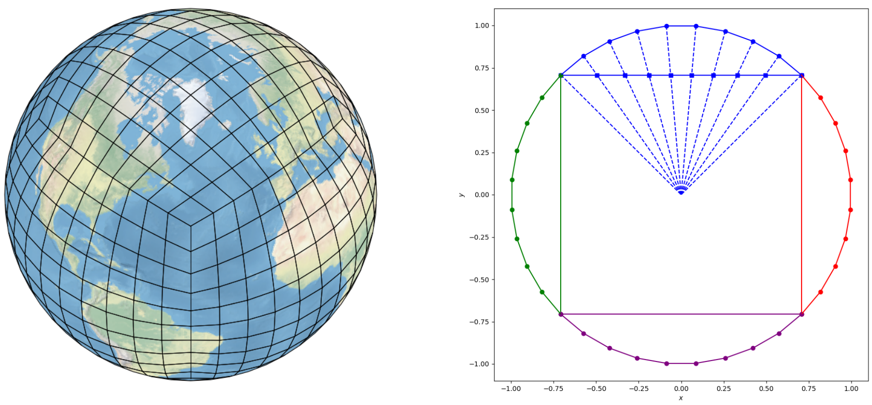
Now that we have established between the mapping between the cube and sphere with coordinate changes, we may introduce the cubed-sphere grids proposed in the literature.

4.2.1 Equidistant cubed-sphere

The first cubed-sphere grid was proposed by Sadourny (1972). This grid is obtained by using $\beta(x) = x$, $\alpha = 1$ in the $\mathbf{\Phi}_p$ mapping described in Section 4.1.1. This grid partitions the cube face into equally spaced points and projects them onto the sphere, as illustrated in Figure 4.1, hence the name equidistant. We shall denote this grid by **g1** since the parameter **grid_type** in FV3 is set equal to 1 to use this grid.

4.2.2 Equiangular cubed-sphere

Another cubed-sphere mapping is the equiangular mapping, introduced by Ronchi et al. (1996), which leads to a more uniform grid. This grid is obtained by considering the mapping $\mathbf{\Phi}_p$ described in Section 4.1.1 with $\beta(x) = \tan x$ and $\alpha = \frac{\pi}{4}$. In this case, $\beta(x)$ represents the angular coordinates, and the cube-sphere is obtained by partitioning the angle between grid points equally, as illustrated in Figure 4.3, hence the name equiangular. This grid is denoted by **g2**, for the same reason of the notation **g1**.

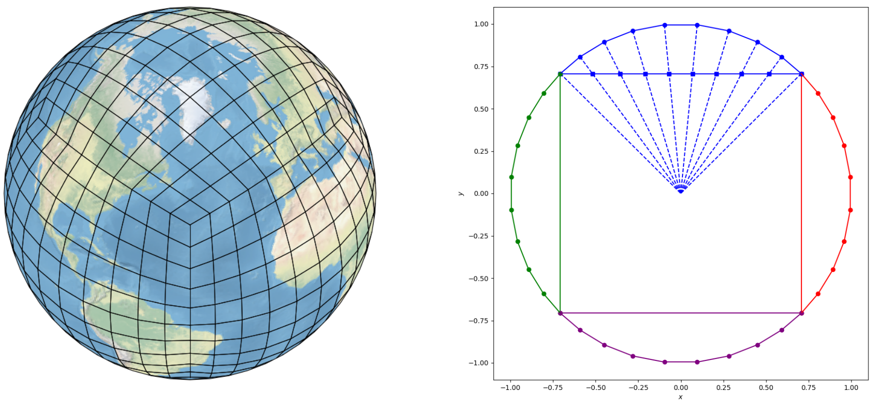


(a) *Gridlines of the cube to the sphere equiangular mapping* **(b)** *Cube and sphere equiangular mapping for* $Z = 0$.

Figure 4.3: *(a) Illustration of the resulting cube-to-sphere mapping and (b) illustration of the cube-to-sphere projection using the equiangular mapping.*

4.2.3 Equi-edge cubed-sphere

An equiangular cubed-sphere modification X. Chen (2021) by using $\beta(x) = \sqrt{2}\tan x$ and $\alpha = \arcsin\left(\frac{1}{\sqrt{3}}\right)$. Figure 4.4 illustrates the equi-edge mapping.



(a) *Gridlines of the cube to the sphere equi-edge mapping* **(b)** *Cube and sphere equi-edge mapping for* $Z = 0$.

Figure 4.4: *(a) Illustration of the resulting cube-to-sphere mapping and (b) illustration of the cube-to-sphere projection using the equi-edge mapping.*

The idea behind the equi-edge cubed-sphere lies in partitioning the edges of the spherical cube equally, and then generating the other cells, hence the name equi-edge. This grid is denoted by **g0**, for the same reason of the notation **g1**. Also, this grid leads to more uniform cells after applying the grid stretching option of FV3 (X. Chen, 2021; L. M. Harris et al., 2016).

4.2.4 Geometric properties

We will utilize the notation introduced in Section 3.1.1 throughout this Chapter. We shall used the earth radius $R = 6.371 \times 10^6$ meters. The parameter ν represents a non-negative integer indicating the number of ghost cell layers in each panel boundary, called halo size. To generate the cubed-sphere, we consider a $(\Delta x, \Delta y)$-grid denoted by $\Omega_{\Delta x, \Delta y} = (\Omega_{ij})_{i,j=-\nu+1,\ldots,N+\nu}$, where $\Delta x = \Delta y$, and it covers the domain Ω. A control volume of the cubed-sphere is denoted by Ω_{ijp}, defined as follows:

$$\Omega_{ijp} = \Phi_p(\Omega_{ij}) \quad -\nu + 1 \leq i, j \leq N + \nu, \quad 1 \leq p \leq 6.$$

The cubed-sphere grid refers to the collection of control volumes $(\Omega_{ijp})_{i,j=-\nu+1,\ldots,N+\nu}^{p=1,\ldots,6}$. In Figures 4.1, 4.3 and 4.4 examples of the cubed-sphere grids are depicted, excluding the ghost cells. These grids are generated using the equidistant, equiangular and equi-edge mappings for $N = 10$.

We will denote the area of Ω_{ijp} by $|\Omega_{ij}|$. Notice that the area does not depend on the panel due to the grid symmetry. We also define the diameter of a cell as $2\sqrt{\frac{|\Omega_{ij}|}{\pi}}$, which corresponds to the diameter of a circle with area $|\Omega_{ij}|$. The control volume area is given by:

$$|\Omega_{ij}| = \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}} \sqrt{\mathfrak{g}}(x, y)\, dx\, dy = |\hat{\Omega}_{ij}| + \mathcal{O}(\Delta x^2), \tag{4.3}$$

where $|\hat{\Omega}_{ij}| = \sqrt{\mathfrak{g}}(x_i, y_j)\Delta x \Delta y$, and the last equality follows from Proposition 3.1. In tables 4.1, 4.2, and 4.3, we display the diameters of the grids g0, g1, and g2 for $N = 48 \times 2^k$, where $k = 0, \ldots, 4$. These values of N are considered in this work. Similarly, in tables 4.4, 4.5, and 4.6, we display the areas.

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

Table 4.1: *Mean diameter, minimum diameter, and maximum diameter for different values of N considering the equi-edge grid (g0).*

N	Mean Length (km)	Min Length (km)	Max Length (km)	$\frac{\text{Max}}{\text{Min}}$
48	215	134	305	2.2780
96	107	66	151	2.2791
192	53	33	75	2.2794
384	26	16	37	2.2795
768	13	8	18	2.2795

Table 4.2: *As Table 4.1 but considering the equidistant grid (g1).*

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

Table 4.3: *As Table 4.1 but considering the equiangular grid (g2).*

We can observe that in terms of areas and diameters of the cells, grid g1 is the least uniform, while g2 is the most uniform grid. Grid g0 is more uniform than g1, but the maximum/minimum ratio of the areas is almost 2.3. Despite this, g0 is operational in some applications of FV3 (X. Chen, 2021; L. Harris et al., 2021), such as, for instance, the Next Generation Global Prediction System (NGGPS) (Zhou et al., 2019), because this grid is expected to produce less grid imprinting due to its greater uniformity near the cubed edges. Therefore, in this thesis, we shall constrain our attention only to grids g0 and g2, since g2 is ideally more uniform and g0 is currently used in FV3. In Figure 4.5, we illustrate the areas of both grids, g0 and g2. We can observe that the areas of g0 exhibit a higher gradient near the cube corners, while g2 appears to have a higher gradient near the middle of the cube edges.

N	Mean Area (km^2)	Min Area (km^2)	Max Area (km^2)	$\frac{\text{Max}}{\text{Min}}$
48	38033	24113	55650	2.3078
96	9364	5902	13628	2.3090
192	2323	1460	3371	2.3093
384	578	363	838	2.3094
768	144	90	209	2.3094

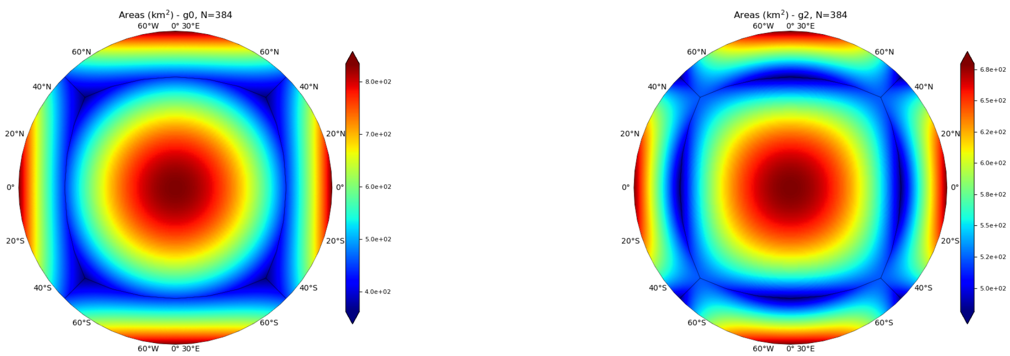
Table 4.4: *Mean area, minimum area, and maximum area for different values of N considering the equi-edge grid (g0).*

N	Mean Area (km^2)	Min Area (km^2)	Max Area (km^2)	$\frac{\text{Max}}{\text{Min}}$
48	37762	14145	73403	5.1891
96	9331	3462	17985	5.1944
192	2319	856	4450	5.1957
384	578	213	1106	5.1960
768	144	53	276	5.1961

Table 4.5: *As Table 4.4 but considering the equidistant grid (g1).*

N	Mean Area (km^2)	Min Area (km^2)	Max Area (km^2)	$\frac{\text{Max}}{\text{Min}}$
48	38269	32062	45327	1.4137
96	9393	7847	11096	1.4141
192	2327	1941	2745	1.4142
384	579	482	682	1.4142
768	144	120	170	1.4142

Table 4.6: *As Table 4.4 but considering the equiangular grid (g2).*



(a) *g0 grid areas (km^2).*

(b) *g2 grid areas (km^2).*

Figure 4.5: *Areas for the grid g0 and g2 using $N = 384$.*

There are four types of grid points on the cubed-sphere that we need to compute: the

A-grid, B-grid, C-grid, and D-grid points. These names are based on the Arakawa grids (Arakawa & Lamb, 1977). The locations of these points are illustrated in Figure 4.6 for the g2 grid.

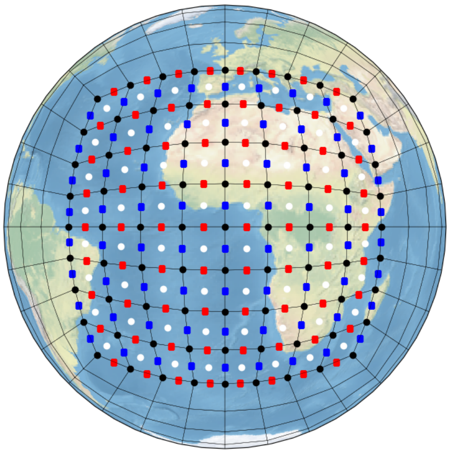


Figure 4.6: *Illustration of the A-grid (white) B-grid (black), C-grid (blue) and D-grid (red) points for the g2 grid with $N = 10$.*

One possible approach is to use the cubed-sphere mapping to generate these points, based on the grid points projected onto the sphere from the plane. These points shall be denoted using a superscript 'c'. The B-grid points represent the corners of the control volume, namely

$$\Phi^c_{i+\frac{1}{2},j+\frac{1}{2},p} := \Phi_p(x_{i+\frac{1}{2}}, y_{j+\frac{1}{2}}). \tag{4.4}$$

Similarly, the A grid points are the cell centers:

$$\Phi^c_{ijp} := \Phi_p(x_i, y_j). \tag{4.5}$$

The C-grid points are the midpoints of the edge in the y-direction, namely:

$$\Phi^c_{i+\frac{1}{2},j,p} := \Phi_p(x_{i+\frac{1}{2}}, y_j). \tag{4.6}$$

The D-grid points are the midpoints of the edge in the x-direction, namely:

$$\Phi^c_{i,j+\frac{1}{2},p} := \Phi_p(x_i, y_{j+\frac{1}{2}}). \tag{4.7}$$

The grids g0 and g2, formulated with these grid points, are denoted by **g0.c** and **g2.c**, respectively. We refer to this grid point formulation as cube midpoints, which is a common approach used in the literature (Guo et al., 2014; Katta et al., 2015a, 2015b; Nair et al., 2005; Ullrich et al., 2010).

Another way to compute the grid points, which is used in FV3, is to compute the B-grid points using as before and obtain the A, C and D-grids using the spherical midpoints. These points shall be denoted using a superscript 's'. The B-grid points in this case are:

$$\Phi^s_{i+\frac{1}{2},j+\frac{1}{2},p} := \Phi^c_{i+\frac{1}{2},j+\frac{1}{2},p}. \tag{4.8}$$

The A-grid points are computed by averaging the values of 4 B-grid points:

$$\Phi^s_{ijp} := \frac{\Phi^s_{i+\frac{1}{2},j+\frac{1}{2},p} + \Phi^s_{i+\frac{1}{2},j-\frac{1}{2},p} + \Phi^s_{i-\frac{1}{2},j+\frac{1}{2},p} + \Phi^s_{i-\frac{1}{2},j-\frac{1}{2},p}}{\|\Phi^s_{i+\frac{1}{2},j+\frac{1}{2},p} + \Phi^s_{i+\frac{1}{2},j-\frac{1}{2},p} + \Phi^s_{i-\frac{1}{2},j+\frac{1}{2},p} + \Phi^s_{i-\frac{1}{2},j-\frac{1}{2},p}\|}. \tag{4.9}$$

Similarly, the C-grid points are obtained by averaging the values of 2 B-grid points.

$$\Phi^s_{i+\frac{1}{2},j,p} := \frac{\Phi^s_{i+\frac{1}{2},j+\frac{1}{2},p} + \Phi^s_{i+\frac{1}{2},j-\frac{1}{2},p}}{\|\Phi^s_{i+\frac{1}{2},j+\frac{1}{2},p} + \Phi^s_{i+\frac{1}{2},j-\frac{1}{2},p}\|}. \tag{4.10}$$

and the D-grid points are also given by the average the values of 2 B-grid points:

$$\Phi^s_{i,j+\frac{1}{2},p} := \frac{\Phi^s_{i+\frac{1}{2},j+\frac{1}{2},p} + \Phi^s_{i-\frac{1}{2},j+\frac{1}{2},p}}{\|\Phi^s_{i+\frac{1}{2},j+\frac{1}{2},p} + \Phi^s_{i-\frac{1}{2},j+\frac{1}{2},p}\|}. \tag{4.11}$$

The grids g0 and g2, formulated with these grid points, are denoted by **g0.s** and **g2.s**, respectively. We refer to this grid points formulation as spherical midpoints. One can easily see that:

$$\Phi^s_{i+\frac{1}{2},j,p} = \Phi^c_{i+\frac{1}{2},j,p} + \mathcal{O}(\Delta x^2), \tag{4.12}$$

$$\Phi^s_{i,j+\frac{1}{2},p} = \Phi^c_{i,j+\frac{1}{2},p} + \mathcal{O}(\Delta x^2), \tag{4.13}$$

$$\Phi^s_{i,j,p} = \Phi^c_{i,j,p} + \mathcal{O}(\Delta x^2). \tag{4.14}$$

Then, we should expect similar results when using different grid point formulations, especially when using a high-resolution grid. In g0.c or g2.c grids, the points are aligned along geodesics. This happens because the cube-mapping maps lines on the plane onto geodesics on the sphere. However, one can see that this does not occur on g0.s or g2.s, as the A, C, and D-grid points are not aligned on the same geodesic. Although this misalignment becomes negligible for high resolutions, it impacts ghost cell interpolation accuracy, as we shall see in Section 4.3.

Hereafter, we are going to omit the superscripts 's' and 'c'. Finally, we introduce the following geodesic distances in x and y directions, respectively,

$$\delta x_{ij} = d(\Phi_{i+\frac{1}{2},j,p}, \Phi_{i-\frac{1}{2},j,p}), \tag{4.15}$$

$$\delta y_{ij} = d(\Phi_{i,j+\frac{1}{2},p}, \Phi_{i,j-\frac{1}{2},p}), \tag{4.16}$$

where $d(P,Q) = R \arccos\left(\langle P, Q\rangle\right)$, for $P, Q \in \mathbb{S}^2_R$, and we assume that i and j can be integers or half-integers. Notice that these distances do not depend on the panel due to the grid symmetry. These distances may be represented in terms of the tangent vector norms as:

$$\delta x_{ij} = \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \|\partial_x \Phi_p\|(x, y_j)\, dx, \tag{4.17}$$

$$\delta y_{ij} = \int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}} \|\partial_y \Phi_p\|(x_i, y)\, dy. \tag{4.18}$$

Hence, their midpoint approximations are defined as:

$$\hat{\delta} x_{ij} = \|\partial_x \Phi_{ijp}\| \Delta x, \tag{4.19}$$

$$\hat{\delta} y_{ij} = \|\partial_y \Phi_{ijp}\| \Delta y, \tag{4.20}$$

which are second-order accurate (see Theorem A.4):

$$\delta x_{ij} = \hat{\delta} x_{ij} + \mathcal{O}(\Delta x^2), \tag{4.21}$$

$$\delta y_{ij} = \hat{\delta} y_{ij} + \mathcal{O}(\Delta y^2). \tag{4.22}$$

4.2.5 Duo-grid points

The B duo-grid points are generated by computing the mappings Φ_p for the grid points $(x_{i+\frac{1}{2}}, y_{j+\frac{1}{2}})$ where i and j out of the range 0 to N. When using the cube midpoint formulation, the A, C, and D-grid points are computed analogously. In Figure 4.7, we illustrate the duo-grid points obtained for both g0 and g2 grids. We can observe in Figure 4.7c that the B duo-grid points of g2 are aligned on common geodesics. This property has been known since the work of Ronchi et al. (1996), and similarly, it holds for A, C, and D duo-grid points when using the cube midpoint formulation. This property is very useful because it allows us to use 1D Lagrange interpolation to estimate the duo-grid values using values from neighboring panels, and it has been widely used in the literature (X. Chen, 2021; Croisille, 2013; Katta et al., 2015a, 2015b; Rossmanith, 2006).

However, it is evident from Figure 4.7a that the analogous property does not hold for the g0 grid. To address this problem, X. Chen (2021) proposes modifying the ghost values of the x and y coordinates by mirroring certain points. This generates the new duo-grid points, aligning them on the same geodesic as those from the neighboring panel. More formally, for $g = 1, 2, \dots, \nu$, we introduce the mirrored values

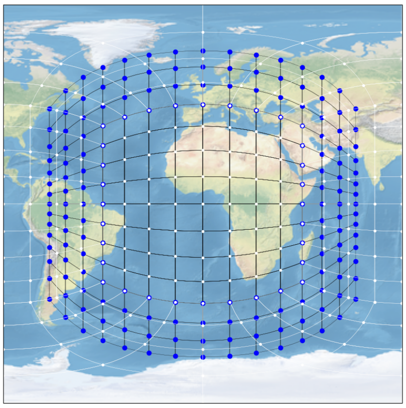
$$\hat{x}_{-g+\frac{1}{2}} = \arctan\left(\frac{1}{\alpha}\tan\left(-\frac{\pi}{2} - \arctan\left(\alpha \tan x_{g+\frac{1}{2}}\right)\right)\right), \tag{4.23}$$

$$\hat{x}_{N+g+\frac{1}{2}} = -\hat{x}_{-g+\frac{1}{2}}, \tag{4.24}$$

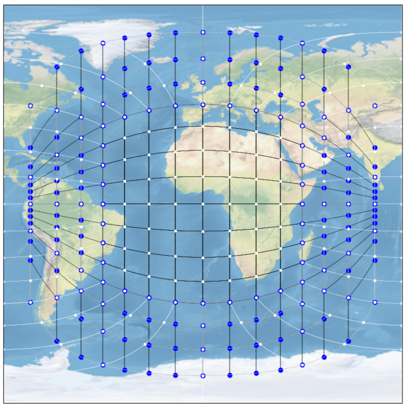
$$\hat{y}_{-g+\frac{1}{2}} = \arctan\left(\frac{1}{\alpha}\tan\left(-\frac{\pi}{2} - \arctan\left(\alpha \tan y_{g+\frac{1}{2}}\right)\right)\right), \tag{4.25}$$

$$\hat{y}_{N+g+\frac{1}{2}} = -\hat{y}_{-g+\frac{1}{2}}, \tag{4.26}$$

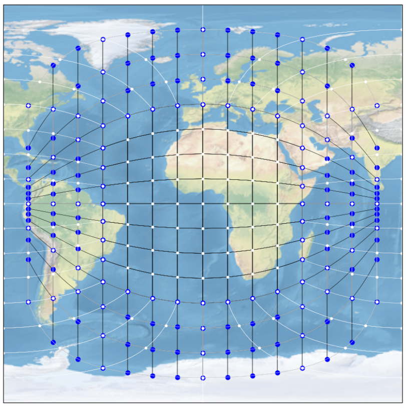
to replace $x_{-g+\frac{1}{2}}, x_{N+g+\frac{1}{2}}, y_{-g+\frac{1}{2}}$ and $y_{N+g+\frac{1}{2}}$, respectively. When computing the grid points using the cube midpoints formulation, the values of x_i and y_j are readjusted similarly. Figure 4.7b illustrates how the modified duo-grid of g0 aligns with the geodesics of neighboring panels just as g2 (Figure 4.7c). Notice, however, that the g0 B-grid will no longer be equally spaced in terms of its x and y coordinates, in contrast to g2, where the x and y coordinates of the B-grid are uniformly spaced. This may require special attention from numerical schemes using g0 near edges due to the loss of uniformity.



(a) *g0 duo-grid.*



(b) *Mirrored g0 duo-grid.*



(c) *g2 duo-grid.*

Figure 4.7: *Duo-grid lines of panel 1 for the g0 (a) and g2 (c). (b) shows the mirrored duo-grid of g0. B duo-grid points are denoted by blue circles, the B-grid points are denoted by white points.*

4.2.6 Tangent vectors on the sphere

The tangent space at $P \in \mathbb{S}^2_R$ is denoted by $T_P\mathbb{S}^2$. It is easy to see that:

$$T_P\mathbb{S}^2_R = \{P_0 \in \mathbb{R}^3 \ : \ \langle P, P_0 \rangle = 0\}.$$

We are going to consider three ways to represent an element of $\mathbb{S}^2_R$: using (X, Y, Z) coordinates, or using (λ, ϕ) latitude-longitude coordinates, or, at last, using the cubed-sphere coordinates (x, y, p), where (x, y) are the cube face coordinates and $p \in \{1, 2, \cdots, 6\}$ stands for a cube panel. We say that a vector field $\boldsymbol{u} : \mathbb{S}^2_R \to \mathbb{R}^3$ is tangent on the sphere if $\boldsymbol{u}(P) \in T_P\mathbb{S}^2_R, \forall P \in \mathbb{S}^2_R$.

Conversions between latitude-longitude and contravariant coordinates

We consider the latitude-longitude mapping $\boldsymbol{\Pi} : [0, 2\pi] \times [-\frac{\pi}{2}, \frac{\pi}{2}] \to \mathbb{S}^2_R$, $\boldsymbol{\Pi} = (\Pi_1, \Pi_2, \Pi_3)$, given by:

$$\Pi_1(\lambda, \phi) = R\cos\phi\cos\lambda, \tag{4.27}$$
$$\Pi_2(\lambda, \phi) = R\cos\phi\sin\lambda, \tag{4.28}$$
$$\Pi_3(\lambda, \phi) = R\sin\phi. \tag{4.29}$$

The derivative or Jacobian matrix of the mapping $\boldsymbol{\Pi}$ is given by:

$$d\boldsymbol{\Pi}(\lambda, \phi) = R\begin{bmatrix} -\cos\phi\sin\lambda & -\sin\phi\cos\lambda \\ \cos\phi\cos\lambda & \sin\phi\sin\lambda \\ 0 & \cos\phi \end{bmatrix}. \tag{4.30}$$

Using this matrix columns, we can define the tangent vectors:

$$\partial_\lambda\boldsymbol{\Pi}(\lambda, \phi) = d\boldsymbol{\Pi}(\lambda, \phi)\begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad \partial_\phi\boldsymbol{\Pi}(\lambda, \phi) = d\boldsymbol{\Pi}(\lambda, \phi)\begin{bmatrix} 0 \\ 1 \end{bmatrix}. \tag{4.31}$$

We normalize the vectors $\partial_\lambda\boldsymbol{\Pi}$ and $\partial_\phi\boldsymbol{\Pi}$ and we obtain unit tangent vectors on the sphere at $\Pi(\lambda, \phi)$:

$$\boldsymbol{e}_\lambda(\lambda, \phi) = \begin{bmatrix} -\sin\lambda \\ \cos\lambda \\ 0 \end{bmatrix}, \quad \boldsymbol{e}_\phi(\lambda, \phi) = \begin{bmatrix} -\sin\phi\cos\lambda \\ -\sin\phi\sin\lambda \\ \cos\phi \end{bmatrix}. \tag{4.32}$$

Let us consider a tangent vector field $\boldsymbol{u} : \mathbb{S}^2_R \to \mathbb{R}^3$ on the sphere, represented as

$$\boldsymbol{u}(\lambda, \phi) = u_\lambda(\lambda, \phi)\boldsymbol{e}_\lambda(\lambda, \phi) + v_\phi(\lambda, \phi)\boldsymbol{e}_\phi(\lambda, \phi). \tag{4.33}$$

We call u_λ as zonal component of the wind and v_ϕ as meridional component of the wind. Or, we may also represent this vector field using the basis obtained by cubed-sphere coordinates:

$$\boldsymbol{u}(x, y, p) = \mathfrak{u}(x, y, p)\partial_x\boldsymbol{\Phi}_p(x, y) + \mathfrak{v}(x, y, p)\partial_y\boldsymbol{\Phi}_p(x, y). \tag{4.34}$$

This representation is known as contravariant representation. In order to relate the latitude-longitude representation with the contravariant representation, we notice that:

$$\partial_x\boldsymbol{\Phi}_p(x, y, p) = \langle\partial_x\boldsymbol{\Phi}_p, \boldsymbol{e}_\lambda\rangle\boldsymbol{e}_\lambda(\lambda, \phi) + \langle\partial_x\boldsymbol{\Phi}_p, \boldsymbol{e}_\phi\rangle\boldsymbol{e}_\phi(\lambda, \phi), \tag{4.35}$$
$$\partial_y\boldsymbol{\Phi}_p(x, y, p) = \langle\partial_y\boldsymbol{\Phi}_p, \boldsymbol{e}_\lambda\rangle\boldsymbol{e}_\lambda(\lambda, \phi) + \langle\partial_y\boldsymbol{\Phi}_p, \boldsymbol{e}_\phi\rangle\boldsymbol{e}_\phi(\lambda, \phi), \tag{4.36}$$

which holds since the vectors $\boldsymbol{e}_\lambda(\lambda, \phi)$ and $\boldsymbol{e}_\phi(\lambda, \phi)$ are orthogonal. Replacing Equations (4.35) and (4.36) in Equation (4.34), we obtain the values (u_λ, v_ϕ) in terms of the contravariant components (u, v) as the following matrix equation:

$$\begin{bmatrix} u_\lambda(\lambda,\phi) \\ v_\phi(\lambda,\phi) \end{bmatrix} = \begin{bmatrix} \langle \partial_x \boldsymbol{\Phi}_p, \boldsymbol{e}_\lambda \rangle & \langle \partial_y \boldsymbol{\Phi}_p, \boldsymbol{e}_\lambda \rangle \\ \langle \partial_x \boldsymbol{\Phi}_p, \boldsymbol{e}_\phi \rangle & \langle \partial_y \boldsymbol{\Phi}_p, \boldsymbol{e}_\phi \rangle \end{bmatrix} \begin{bmatrix} \mathfrak{u}(x,y,p) \\ \mathfrak{v}(x,y,p) \end{bmatrix}. \tag{4.37}$$

Conversely, we may express the contravariant components in terms of latitude-longitude components by inverting Equation (4.37).

In practice when discretizing PDEs on the cubed-sphere, FV3 schemes use the normalized contravariant wind (u, v) given by:

$$\boldsymbol{u}(x,y,p) = u(x,y,p)\boldsymbol{e}_x(x,y,p) + v(x,y,p)\boldsymbol{e}_y(x,y,p), \tag{4.38}$$

where $\boldsymbol{e}_x$ and $\boldsymbol{e}_y$ are the normalized cubed-sphere tangent vectors:

$$\boldsymbol{e}_x(x,y,p) = \frac{\partial_x \boldsymbol{\Phi}_p(x,y)}{\|\partial_x \boldsymbol{\Phi}_p(x,y)\|}, \quad \boldsymbol{e}_y(x,y,p) = \frac{\partial_y \boldsymbol{\Phi}_p(x,y)}{\|\partial_y \boldsymbol{\Phi}_p(x,y)\|}. \tag{4.39}$$

It is easy to see that:

$$u(x,y,p) = \frac{\mathfrak{u}(x,y,p)}{\|\partial_x \boldsymbol{\Phi}_p(x,y)\|}, \quad v(x,y,p) = \frac{\mathfrak{v}(x,y,p)}{\|\partial_y \boldsymbol{\Phi}_p(x,y)\|}. \tag{4.40}$$

The normalized contravariant form is used because it offers greater generality and flexibility when working with optimized cubed-sphere grids, as discussed in Putman and Lin (2007), and stretched grids (L. M. Harris et al., 2016), where explicit expressions of the exact, non-normalized tangent vectors are either available or can be overly complicated. On the other hand, the normalized tangent vectors at grid points may be computed easily in terms of the grid points (see Appendix C2 of X. Chen (2021)). The latitude-longitude representation is related with the normalized contravariant representation by the expression:

$$\begin{bmatrix} u_\lambda(\lambda,\phi) \\ v_\phi(\lambda,\phi) \end{bmatrix} = \begin{bmatrix} \langle \boldsymbol{e}_x, \boldsymbol{e}_\lambda \rangle & \langle \boldsymbol{e}_y, \boldsymbol{e}_\lambda \rangle \\ \langle \boldsymbol{e}_x, \boldsymbol{e}_\phi \rangle & \langle \boldsymbol{e}_y, \boldsymbol{e}_\phi \rangle \end{bmatrix} \begin{bmatrix} u(x,y,p) \\ v(x,y,p) \end{bmatrix}. \tag{4.41}$$

Conversely, we may express the normalized contravariant components in terms of latitude-longitude components by inverting Equation (4.41).

Covariant/contravariant conversion

Let us consider again a tangent vector field $\boldsymbol{u} : \mathbb{S}^2_R \to \mathbb{R}^3$ on the sphere. Its contravariant representation is given by Equation (4.34). The covariant components $(\mathfrak{U}, \mathfrak{V})$ are given by:

$$\mathfrak{U}(x,y,p) = \langle \boldsymbol{u}(x,y,p), \partial_x \boldsymbol{\Phi}_p(x,y,p) \rangle, \tag{4.42}$$

$$\mathfrak{V}(x,y,p) = \langle \boldsymbol{u}(x,y,p), \partial_y \boldsymbol{\Phi}_p(x,y,p) \rangle. \tag{4.43}$$

Replacing Equation (4.34) in Equations (4.42) and (4.43) we obtain the relation between covariant components in terms of the contravariant terms:

$$\begin{bmatrix} \mathfrak{U}(x,y,p) \\ \mathfrak{V}(x,y,p) \end{bmatrix} = \begin{bmatrix} \langle \partial_x \Phi_p, \partial_x \Phi_p \rangle & \langle \partial_x \Phi_p, \partial_y \Phi_p \rangle \\ \langle \partial_x \Phi_p, \partial_y \Phi_p \rangle & \langle \partial_y \Phi_p, \partial_y \Phi_p \rangle \end{bmatrix} \begin{bmatrix} u(x,y,p) \\ v(x,y,p) \end{bmatrix}. \tag{4.44}$$

Like the contravariant component, FV3 works with the normalized covariant wind (U, V) given by:

$$U(x,y,p) = \langle \boldsymbol{u}(x,y,p), \boldsymbol{e}_y(x,y,p) \rangle, \tag{4.45}$$

$$V(x,y,p) = \langle \boldsymbol{u}(x,y,p), \boldsymbol{e}_y(x,y,p) \rangle. \tag{4.46}$$

It is easy to see that:

$$U(x,y,p) = \frac{\mathfrak{U}(x,y,p)}{\|\partial_x \Phi_p(x,y)\|}, \quad V(x,y,p) = \frac{\mathfrak{V}(x,y,p)}{\|\partial_y \Phi_p(x,y)\|}. \tag{4.47}$$

Replacing Equation (4.38) in Equations (4.45) and (4.46) we obtain the relation between normalized covariant components in terms of the normalized contravariant terms:

$$\begin{bmatrix} U(x,y,p) \\ V(x,y,p) \end{bmatrix} = \begin{bmatrix} 1 & \langle \boldsymbol{e}_x, \boldsymbol{e}_y \rangle \\ \langle \boldsymbol{e}_x, \boldsymbol{e}_y \rangle & 1 \end{bmatrix} \begin{bmatrix} u(x,y,p) \\ v(x,y,p) \end{bmatrix}. \tag{4.48}$$

Recall that

$$\langle \boldsymbol{e}_x, \boldsymbol{e}_y \rangle (x,y,p) = \cos\alpha(x,y,p), \tag{4.49}$$

where $\alpha(x,y,p)$ is the angle between and $\boldsymbol{e}_x$, $\boldsymbol{e}_y$, which is the formula implemented in FV3 following Putman and Lin (2007). We may express the normalized contravariant components in terms of the normalized covariant terms inverting Equation (4.48). Notice that combining Equation (4.48) with Equations (4.41) one may get relations between the latitude-longitude components and the covariant components.

4.3 Edges treatment

4.3.1 Notation

We also utilize the notation $\mathcal{CS}_N = \mathbb{R}^{(N+\nu)\times(N+\nu)\times 6}$ to represent grid functions on the cubed-sphere at cell centers. We define the average values of a function q with the aid of the metric term $\sqrt{\mathfrak{g}}(x,y)$ at time t:

$$Q_{ijp}(t) = \frac{1}{|\Omega_{ij}|} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}} q(x,y,p,t) \sqrt{\mathfrak{g}}(x,y) \, dx \, dy. \tag{4.50}$$

Let us assume we have a function $q : \mathbb{S}_R^2 \times [0,T] \to \mathbb{R}$, and we have a $(\Delta x, \Delta y, \Delta t, \lambda)$-discretization of $\Omega \times [0,T]$. We introduce $q^n \in \mathcal{CS}^N$, which represents the grid function q evaluated at the discrete points. In other words, $q^n_{ijp} = q(x_i, y_j, p, t^n)$, where $i, j = -\nu + 1, \dots, N+\nu$, and $p = 1, \dots, 6$. Furthermore, we use the notations $q^n_{i+\frac{1}{2},j,p} = q(x_{i+\frac{1}{2}}, y_j, t^n)$ for $i = -\nu, \dots, N+\nu$ and $j = -\nu+1, \dots, N+\nu$ to represent q at C-grid points. Similarly, we use $q^n_{i,j+\frac{1}{2},p} = q(x_i, y_{j+\frac{1}{2}}, t^n)$ for $i = -\nu + 1, \dots, N + \nu$ and $j = -\nu, \dots, N + \nu$ to represent q at the D-grid points. When q does not depend on the time variable t, we can omit the index n. For a grid function Q we also use the notations:

$$\begin{aligned} Q_{\times,j,p} &:= (Q_{-\nu+1,j,p}, \dots, Q_{N+\nu,j,p}) \in \mathbb{R}^N_\nu, \\ Q_{i,\times,p} &:= (Q_{i,-\nu+1,p}, \dots, Q_{i,M+\nu,p}) \in \mathbb{R}^N_\nu. \end{aligned}$$

In this work, we shall always approximate the average values since our schemes are expected to be at most second-order, this approximation does not deteriorate the convergence order.

4.3.2 Ghost cells scalar field interpolation

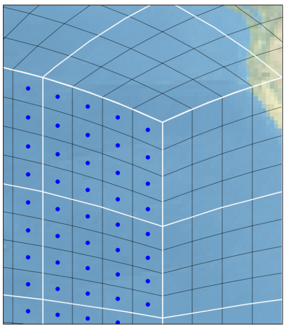
Let us consider a function $q : \mathbb{S}^2_R \to \mathbb{R}$ given at the A grid points denoted by q_{ijp}, where $i, j = 1 \dots, N$ and $p = 1, \dots, 6$. Our objective is to estimate these values at positions outside the range $1, \dots, N$, specifically at ghost cell positions.

To solve this problem, we will employ the strategy outlined in Zerroukat and Allen (2022), named duo grid by X. Chen (2021). As previously mentioned, the ghost cells in the local Cartesian systems are mapped onto the geodesics of adjacent panels, which enables us to use Lagrange interpolation to obtain the values of ghost cells.

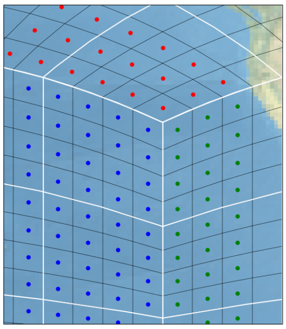
To illustrate this process in Panel 1, we depict the values of q_{ijp} in Figure 4.8. The blue circles represent the values in Panel 1 (Figure 4.8a), while the red and green circles represent the so called **kinked** values in the other panels (Figure 4.8b). Assuming a halo size of 3, we also indicate the target values at the ghost cell positions using blue squares (Figure 4.8c). It is worth noting that the dashed blue lines in Figure 4.8d illustrate how the ghost cell points lie on geodesics containing grid positions from adjacent panels. With the exception of the blue squares that lie on a cube corner (Figure 4.8d), all the ghost cell values can be obtained using 1D Lagrange interpolation, utilizing the surrounding red/green circles on the geodesic. This interpolation procedure can be performed for all panels. Subsequently, the blue squares that are on a cube corner can be interpolated using the values obtained in the first step of interpolation while preserving the order of accuracy of the interpolation, assuming it is fixed.

There are two ways of computing the Lagrange polynomials. The first one is based on the geodesic distances of the duo-grid line points. This approach was explored in X. Chen (2021) and Mouallem et al. (2023), and we are going to consider it here, calling it **dg1**. The second one is to use cube-based distances, where all duo-grid and kinked points are remapped to the plane using the inverse of the cube mapping. This approach has the advantage of having uniformly spaced data (the remapped kinked values) used in the duo-grid points interpolation. This approach is called **dg2**.

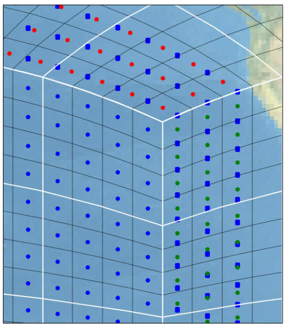
We are going to show a numerical example of this interpolation process using a halo region of size 3 and cubic polynomials. We shall consider the following trigonometric



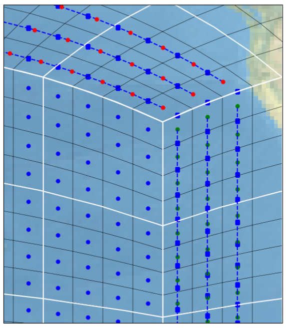
(a) *A-grid values (blue circles).*



(b) *Kinked A-grid values (red and green circles)*



(c) *A duo-grid points (blue squares)*



(d) *Interpolation lines (dashed blue lines)*

Figure 4.8: *Illustration of the A duo-grid interpolation for a scalar field.*

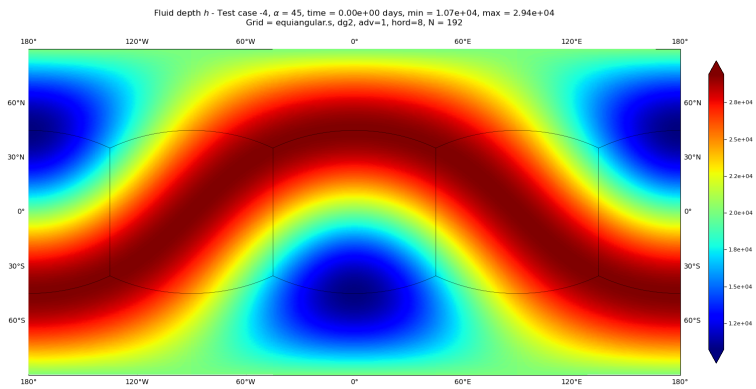


Figure 4.9: *Scalar field from Equation* (4.51).

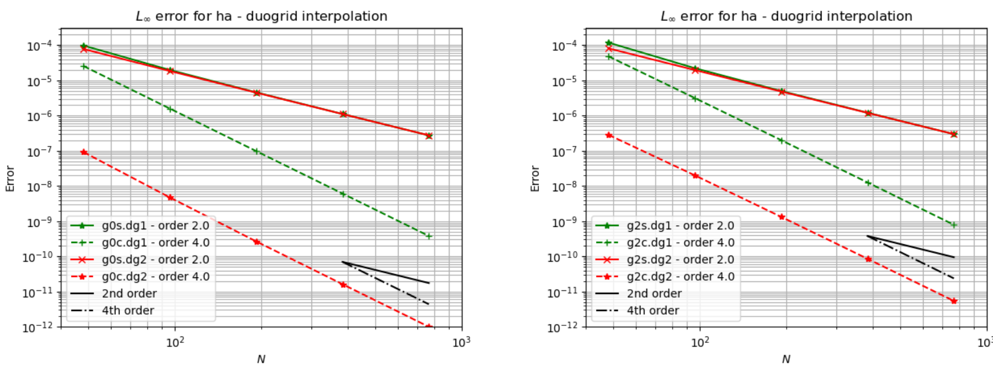
function from test case 2 in D. Williamson et al. (1992) in our tests:

$$q(\lambda, \phi) = h_0 - \frac{1}{g}\left(R\Omega u_0 + \frac{u_0^2}{2}\right)\left(-\cos(\lambda)\cos(\phi)\sin(\alpha) + \sin(\phi)\cos(\alpha)\right)^2, \tag{4.51}$$

where $h_0 = 3 \times 10^3$, $\alpha = \frac{\pi}{4}$, $u_0 = \frac{2\pi R}{12\text{days}}$, $g = 9.8$ is the gravity and $\Omega = 7.2921 \times 10^{-5}$ is the earth angular rotation speed. In Figure 4.9 we depict the graph of this field.

We compute the maximum errors for values of N of the form $N = 48 \times 2^k$, where k ranges from 0 to 4. We consider g0.s and g0.c, each one with dg1 and dg2, whose errors are depicted in Figure 4.10a. Additionally, we analyze g2.s and g2.c, each one with dg1 and dg2, and their errors are illustrated in Figure 4.10b.

From the dashed lines in Figure 4.10, we observe that both dg1 and dg2 achieve fourth-order accuracy when the midpoints use the cube formulation, with dg2 being much more accurate than dg1. Both dg1 and dg2 are very similar in this case. However, when spherical midpoints are employed, we observe a reduction in accuracy by two orders, as indicated by the solid lines. This discrepancy arises due to a second-order mismatch between cube and spherical midpoints, as discussed in Section 4.2.5. Finally, we observe that the g0 grid yields slightly better results than g2 for cube midpoints formulation.



(a) *Error for g0.* **(b)** *Error for g2.*

Figure 4.10: *Error for the duo-grid interpolation of the scalar field of* (4.51) *for the grid g0 (left) and g2 (right). Dashed lines use the cube midpoint formulation, while solid lines use the spherical midpoint formulation. Green lines represent dg1, and red lines represent dg2.*

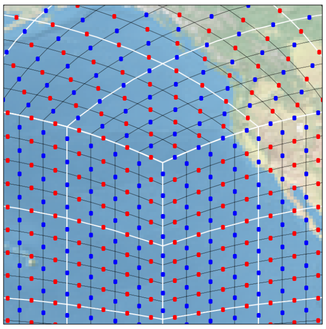
4.3.3 Ghost cells wind interpolation

Let us consider the following problem: Assume that we are given a tangent vector field of the sphere, denoted as $\boldsymbol{u} : \mathbb{S}_R^2 \to \mathbb{R}^3$. We also have its normalized covariant components at the C and D grid points, namely $u_{i+\frac{1}{2},j,p}$ for $i = 0, \ldots, N$ and $j = 1, \ldots, N$, as well as $v_{i,j+\frac{1}{2},p}$ for $i = 1, \ldots, N$ and $j = 0, \ldots, N$. This grid function is called C-grid wind (Figure 4.11a).

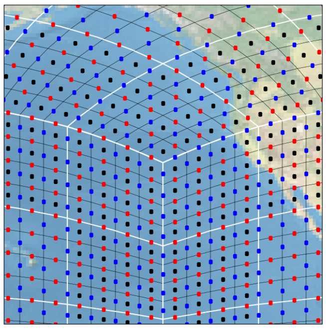
Our objective is to obtain the values

$$\begin{aligned} &u_{i+\frac{1}{2},j,p} \quad \text{for} \quad i = -1, \ldots, N+1, \qquad j = -\nu+1, \ldots, 0, \quad j = N, \ldots, N+\nu, \\ &v_{i,j+\frac{1}{2},p} \quad \text{for} \quad j = 0, \ldots, N, \qquad i = -\nu+1, \ldots, 0, \quad i = N, \ldots, N+\nu. \end{aligned}$$

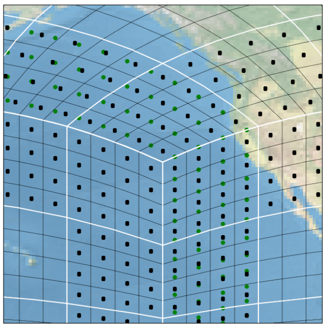
This problem arises when we apply the dimension splitting method on each panel of the cubed-sphere.



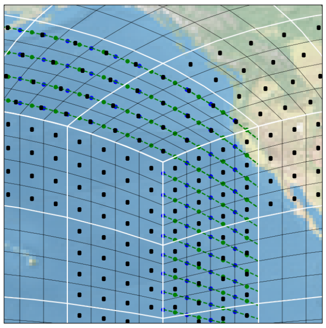
(a) *C-grid wind values (blue and red squares).*



(b) *A-grid wind (black circles)*



(c) *A-duo-grid points (green circles)*



(d) *C-duo-grid wind values (green circles)*

Figure 4.11: *Illustration of the C duo-grid interpolation for a C-grid wind.*

This problem can be solved by using the duo-grid interpolation process described earlier for a scalar field. To apply that interpolation process, we first need to interpolate the values of u and v from the edges to the A grid points required for the ghost cells interpolation (Figure 4.11b). Specifically, we need the values:

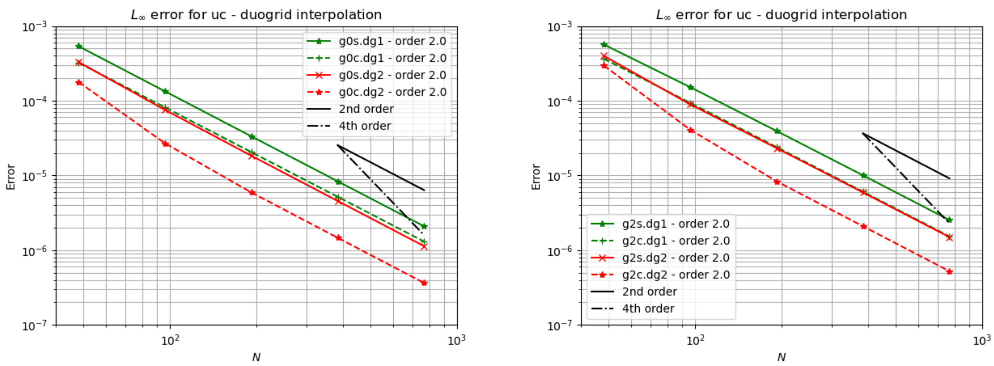
$$\begin{aligned}
u_{1+k,j,p}, v_{1+k,j,p} &\quad \text{for} \quad j = 1, \dots, N, \quad k = 0, \dots, \nu, \\
u_{N-k,j,p}, v_{N-k,j,p} &\quad \text{for} \quad j = 1, \dots, N, \quad k = 0, \dots, \nu, \\
u_{i,1+k,p}, v_{i,1+k,p} &\quad \text{for} \quad i = 1, \dots, N, \quad k = 0, \dots, \nu, \\
u_{i,N-k,p}, v_{i,N-k,p} &\quad \text{for} \quad i = 1, \dots, N, \quad k = 0, \dots, \nu.
\end{aligned}$$

We apply a simple linear interpolation to remap u and v to the A-grid points (Figure 4.11). However, we are going to use one extra layer of A-grid points since they will be needed for re-interpolation to the C and D grid points For instance, for 3 layers of ghost cells, we need 4 layers of A duo-grid points as shown in Figure 4.11b. Once these interpolated values are computed, we convert the covariant values u_{ijp}, v_{ijp} to their latitude-longitude components $(u_\lambda)_{ijp}, (v_\phi)_{ijp}$ using Equations (4.48) and (4.41). This conversion avoid any coordinate system discontinuity. Then, we can use the ghost cell centers interpolation procedure described before for the latitude-longitude components to recover the wind at the ghost cell centers using any polynomial degree (Figures 4.11c). Finally, we can use the values at the ghost cell centers to obtain the values at the ghost cell edges by employing a linear interpolation once again (Figures 4.11d). Subsequently, the covariant components can be obtained by using Equations (4.41) and (4.48).

We will consider the following rotated zonal field, as a numerical test, based on D. Williamson et al. (1992):

$$\begin{cases} u_\lambda(\lambda,\phi,t) = u_0(\cos(\phi)\cos(\alpha) + \sin(\phi)\cos(\lambda)\sin(\alpha)), \\ v_\phi(\lambda,\phi,t) = -u_0\sin(\lambda)\sin(\alpha). \end{cases} \tag{4.52}$$

Here, $u_0 = \frac{2\pi R}{12\text{days}}$ and $\alpha = \frac{\pi}{4}$. We will adopt the same grids and schemes as in Section 4.3.2. Next, we will compute the relative errors of the covariant components at the midpoint edges. The errors are presented in Figure 4.12, along with the convergence rate for different degrees employed in the ghost cell center interpolation. We emphasize that, since we utilize a linear interpolation to retrieve the wind at A-grid from the edges, as well as in the interpolation from the A duo-grid to C/D duo-grid points, the maximum attainable scheme order is 2. Indeed, from Figure 4.12, we observe that when employing a cubic polynomials in the duo-grid interpolation step, the final order achieved is 2. We can also observe again that the spherical midpoints yield larger errors than the cube midpoints formulation. Additionally, dg2 yields smaller errors, and overall, g0 performs slightly better than g2.



(a) *Error for g0.* **(b)** *Error for g2.*

Figure 4.12: *As Figure 4.10 but using the C-grid wind given by Equation (4.52).*

4.3.4 Edges reconstruction

Let us consider the following problem: given the values q_{ijp} we wish to find approximations of the function q at the C and D grid points denoted by

$$q^{L,x}_{ijp} \approx q_{i-\frac{1}{2},j,p}, \quad q^{R,x}_{ijp} \approx q_{i+\frac{1}{2},j,p}, \quad q^{L,y}_{ijp} \approx q_{i,j-\frac{1}{2},p}, \quad q^{R,y}_{ijp} \approx q_{i,j+\frac{1}{2},p}.$$

We can estimate the desired values by using the one-dimensional reconstruction schemes described in Section 2.4, performing PPM reconstruction independently in the x and y directions. It is worth noting that all the schemes discussed in those sections are expected to be second-order accurate due to the centroid point approximation.

There are some differences in the computation of the stencil near the cube edges. Unlike in the previous chapters, where periodic boundary conditions were assumed, the boundary conditions in this context are related to the adjacent panels. One way to address this issue is to use the duo-grid as discussed in Section 4.3.2 to compute the stencils. We are going to consider the dg2 method, since it yields better results overall.

Another approach, employed in Sadourny (1972), involves ignoring the discontinuity of the coordinate system and simply using the values of the cells in the adjacent panels as the ghost cell values. Additionally, an alternative method that avoids the use of ghost cells was developed by Putman and Lin (2007), which entails extrapolation at the cells surrounding the cube edge. We will refer to this scheme as **kinked** method. This scheme uses the following extrapolations:

$$\begin{aligned} q^{L,x}_{1,j,p} &= \frac{1}{2}\left(3Q_{1,j,p} - Q_{2,j,p}\right), \\ q^{R,x}_{N,j,p} &= \frac{1}{2}\left(3Q_{N,j,p} - Q_{N-1,j,p}\right), \end{aligned}$$

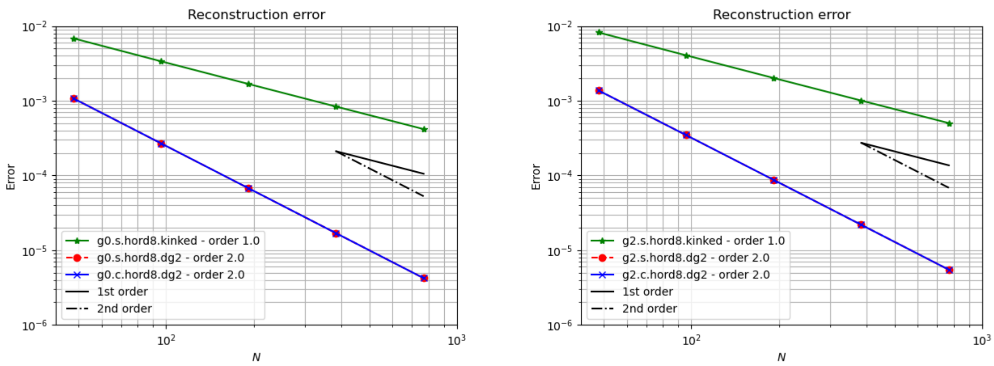
at the points that are located on the cube edges. The other edge values are estimated as:

$$\begin{aligned} q^{R,x}_{1,j,p} &= \frac{1}{14}\left(3Q_{1,j,p} + 11Q_{2,j,p} - 2(Q_{3,j,p} - Q_{1,j,p})\right), \\ q^{L,x}_{2,j,p} &= q^{R,x}_{1,j,p}, \\ q^{L,x}_{N,j,p} &= \frac{1}{14}\left(3Q_{N,j,p} + 11Q_{N-1,j,p} - 2(Q_{N-2,j,p} - Q_{N,j,p})\right), \\ q^{R,x}_{N-1,j,p} &= q^{L,x}_{N,j,p}, \end{aligned}$$

in the x direction. Similar formulas are used in the y direction. We are going to use the trigonometric function (Equation (4.51)) as before on the unit sphere to compare the schemes kinked and dg2. The scheme dg2 uses cubic polynomials. We introduce the relative errors:

$$
\begin{aligned}
e_{i-\frac{1}{2},j,p} &= (|q_{i-\frac{1}{2},j,p} - q^{L,x}_{ijp}|)/|q_{i-\frac{1}{2},j,p}|,\\
e_{i+\frac{1}{2},j,p} &= (|q_{i+\frac{1}{2},j,p} - q^{R,x}_{ijp}|)/|q_{i+\frac{1}{2},j,p}|,\\
e_{i,j-\frac{1}{2},p} &= (|q_{i,j-\frac{1}{2},p} - q^{L,y}_{ijp}|)/|q_{i,j-\frac{1}{2},p}|,\\
e_{i,j+\frac{1}{2},p} &= (|q_{i,j+\frac{1}{2},p} - q^{R,y}_{ijp}|)/|q_{i,j+\frac{1}{2},p}|,\\
e_{ijp} &= \max\{e_{i-\frac{1}{2},j,p}, e_{i+\frac{1}{2},j,p}, e_{i,j-\frac{1}{2},p}, e_{i,j+\frac{1}{2},p}\},\\
E &= \max\{e_{ijp}\}.
\end{aligned}
$$

We are going to compute E for different values of N as in the numerical experiments of Section 4.3.2. We consider the kinked scheme for the grids g0.s and g2.s, and the dg2 scheme for the grids g0.s, g0.c, g2.s, and g2.c. The reconstruction scheme employed is hord8. We depict the errors in Figure 4.13. We can observe that the error for g0 is only slightly better than the error for g2. Also, the errors for the cube midpoint formulation and the spherical midpoint formulation are essentially the same when using dg2.



(a) *Error for g0.* **(b)** *Error for g2.*

Figure 4.13: *Relative error for the PPM reconstruction using scalar field from Equation (4.51). g0 grid results are on the left, and g2 results are on the right. Green lines represent g2.s with the kinked method; red lines represent g2.s with dg2; blue lines represent g2.c with dg2. The reconstruction scheme is hord8.*

From Figure 4.13, we see that the dg2 method leads to second-order accuracy, while the kinked method leads to first-order accuracy. Since the kinked and dg2 affect hord8 only near to the cube edges, we expect that the error of the kinked method is larger only at the corners, leading to grid imprinting. Indeed, in Figure 4.14, we depict the logarithm of the error (on base 10 for plotting purposes) for the g0 grids. It becomes clear from Figure 4.14a that the kinked method leads to grid imprinting, while Figures 4.14b and 4.14c show that dg2 introduces less grid imprinting. Figure 4.15 shows similar results to Figure 4.14, but considering the g2 grid instead, from which we can draw similar conclusions. In general, dg2 is not sensitive to changing midpoints formulation. Additionally, the kinked method exhibits less grid imprinting in the g0 grid (Figure 4.14a) than in the g2 grid (Figure 4.15a).

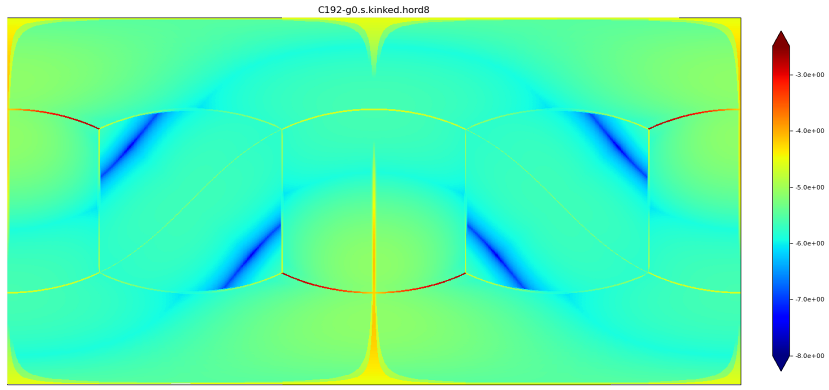
4.4 Concluding remarks

In this Chapter, we reviewed cubed-sphere mappings with a special focus on the equiangular and equi-edge mappings, which leads to a more uniform cubed-sphere grid, with the equiangular being the most uniform. The B-grid points are generated using the equiangular and equi-edge mappings; however, the A, C, and D grid points may be generated using the cubed-sphere mappings or using midpoints based on spherical midpoints of the B-grid points (Section 4.2.4).

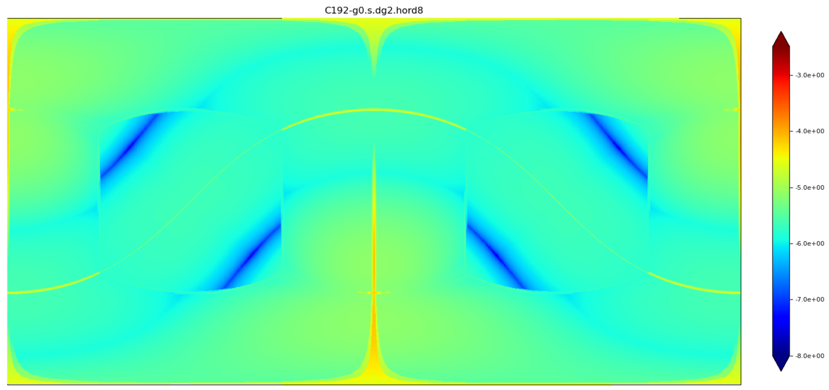
We observed that the equiangular cubed-sphere ghost cells, obtained by extending the gridlines, has a nice property: their edge and center ghost points are located on a common geodesic that contains the edge and center points of the adjacent panels. This property allows us to use 1D Lagrange interpolation to obtain the values of scalar and vector fields in the ghost cells. The equi-edge grid does not have this property, but we may mirror some points to make its ghost cells have the same property as in the equiangular grid. In fact, we demonstrated the accuracy of this interpolation on the duo-grid method through numerical examples in Sections 4.3.2 and 4.3.3. We explored two ways of computing the Lagrange polynomials: one based on geodesic distance and the other based on local coordinate distances. Overall, the method based on local coordinate distances showed to introduce smaller errors, especially when using the cube midpoints formulation instead of the spherical midpoints formulation.

Afterward, in Section 4.3.4, we investigated different methods for computing stencils near the cube edges. We considered the scheme based on 1D Lagrange interpolation and a scheme based on extrapolations from Putman and Lin (2007), which is currently implemented in FV3. Through numerical examples, we demonstrated that the reconstruction at cell edges using hord8 PPM scheme generates grid imprinting near the cubed edges when using extrapolations. The grid imprinting is greatly reduced when we apply the duo-grid Lagrange interpolation, showing that this scheme is much better for filling the ghost cells of the cube panels.

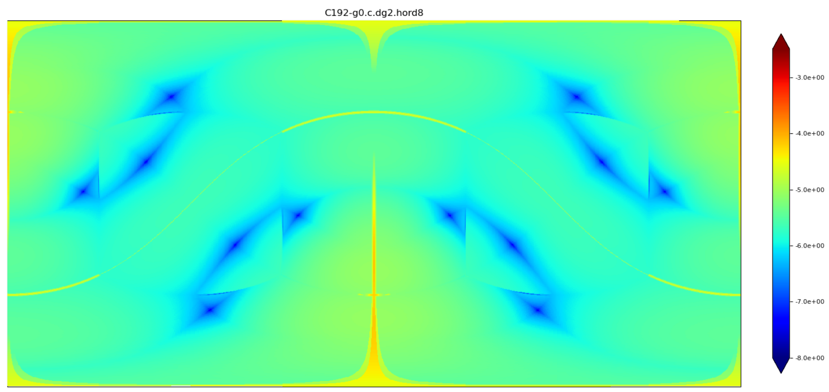
One major conclusion of this Chapter is that the cube and spherical midpoints formulation have a second-order difference. This impacts severely the duo-grid interpolation step, where we attain the expected orders of interpolation only for the cube midpoints formulation. However, in the reconstruction from A-grid to C/D grid points, both formulations have essentially the same errors. Hence, we expect that both midpoint formulations should lead to the very similar results.



(a) *g0.s with kinked.*

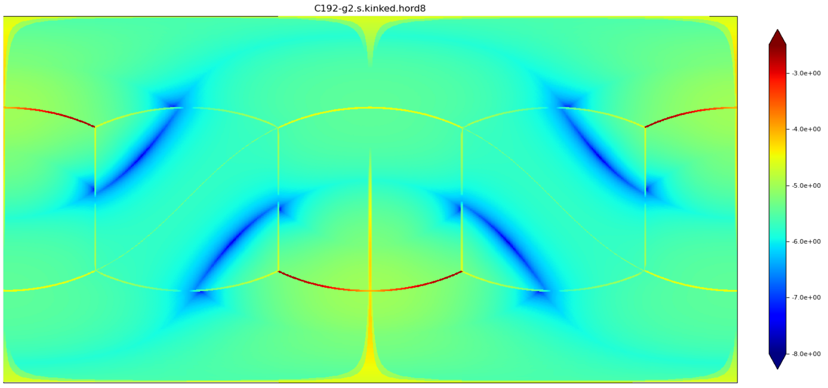


(b) *g0.s with dg2.*

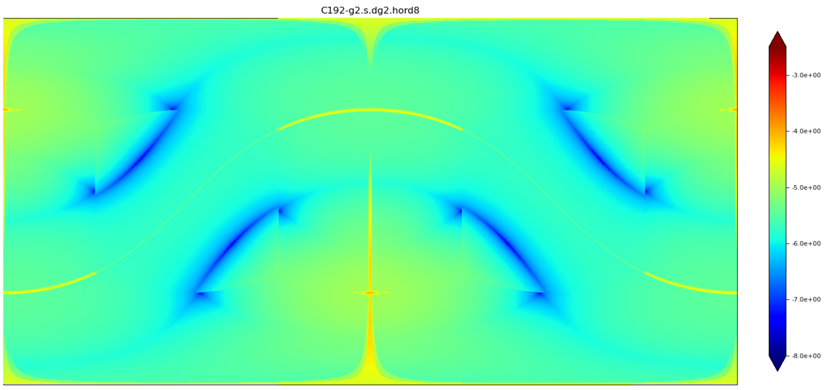


(c) *g0.c with dg2.*

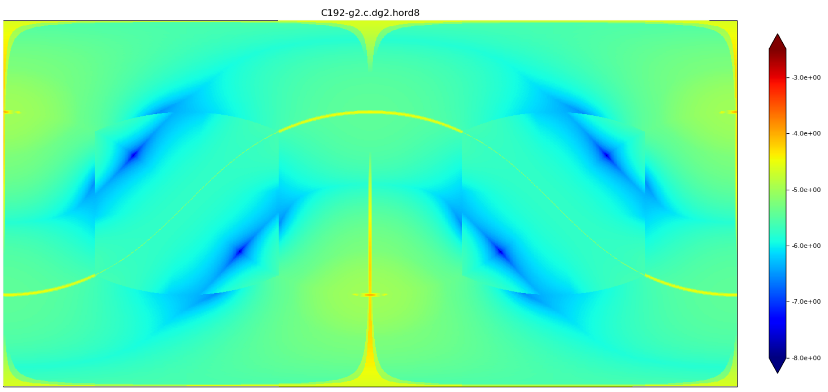
Figure 4.14: *The logarithm in base 10 of the errors of the C and D grid reconstruction from A-grid values of the scalar field, given by Equation (4.51), using the g0 grid with* $N = 192$.



(a) *g0.s with kinked.*



(b) *g0.s with dg2.*



(c) *g0.c with dg2.*

Figure 4.15: *As Figure 4.14 but using the g2 grid.*

Chapter 5

Cubed-sphere finite-volume methods

Now that we have described in Chapter 4 how we can obtain the ghost cell values of each panel on the cubed-sphere using Lagrange interpolation, we are ready to apply the dimension-splitting methods presented in Chapter 3 to solve the advection equation on the cubed-sphere. One significant difference is that we have the metric term, which is not present in the plane simulations. Additionally, when employing ghost cell layers using the duo-grid, the flux at the cube edges is computed twice, requiring the averaging of fluxes at the edges to ensure a unique value in order to achieve mass conservation.

This Chapter is organized as follows: Section 5.1 introduces the advection equation on the cubed-sphere. Section 5.2 presents its finite-volume discretization with a focus on the extension of dimension splitting (Section 5.3) as presented in Section 3.3. Numerical experiments are presented in Section 5.4, where we use dimension splitting to assess its accuracy in computing the divergence of a given vector field to check its numerical consistency, as well as to solve the advection equation. In particular, we explore different treatments for the cube edges. Section 5.5 presents the final thoughts.

5.1 Cubed-sphere advection equation in integral form

Given a tangent velocity field $\boldsymbol{u}$ on the sphere, we denote its contravariant components by $\mathfrak{u}$ and $\mathfrak{v}$. We shall use all the notations introduced in Section 4.3.1. The advection equation on panel the p of the cubed-sphere with initial condition q_0 is given by:

$$\begin{cases} \left[\partial_t q + \frac{1}{\sqrt{\mathfrak{g}}}\left(\partial_x(\mathfrak{u} q\sqrt{\mathfrak{g}}) + \partial_y(\mathfrak{v} q\sqrt{\mathfrak{g}})\right)\right](x,y,p,t) = 0, \\ q(x,y,p,0) = q_0(x,y,p), \end{cases} \tag{5.1}$$

$\forall(x,y) \in \Omega := [-\alpha,\alpha]^2$, $t \in [0,T]$. We denote by $\nabla \cdot (q\boldsymbol{u})$ the divergence operator:

$$\nabla \cdot (q\boldsymbol{u})(x,y,p,t) = \frac{1}{\sqrt{\mathfrak{g}}}[\partial_x(\mathfrak{u} q\sqrt{\mathfrak{g}}) + \partial_y(\mathfrak{v} q\sqrt{\mathfrak{g}})](x,y,p,t). \tag{5.2}$$

We recall that we say the $\boldsymbol{u}$ is **non-divergent** if $\nabla \cdot \boldsymbol{u} = 0$. We define the $\mathcal{CS}_N$ grid function δ^n as the exact divergence of $q\boldsymbol{u}$ at the cell centers, namely

$$\delta^n_{ijp} = \nabla \cdot (\boldsymbol{u}q)(x_i, y_j, p, t^n). \tag{5.3}$$

In this Chapter, it shall be useful to define the average value of $q\sqrt{\mathfrak{g}}$ on the 2D coordinates as:

$$\overline{(\sqrt{\mathfrak{g}}q)}_{ijp}(t) = \frac{1}{\Delta x \Delta y}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}}\int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}} q(x,y,p,t)\sqrt{\mathfrak{g}}(x,y)\,dx\,dy. \tag{5.4}$$

This average value simplifies the deduction of finite-volume method on the cubed-sphere instead of using the spherical average values (Equation (4.50)). Since the metric term does not depend on t, we may rewrite Equation (5.1) as

$$\left[\partial_t(q\sqrt{\mathfrak{g}}) + \partial_x(\mathfrak{u} q\sqrt{\mathfrak{g}}) + \partial_y(\mathfrak{v} q\sqrt{\mathfrak{g}})\right](x,y,p,t) = 0. \tag{5.5}$$

Therefore, as in Problem (3.1), the integral form of Equation (5.1) is stated in Problem (5.1).

Problem 5.1. *Given an initial condition q_0 and a velocity on the sphere $\boldsymbol{u}$, with contravariant components $(\mathfrak{u},\mathfrak{v})$ on the cubed-sphere coordinate system, we would like to find a weak solution q of the cubed-sphere advection equation in its integral form:*

$$\begin{aligned}\int_{x_1}^{x_2}\int_{y_1}^{y_2}(q\sqrt{\mathfrak{g}})(x,y,p,t)\,dx\,dy &= \int_{x_1}^{x_2}\int_{y_1}^{y_2}(q\sqrt{\mathfrak{g}})(x,y,p,t)\,dx\,dy \\ &\quad - \int_{t_1}^{t_2}\int_{y_1}^{y_2}\Big((\mathfrak{u} q\sqrt{\mathfrak{g}})(x_2,y,t) - (\mathfrak{u} q\sqrt{\mathfrak{g}})(x_1,y,t)\Big)\,dy\,dt \\ &\quad - \int_{t_1}^{t_2}\int_{x_1}^{x_2}\Big((\mathfrak{v} q\sqrt{\mathfrak{g}})(x,y_2,t) - (\mathfrak{v} q\sqrt{\mathfrak{g}})(x,y_1,t)\Big)\,dx\,dt.\end{aligned}$$

$\forall[x_1,x_2]\times[y_1,y_2]\times[t_1,t_2] \subset \Omega\times[0,T]$, *and* $q(x,y,p,0) = q_0(x,y,p)$.

Similarly to Section 3.1.2, Equation (5.1) and Problem (5.1) are equivalent when $q,\boldsymbol{u} \in C^1(\mathbb{S}^2_R)$. For Problem 5.1, the total mass in $\mathbb{S}^2_R$ is defined by:

$$M_{\mathbb{S}^2_R}(t) = \sum_{p=1}^{6}\int_{\Omega}(q\sqrt{\mathfrak{g}})(x,y,p,t)\,dx\,dy, \quad \forall t \in [0,T], \tag{5.6}$$

and is conserved within time:

$$M_{\mathbb{S}^2_R}(t) = M_{\mathbb{S}^2_R}(0), \quad \forall t \in [0,T]. \tag{5.7}$$

We define a discretized version of Problem (5.1) as Problem (5.2).

Problem 5.2. *Assume the framework of Problem 5.1 and consider a $(\Delta x, \Delta y, \Delta t, \lambda)$-discretization of $\Omega \times [0,T]$, with $\Delta x = \Delta y$. Since we are in the framework of Problem 5.1, it follows that:*

$$\overline{(\sqrt{\mathfrak{g}}q)}_{ijp}(t_{n+1}) = \overline{(\sqrt{\mathfrak{g}}q)}_{ijp}(t_n) - \lambda\delta_x\left(\frac{1}{\Delta t\Delta y}\int_{t^n}^{t^{n+1}}\int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}}(\mathfrak{u}q\sqrt{\mathfrak{g}})(x_i, y, p, t)\,dy\,dt\right) - \lambda\delta_y\left(\frac{1}{\Delta t\Delta x}\int_{t^n}^{t^{n+1}}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}}(\mathfrak{v}q\sqrt{\mathfrak{g}})(x, y_j, p, t)\,dx\,dt\right),$$

where

$$\overline{(\sqrt{\mathfrak{g}}q)}_{ijp}(t) = \frac{1}{\Delta x\Delta y}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}}\int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}}(q\sqrt{\mathfrak{g}})(x, y, p, t)\,dx\,dy. \tag{5.8}$$

Our problem now consists of finding the values $Q_{ijp}(t_n)$, $\forall i = 1, \dots, N$, $\forall j = 1, \dots, M$, $\forall n = 0, \dots, N_T - 1$, given the initial values $(\sqrt{\mathfrak{g}}q)_{ijp}(0)$, $\forall i = 1, \dots N$, $\forall j = 1, \dots, M$. In other words, we aim to find the average values of $(\sqrt{\mathfrak{g}}q)_{ijp}$ in each control volume Ω_{ijp} at the specified time instances.

It is important to note that no approximations have been made in Problems (5.1) and (5.2).

5.2 Finite-volume on the cubed-sphere approach

We are ready to introduce the finite-volume scheme on the cubed-sphere (CS-FV). A CS-FV scheme problem as follows in Problem 5.3. Before that, we consider the following approximation, which follows from the midpoint rule (Theorem A.5):

$$\overline{(\sqrt{\mathfrak{g}}q)}_{ijp}(t) = \sqrt{\mathfrak{g}_{ij}}q_{ijp}(t) + \mathcal{O}(\Delta x^2). \tag{5.9}$$

We use this approximation in Problemchp5-prob2 and we obtain the following CS-FV scheme:

Problem 5.3 (CS-FV scheme)**.** *Assume the framework defined in Problem 5.2. The finite-volume approach of Problem 5.1 consists of a finding a scheme of the form:*

$$q_{ijp}^{n+1} = q_{ijp}^n - \frac{\lambda}{\sqrt{\mathfrak{g}}_{ij}}\delta_i F_{ijp}^n - \frac{\lambda}{\sqrt{\mathfrak{g}}_{ij}}\delta_j G_{ijp}^n, \tag{5.10}$$

$$\forall i = 1, \dots, N, \quad \forall j = 1, \dots, M, \quad p = 1, \dots, 6, \quad \forall n = 0, \dots, N_T - 1,$$

where $\delta_i F_{ijp}^n = F_{i+\frac{1}{2},j,p}^n - F_{i-\frac{1}{2},j,p}^n$, $\delta_j G_{ijp}^n = G_{i,j+\frac{1}{2},p}^n - G_{i,j-\frac{1}{2},p}^n$ and $q^n \in CS_N$ is intended to be an approximation of $q(t_n) \in CS_N$ in some sense. We define $q_{ijp}^0 = q_{ijp}^0$.

The term $F_{i+\frac{1}{2},j,p}^n$ is known as numerical flux in the x direction and it approximates $\frac{1}{\Delta t\Delta y}\int_{t_n}^{t_{n+1}}\int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}}(\mathfrak{u}q\sqrt{\mathfrak{g}})(x_{i+\frac{1}{2}}, y, p, t)\,dy\,dt$, $\forall i = 0, 1, \dots, N$, and $G_{i,j+\frac{1}{2},p}^n$ is known as numerical flux in the y direction and it approximates $\frac{1}{\Delta t\Delta x}\int_{t_n}^{t_{n+1}}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}}(\mathfrak{v}q\sqrt{\mathfrak{g}})(x, y_{j+\frac{1}{2}}, p, t)\,dx\,dt$, $\forall j = 0, 1, \dots, M$, or, in other words, they estimate the time-averaged fluxes at the control volume

Ω_{ijp} boundaries.

Remark 5.1. *For Problem 5.3, we define the CFL number in the x and y direction by $\max\{|\mathfrak{u}^n_{i+\frac{1}{2},j,p}|\}\frac{\Delta t}{\Delta x}$ and $\max\{|\mathfrak{v}^n_{i,j+\frac{1}{2},p}|\}\frac{\Delta t}{\Delta y}$, respectively. The CFL number is maximum between these numbers and we say that the CFL condition is satisfied if the CFL number is less than one.*

As we mentioned in Problem 5.3, the initial condition may be assumed as q^0_{ijp} or $Q_{ijp}(0)$. We are going to assume q^0_{ijp} as initial data to avoid the computation of integrals. Furthermore, the errors will be calculated using the values q^n_{ijp} instead of $Q_{ijp}(t_n)$. As in Section 3.2 this approximation leads to a second-order error.

5.3 Dimension splitting

In this Section, we will utilize the dimension splitting method described in Section 3.3 to obtain a CS-FV scheme. To facilitate notation, we shall omit the index p whenever it may appear in this Section, as what is described here does not depend on p. Also, the ghost cell values are assumed to be filled using the duo-grid interpolation.

5.3.1 PPM and the metric term

Recall that the dimension splitting technique requires the numerical solution of advection in the x and y directions for separability. For instance, in the case of the advection equation on the cubed-sphere (Equation (5.5)), we need to solve the following equations in the x direction:

$$[\partial_t(\sqrt{\mathfrak{g}}q^x) + \partial_x(\mathfrak{u}\sqrt{\mathfrak{g}}q^x)](x, y_j, p, t), \tag{5.11}$$

for $j = -\nu+1, \dots, N+\nu$, at certain time levels t^n, $n = 1, \dots, N_T$ (Section 3.3.1). We are particularly interested in approximating $q^{x,n+1}_{ij}$ for $i = 1, \dots, N$, which represents the values of q^x at the cell centroids. This involves providing an approximation of the solution to Equation (5.5), denoted as q^n_{ij}, serving as initial data at time level n, specifically $q^{x,n}_{ij} = q^n_{ij}$.

Considering the midpoint approximation of the average value (Equation (5.9)), we approximate the solution of the desired problem using an general 1D FV-SL scheme as discussed in Section 2.2:

$$q^{x,n+1}_{ij} = q^n_{ij} - \frac{\Delta t}{\sqrt{\mathfrak{g}_{ij}}\Delta x}\left[F_{i+\frac{1}{2},j}\left(q^n;\tilde{c}^{x,n}\right) - F_{i-\frac{1}{2},j}\left(q^n;\tilde{c}^{x,n}\right)\right], \tag{5.12}$$

for $j = -\nu+1, \dots, N+\nu$ and $i = 1, \dots, N$, where

$$F_{i\pm\frac{1}{2},j} = \frac{1}{\Delta t}\int_{\tilde{x}^n_{i\pm\frac{1}{2},j}}^{x_{i\pm\frac{1}{2}}} (\widetilde{\sqrt{\mathfrak{g}}q})_j(x, t^n)\, dx, \tag{5.13}$$

$\tilde{x}^n_{i\pm\frac{1}{2},j}$ is an estimate of the departure point in x direction using the time-averaged CFL number $\tilde{c}^{x,n}_{i+\frac{1}{2},j}$ (Section 2.3), and $\widetilde{\sqrt{\mathfrak{g}}q}_j$ is a PPM reconstruction (or any other reconstruction) of $\sqrt{\mathfrak{g}}q$ (Section 2.4) in the x direction (j is fixed).

It is also possible to compute the PPM reconstruction in terms only of q, ignoring the metric term $\sqrt{\mathfrak{g}}$. In other words, we may assume that the metric is constant on each integration domain, which leads to the following first-order error:

$$\int_{\tilde{x}^n_{i\pm\frac{1}{2},j}}^{x_{i\pm\frac{1}{2}}} (\widetilde{\sqrt{\mathfrak{g}}q})(x,t^n)\,dx = \sqrt{\mathfrak{g}}_{i\pm\frac{1}{2},j} \int_{\tilde{x}^n_{i\pm\frac{1}{2},j}}^{x_{i\pm\frac{1}{2}}} \widetilde{q}(x,t^n)\,dx + \mathcal{O}(\Delta x). \tag{5.14}$$

In this case, the flux reads:

$$F_{i\pm\frac{1}{2},j} = \frac{\sqrt{\mathfrak{g}}_{i\pm\frac{1}{2},j}}{\Delta t} \int_{\tilde{x}^n_{i\pm\frac{1}{2},j}}^{x_{i\pm\frac{1}{2}}} \widetilde{q}(x,t^n)\,dx. \tag{5.15}$$

Then, in this case, we perform the PPM flux for the grid function q^n. When we compute the flux using Equation (5.13), we denote this by **mt0**; when using Equation (5.15), we denote this by **mt1**.

The works of Lin (2004) and Putman and Lin (2007) use the mt1 method, which is currently employed in FV3. This process, although it introduces a first-order error, significantly simplifies the elimination of the splitting error that arises when $q_{ij} = \overline{q}$, for a constant $\overline{q}$, and when the wind is divergence-free. This occurs because when we use mt1, we have

$$F_{i\pm\frac{1}{2},j} = \overline{q}\frac{\sqrt{\mathfrak{g}}_{i\pm\frac{1}{2},j}}{\Delta t}\delta_i c^{x,n}_{ij}, \tag{5.16}$$

assuming that the departure point is computed using the DP1 method for the departure point calculation (as discussed in Section 3.3.2). The property from Equation (5.16) does not occur for mt0.

5.3.2 The 2D scheme on each cube panel

For a CS-grid function $\psi \in \mathcal{CS}_N$ we introduce the following PPM flux in the x direction (recall Equation (2.66))

$$\mathfrak{F}^{PPM,x}_{i+\frac{1}{2},j}[\psi^n;\tilde{c}^{x,n}] = \begin{cases} \psi^n_{i-1,j} + (1-\tilde{c}^{x,n}_{i+\frac{1}{2}})(b^L_{ij} - \tilde{c}^{x,n}_{i+\frac{1}{2},j})(b^L_{ij}+b^R_{ij}), & \text{if} \quad \tilde{c}^{x,n}_{i+\frac{1}{2},j} > 0, \\ \psi^n_{ij} + (1+\tilde{c}^{x,n}_{i+\frac{1}{2},j})(b^L_{i+1,j} + \tilde{c}^{x,n}_{i+\frac{1}{2},j})(b^L_{i+1,j}+b^R_{i+1,j}), & \text{if} \quad \tilde{c}^{x,n}_{i+\frac{1}{2},j} \le 0. \end{cases} \tag{5.17}$$

for each $j = -\nu+1,\dots,N+\nu$ and $i = 1,\dots,N$, and where the PPM perturbation values b^L and b^R values are computed using hord0 (Section 2.4.1) or hord8 (Section 2.4.2)

Therefore, we may rewrite Equation (5.12) as

$$q^{x,n+1}_{ij} = q^n_{ij} + \mathbf{F}_{ij}[q^n,\tilde{c}^{x,n}], \tag{5.18}$$

for $i = 1,\dots,N$, $j = -\nu+1,\dots,M+\nu$, and where

$$\mathbf{F}_{ij}[q^n,\tilde{c}^{x,n}] = -\frac{1}{|\hat{\Omega}_{ij}|}\left(\mathcal{A}^x_{i+\frac{1}{2},j}\mathfrak{F}^x_{i+\frac{1}{2},j}[q^n_{\times,j},\tilde{c}^{x,n}] - \mathcal{A}^x_{i-\frac{1}{2},j}\mathfrak{F}^x_{i-\frac{1}{2},j}[q^n_{\times,j},\tilde{c}^{x,n}]\right),$$

recalling the term $|\hat{\Omega}_{ij}|$ from defined Equation (4.3), and following the discussion on the metric term, we have the coefficients

$$\mathcal{A}^x_{i+\frac{1}{2},j} = \begin{cases} \hat{\delta}x_{i+\frac{1}{2},j}\hat{\delta}y_{i+\frac{1}{2},j}\sin\alpha_{i+\frac{1}{2},j}\tilde{c}^{x,n}_{i+\frac{1}{2},j}, & \text{for mt0,} \\ \Delta x\Delta y\tilde{c}^{x,n}_{i+\frac{1}{2},j} & \text{for mt1,} \end{cases} \tag{5.19}$$

where we have made use of Equation (4.19) and Equation (4.2), and the PPM fluxes are

$$\mathfrak{F}^x_{i+\frac{1}{2},j}[q^n;\tilde{c}^{x,n}] = \begin{cases} \mathfrak{F}^{PPM,x}_{i+\frac{1}{2},j}[\sqrt{\mathfrak{g}}q^n;\tilde{c}^{x,n}] & \text{for mt0,} \\ \mathfrak{F}^{PPM,x}_{i+\frac{1}{2},j}[q^n;\tilde{c}^{x,n}] & \text{for mt1.} \end{cases} \tag{5.20}$$

Similarly, we may derive a scheme to solve Equation (5.5) in the y direction as

$$q^{y,n+1}_{ij} = q^n_{ij} + \mathbf{G}_{ij}[q^n, \tilde{c}^{x,n}], \tag{5.21}$$

for $i = -\nu+1,\dots,N+\nu$ $j = 1,\dots,N$.

$$\mathbf{G}_{ij}[q^n,\tilde{c}^{y,n}] = -\frac{1}{|\hat{\Omega}_{ij}|}\left(\mathcal{A}^y_{i,j+\frac{1}{2}}\mathfrak{F}^y_{i,j+\frac{1}{2}}[q^n_{i,\times},\tilde{c}^{y,n}] - \mathcal{A}^y_{i,j-\frac{1}{2}}\mathfrak{F}^y_{i,j-\frac{1}{2}}[q^n_{i,\times},\tilde{c}^{y,n}]\right),$$

and

$$\mathcal{A}^y_{i,j+\frac{1}{2}} = \begin{cases} \hat{\delta}x_{i,j+\frac{1}{2}}\hat{\delta}y_{i,j+\frac{1}{2}}\sin\alpha_{i,j+\frac{1}{2}}\tilde{c}^{y,n}_{i,j+\frac{1}{2}}, & \text{for mt0,} \\ \Delta x\Delta y\tilde{c}^{y,n}_{i,j+\frac{1}{2}}, & \text{for mt1,} \end{cases} \tag{5.22}$$

and the PPM fluxes are

$$\mathfrak{F}_{i,j+\frac{1}{2}}[q^n;\tilde{c}^{y,n}] = \begin{cases} \mathfrak{F}^{PPM,y}_{i,j+\frac{1}{2}}[\sqrt{\mathfrak{g}}q^n;\tilde{c}^{y,n}] & \text{for mt0,} \\ \mathfrak{F}^{PPM,y}_{i+\frac{1}{2},j}[q^n;\tilde{c}^{y,n}] & \text{for mt1.} \end{cases} \tag{5.23}$$

In FV3, the terms $\hat{\delta}x_{ij}$ and $\hat{\delta}y_{ij}$ and $|\hat{\Omega}_{ij}|$ (for integers or half integers i and j) are replaced by δx_{ij}, δy_{ij} and $|\hat{\Omega}_{ij}|$, which represent the geodesic distances and areas (Section 4.2.4).

Scheme	$\mathbf{f}_{ij}(q^n,\tilde{c}^{x,n})$	$\mathbf{g}_{ij}(q^n,\tilde{c}^{y,n})$
LT	$\mathbf{F}_{ij}(q^n,\tilde{c}^{x,n})$	$\mathbf{G}_{ij}(q^n,\tilde{c}^{y,n})$
PL	$-q^n_{ij} + \frac{q^n_{ij}+\mathbf{F}_{ij}(q^n,\tilde{c}^{x,n})}{1-\frac{1}{\|\hat{\Omega}_{ij}\|}\left(\mathcal{A}^x_{i+\frac{1}{2},j}-\mathcal{A}^x_{i-\frac{1}{2},j}\right)}$	$-q^n_{ij} + \frac{q^n_{ij}+\mathbf{G}_{ij}(q^n,\tilde{c}^{y,n})}{1-\frac{1}{\|\hat{\Omega}_{ij}\|}\left(\mathcal{A}^y_{i,j+\frac{1}{2}}-\mathcal{A}^y_{i,j-\frac{1}{2}}\right)}$

Table 5.1: *Expression of the inner advective operators considered in this work. LT stands for the average Lie-Trotter scheme, while PL stands for the scheme from Putman and Lin (2007).*

Following the same discussion of Sections 3.3.1 and 3.3.2, we may combine the operators

$\mathbf{F}$ and $\mathbf{G}$ and obtain the following scheme to update the cell centered values:

$$\begin{aligned} q^{n+1} = q^n &+ \frac{1}{2}\mathbf{F}[q^n, \tilde{c}^{x,n}] + \frac{1}{2}\mathbf{G}[q^n, \tilde{c}^{y,n}] \\ &+ \frac{1}{2}\mathbf{F}\left[q^n + \mathbf{g}[q^n, \tilde{c}^{y,n}], \tilde{c}^{x,n}\right] + \frac{1}{2}\mathbf{G}\left[q^n + \mathbf{f}[q^n, \tilde{c}^{x,n}], \tilde{c}^{y,n}\right], \end{aligned} \tag{5.24}$$

where the inner advection operators $\mathbf{f}$ and $\mathbf{g}$ are given in Table 5.1.

5.3.3 The upwind CFL number

When using the DP2 scheme (Section 2.3.2), we define the CFL number as (recall the wind formulation in Section 4.2.6):

$$c^{x,n}_{i+\frac{1}{2},j} = \mathfrak{u}^{x,n}_{i+\frac{1}{2},j,p}\frac{\Delta t}{\Delta x} = u^{x,n}_{i+\frac{1}{2},j}\frac{\Delta t}{\hat{\delta} x_{i+\frac{1}{2},j}}, \tag{5.25}$$

$$c^{y,n}_{i,j+\frac{1}{2}} = \mathfrak{v}^{y,n}_{i,j+\frac{1}{2},p}\frac{\Delta t}{\Delta y} = v^{y,n}_{i,j+\frac{1}{2}}\frac{\Delta t}{\hat{\delta} y_{i,j+\frac{1}{2}}}. \tag{5.26}$$

Therefore, the time-averaged CFL numbers may be computed using Equation (2.36). The current implementation of FV3 and the advection schemes from Lin (2004) and Putman and Lin (2007) uses the DP1 scheme (Section 2.3.1) and the following upwind CFL number introduced in Lin et al. (1994):

$$c^{x,n}_{i+\frac{1}{2},j} = \begin{cases} u^{x,n}_{i+\frac{1}{2},j}\frac{\Delta t}{\hat{\delta} x_{ij}}, & \text{if } u^{x,n}_{i+\frac{1}{2},j} < 0, \\ u^{x,n}_{i+\frac{1}{2},j}\frac{\Delta t}{\hat{\delta} x_{i+1,j}}, & \text{if } u^{x,n}_{i+\frac{1}{2},j} < 0, \end{cases} \tag{5.27}$$

$$c^{y,n}_{i,j+\frac{1}{2}} = \begin{cases} v^{x,n}_{i,j+\frac{1}{2}}\frac{\Delta t}{\hat{\delta} y_{ij}}, & \text{if } v^{y,n}_{i,j+\frac{1}{2}} < 0, \\ v^{x,n}_{i,j+\frac{1}{2}}\frac{\Delta t}{\hat{\delta} y_{i,j+1}}, & \text{if } v^{y,n}_{i,j+\frac{1}{2}} < 0, \end{cases} \tag{5.28}$$

We point out that Equation (5.25) to (5.26) and Equations (5.27) to (5.28) are equivalent when the metric term is constant and equal to one, as on the Cartesian grid on the plane. dditionally, we could use Equations (5.27) to (5.28) for the DP2 scheme, but we observed that the results obtained on advection simulations using Equations (5.25) to (5.26) are much better, while Equations (5.27) to (5.28) limit schemes with DP2 to first-order. Finally, we stress that for DP1, both formulations of the CFL number yield very similar results, but we use the upwind CFL for DP1 since this is what is used in FV3.

5.3.4 Flux at edges treatment

As in Section 3.2 we introduce the notion of discrete divergence, which allow us to check the consistency of CS-FV schemes.

Definition 5.1 (Discrete divergence)**.** *For Problem 5.3, we define the discrete divergence as* *a CS_N-grid function $\mathbb{D}^n(q^n, \mathfrak{u}^n, \mathfrak{v}^n)$ given by:*

$$\mathbb{D}^n_{ijp}(q^n, \mathfrak{u}^n, \mathfrak{v}^n) = \frac{1}{\Delta t \sqrt{\mathfrak{g}_{ij}}}\left(\frac{\delta_i F^n_{ijp}}{\Delta x} + \frac{\delta_j G^n_{ijp}}{\Delta y}\right), \quad i = 1, \ldots, N, \quad j = 1, \ldots, M. \tag{5.29}$$

With the aid of the discrete divergence, Equation (5.10) becomes:

$$q^{n+1} = q^n - \Delta t \mathbb{D}^n(q^n, \mathfrak{u}^n, \mathfrak{v}^n). \tag{5.30}$$

When using the dimension splitting technique, it follows from Equation (5.24) that the discrete divergence may be expressed as:

$$\mathbb{D}^n = \frac{-1}{\Delta t}\left(\frac{1}{2}\mathbf{F}[q^n, \tilde{c}^{x,n}] + \frac{1}{2}\mathbf{G}[q^n, \tilde{c}^{y,n}] + \frac{1}{2}\mathbf{F}\left[q^n + \mathbf{g}[q^n, \tilde{c}^{y,n}], \tilde{c}^{x,n}\right] + \frac{1}{2}\mathbf{G}\left[q^n + \mathbf{f}[q^n, \tilde{c}^{x,n}], \tilde{c}^{y,n}\right]\right). \tag{5.31}$$

For a CS-FV scheme the discrete total mass at the time-step n is given by:

$$M^n = \sum_{p=1}^{6} \sum_{i,j=1}^{N} Q^n_{ijp} |\hat{\Omega}_{ij}|.$$

It follows from Equation (5.30) that:

$$M^{n+1} = M^n - \sum_{p=1}^{6} \sum_{i,j=1}^{N} \mathbb{D}^n_{ijp} |\hat{\Omega}_{ij}|.$$

Hence, to ensure mass conservation, we must ensure that

$$\sum_{p=1}^{6} \sum_{i,j=1}^{N} \mathbb{D}^n_{ijp} |\hat{\Omega}_{ij}| = 0.$$

This property is discrete version of

$$\int_{\mathbb{S}^2_R} \nabla \cdot (\boldsymbol{u} q)\, dS = 0,$$

which follows from the divergence theorem and the fact of the sphere has no boundary, where dS is the surface measure of the sphere.

When computing the flux, if we ignore the discontinuity in the cubed sphere coordinate system and use values from adjacent panels (as in the kinked scheme from Chapter 4) to compute stencils, we can ensure mass conservation because the flux at points lying on the cube edge will be the same. However, if we consider ghost cell layers by extending the gridlines (as in the duo-grid scheme from Chapter 4), the flux is computed twice at points lying on the cube edge. Therefore, in this case, some modification is needed to ensure mass conservation (Figure 5.1). One common alternative used in the literature to handle the issue of values being defined twice at points on the cube edges is to simply average the values (as seen in works such as C. Chen and Xiao (2008), X. Chen (2021), Mouallem

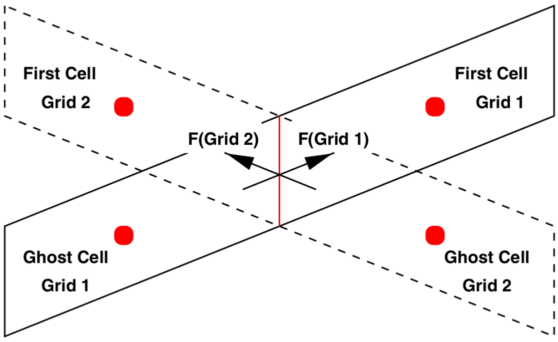


Figure 5.1: *Figure that illustrates the flux being computed twice on the cube edge, breaking the total mass conservation. Figure taken from Rossmanith (2006).*

et al. (2023), and Rossmanith (2006)). When we are using flux averaging, we shall use the label **mf1**. When no mass fixer is used, we employ the label **mf0**.

5.4 Numerical experiments

This Section is dedicated to present the numerical experiments for the advection equation on the sphere. In Table 5.2 we present the initial conditions (IC) and in Table 5.3 we present the velocity fields (VF) considered.

IC name	q_0
IC1	$\exp(b_0((X-X_0)^2+(Y-Y_0)^2+(Z-Z_0)^2))$
IC2	$\exp(b_0[(X-X_1)^2+(Y-Y_1)^2+(Z-Z_1)^2]+\exp(b_0[(X-X_2)^2+(Y-Y_2)^2+(Z-Z_2)^2])$

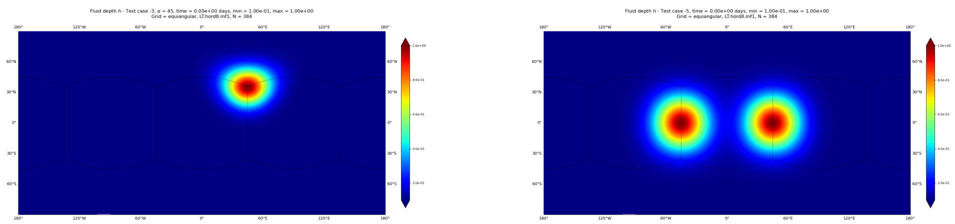
Table 5.2: *Initial conditions considered in the numerical experiments (Figure 5.2).*

VF name	$u_\lambda(\lambda,\phi,t)$	$v_\phi(\lambda,\phi,t)$	$\Delta t^{(0)}$
VF1	$u_0(\cos(\phi)\cos(\alpha)+\sin(\phi)\cos(\lambda)\sin(\alpha))$	$-u_0\sin(\lambda)\sin(\alpha)$	3600
VF2	$u_0\sin^2(\lambda_p)\sin(2\phi)\cos(\frac{\pi t}{T})+u_0\cos\phi$	$u_0\sin(2\lambda_p)\cos(\phi)\cos(\frac{\pi t}{T})$	1600
VF3	$-u_0\sin^2(\frac{\lambda+\pi}{2})\sin(2\phi)\cos^2(\phi)\cos(\frac{\pi t}{T})$,	$\frac{u_0}{2}\sin(\lambda+\pi)\cos^3(\phi)\cos(\frac{\pi t}{T})$	7200

Table 5.3: *Velocity fields considered in the numerical experiments and its initial time step $\Delta t^{(0)}$.*

In Table 5.2, we have $(X_0,Y_0,Z_0)=(\frac{1}{\sqrt{3}},\frac{1}{\sqrt{3}},\frac{1}{\sqrt{3}})$, while (X_1,Y_1,Z_1) and (X_2,Y_2,Z_2) are the Cartesian coordinates of the latitude-longitude points $(\lambda_1,\phi_1)=(-\frac{\pi}{4},0)$ and $(\lambda_2,\phi_2)=(\frac{\pi}{4},0)$, respectively. IC1 represents a Gaussian hill centered at a cube corner and we set $b_0=-10$. IC2 represents two Gaussian hills as suggested by Nair and Lauritzen (2010) and we set $b_0=-5$. The initial conditions are shown in Figure 5.2.

For the velocities provided in Table 5.3, we adopt the following parameter values: $\alpha=-\frac{45\pi}{180}$, $\lambda_p=\lambda-\frac{2\pi t}{T}$, $T=12$ days (12×86400 seconds) and R is the Earth radius. For VF1 and VF2, we use $u_0=\frac{2\pi R}{T}$, while for VF3, we use $u_0=\frac{\pi R}{2T}$. In this context, VF1 represents the non-divergent rotated zonal field introduced in D. Williamson et al. (1992). VF2 corresponds to the non-divergent deformational flow described in Nair and Lauritzen (2010), and VF3



(a) *IC1.* **(b)** *IC2.*

Figure 5.2: *Illustration of the initial conditions considered in this chapter (Table 5.2).*

represents the divergent flow also presented in Nair and Lauritzen (2010). For all velocity fields presented here, the initial condition is equal to the final solution after 12 days.

We are going to consider the schemes LT-DP2 and PL-DP1 since these schemes yield better results on planar simulations (Section 3.4). For a shorter notation, we shall denote LT-DP2 and PL-DP1 by **LT** and **PL** advection schemes. These schemes will be tested using hord0 and hord8 1D PPM schemes. As we mentioned in Section 5.3.1, the PL scheme needs the **mt1** metric term formulation for the 1D flux operators to eliminate the splitting error for a constant scalar field. For the LT scheme, we shall use the **mt0** metric term formulation because for this scheme, we do not have the constraint of eliminating the splitting error for a constant scalar field. Furthermore, this formulation makes the LT scheme much more accurate, while **mt1** for LT makes it first-order. We are also going to consider the simulations without mass fixer (**mf0**) and with flux averaging at cube edges (**mf1**) to investigate the impact of flux averaging on accuracy. Additionally, we are using the duo-grid to fill the ghost cell values. The reader may refer to Mouallem et al. (2023) for a comparison between results on the duo-grid versus the kinked grid. Both equi-edge (g0, Section 4.2.3) and equiangular grids (g2, Section 4.2.2) are going to be consider in this Section.

To compute the convergence, consider cubed-sphere grids with value of $N_k = 48 \times 2^k$, and $\Delta t^{(k)} = \frac{\Delta t^{(k)}}{2^k}$, $k = 0, \ldots, 4$, where the value of $\Delta t^{(0)}$ in Table 5.3 for each VF. The relative error in the maximum norm is computed as

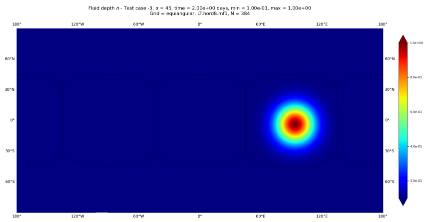
$$E_k = \frac{\max |q^{N_T} - q^0|}{\max |q^0|}, \tag{5.32}$$

and the convergence rate is defined as

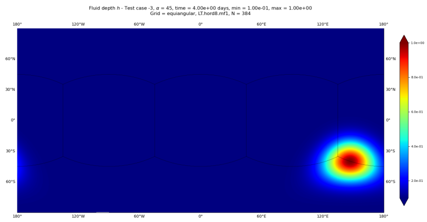
$$CR_k = \frac{\ln\left(\frac{E_k}{E_{k-1}}\right)}{\ln 2}, \quad \text{for} \quad k = 1, \ldots 4.$$

5.4.1 Advection of one Gaussian hill through the rotated zonal wind

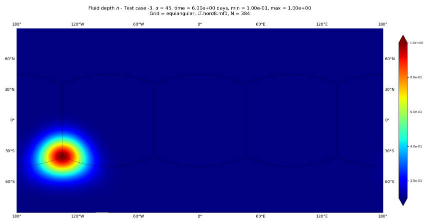
As a second test case, we consider the advection of the Gaussian hill given by IC1 using the rotated zonal wind VF1.



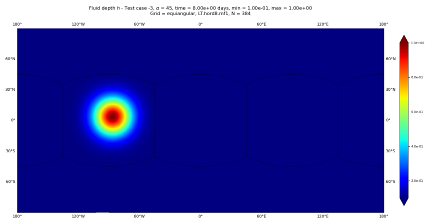
(a) $t = 2$ *days.*



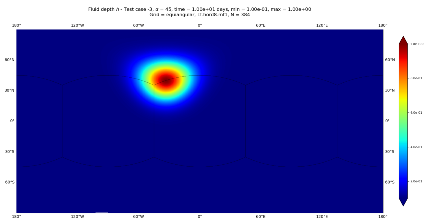
(b) $t = 4$ *days.*



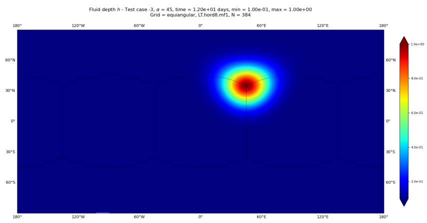
(c) $t = 6$ *days.*



(d) $t = 8$ *days.*

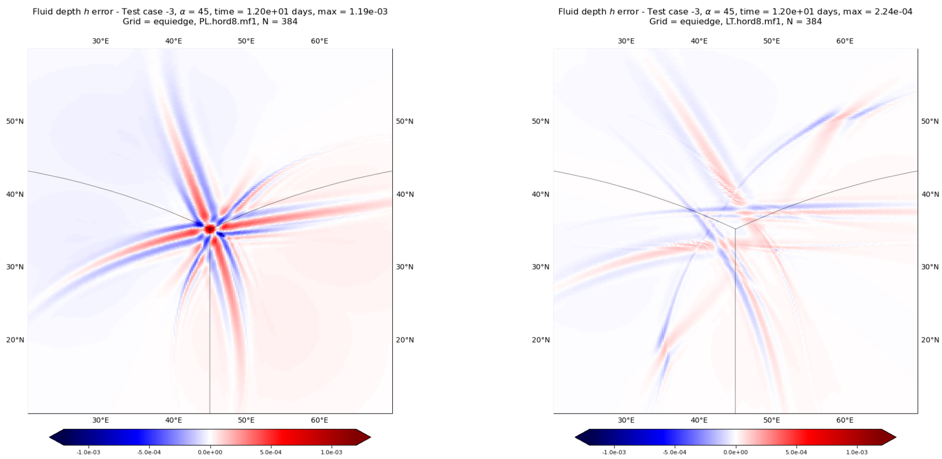


(e) $t = 10$ *days.*



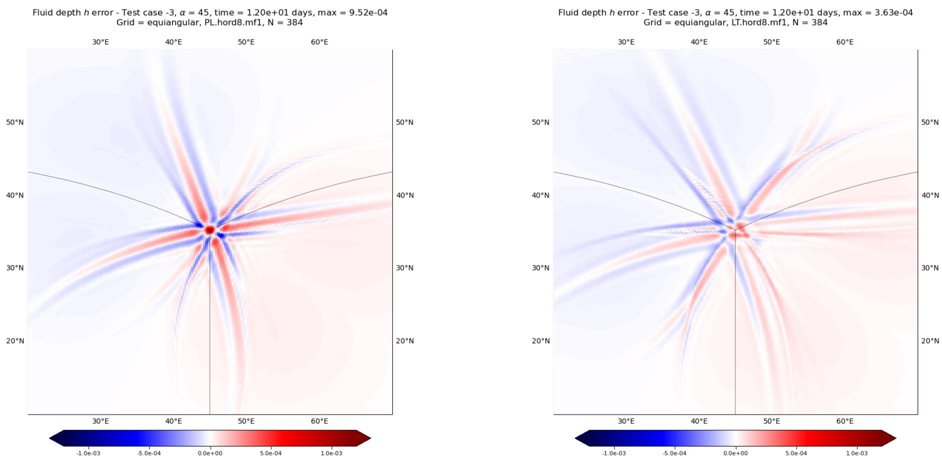
(f) $t = 12$ *days.*

Figure 5.3: *Advection experiment results using IC1 and VF1 from Table 5.2. These figures show the advected profile at 2 (5.3a), 4 (5.3b), 6 (5.3c), 8 (5.3d), 10 (5.3e), and 12 (5.3f) days. We are using the LT-hord8-mf1 scheme on the g2 grid with* $N = 384$.



(a) *PL scheme.* **(b)** *LT scheme.*

Figure 5.4: *Advection experiment errors at a cube corner using IC1 and VF1 from Table 5.2 after 12 days, using hord8 with PL (left) and LT schemes (right) on the g0 grid with* $N = 384$.



(a) *PL scheme.* **(b)** *LT scheme.*

Figure 5.5: *As Figure 5.4 but using the g2 grid.*

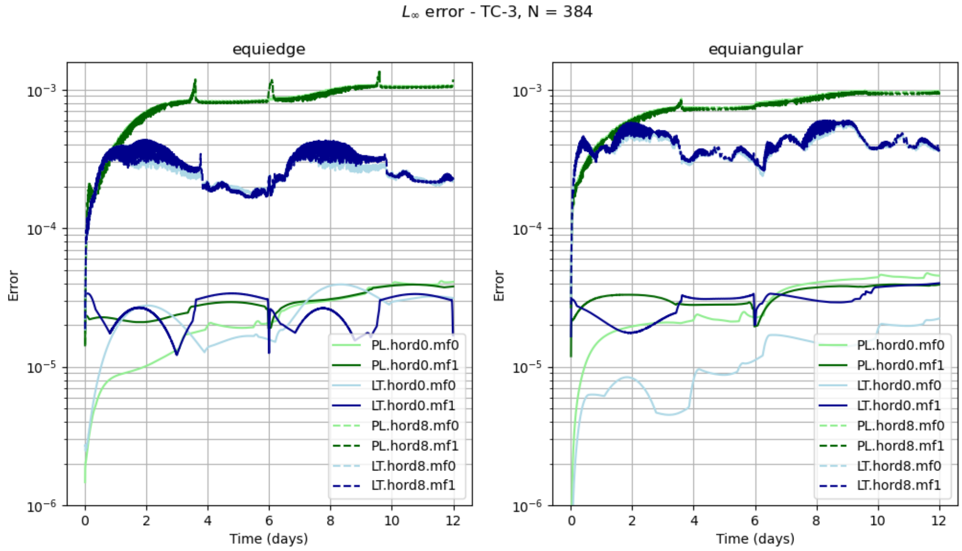


Figure 5.6: *L_∞ error evolution for IC1 and VF1 from Table 5.2 on the g0 (left) and g2 (right) grids for 12 days. Blue lines indicate the use of the LT scheme, while green lines represent the PL scheme. Solid lines represent the results with the hord8 scheme, whereas dashed lines represent the results with hord8. Light colors show the result without mass fixer (mf0), whereas dark colors show the results with flux averaging (mf1).*

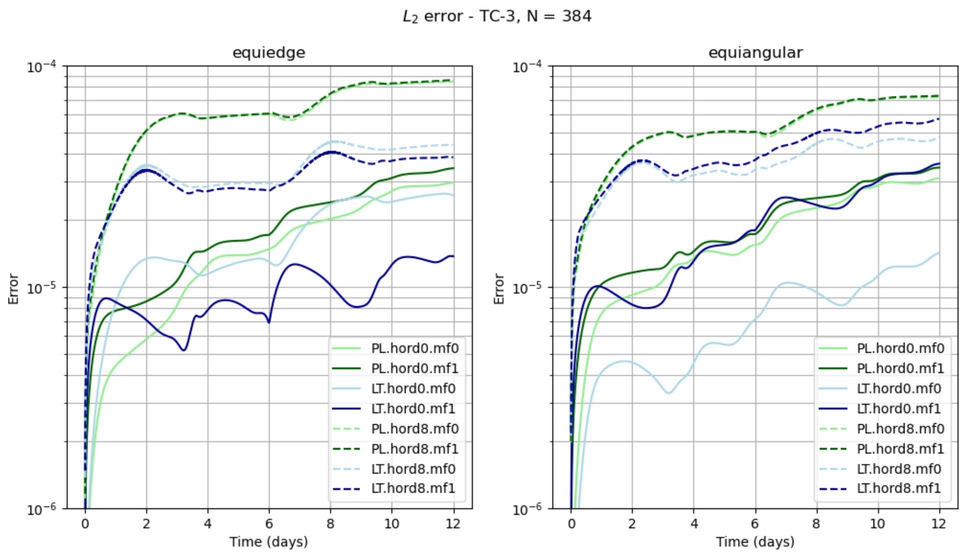


Figure 5.7: *As Figure 5.6 but using the L_2 error.*

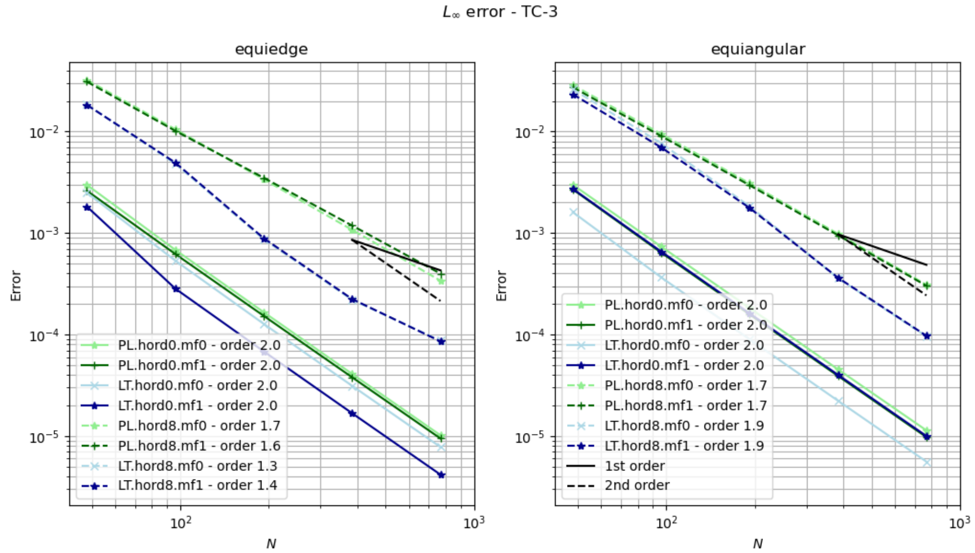


Figure 5.8: *L_∞ error convergence for IC1 and VF1 from Table 5.2 on the g0 (left) and g2 (right) grids after 12 days. Blue lines indicate the use of the LT scheme, while green lines represent the PL scheme. Solid lines represent the results with the hord8 scheme, whereas dashed lines represent the results with hord8. Light colors show the result without mass fixer (mf0), whereas dark colors show the results with flux averaging (mf1).*

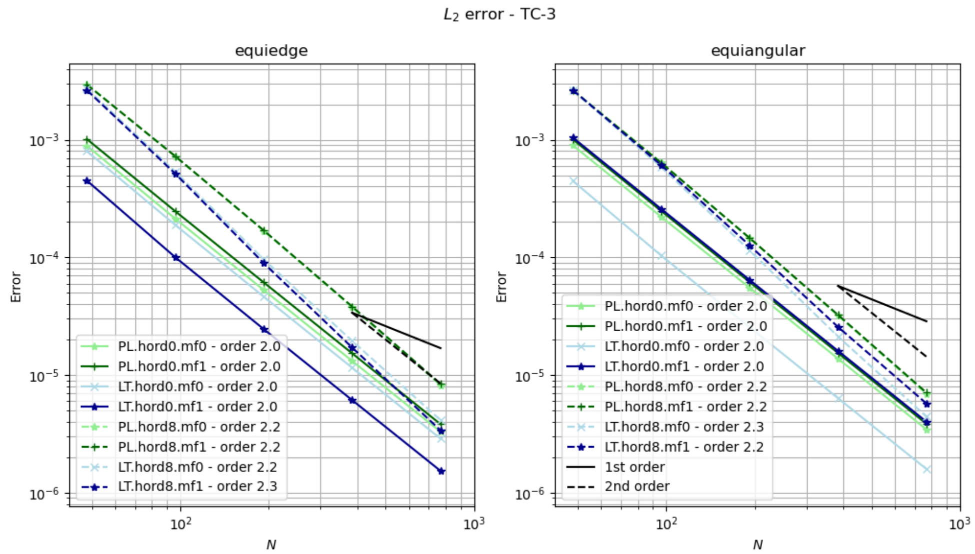
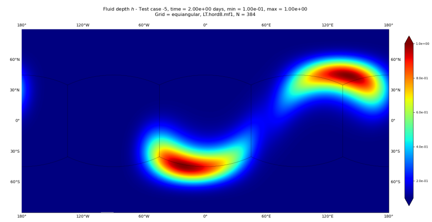


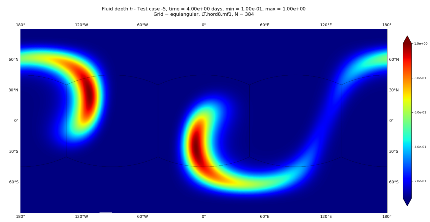
Figure 5.9: *As Figure 5.8 but considering the L_2 norm.*

5.4.2 Non-divergent deformational flow

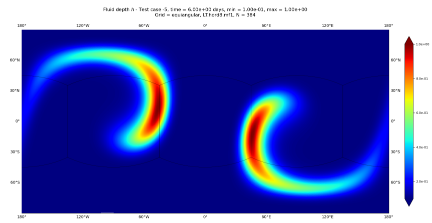
As the fourth test case, we considered the non-divergent vector field VF2 velocity from table 5.3. We used the initial condition IC2 from table 5.2, as suggested by Nair and Lauritzen (2010), which also shows how the solution evolves with time.



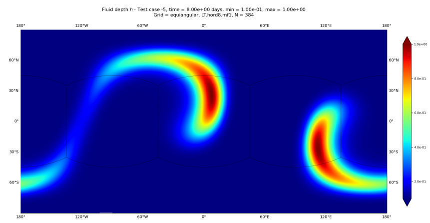
(a) $t = 2$ *days.*



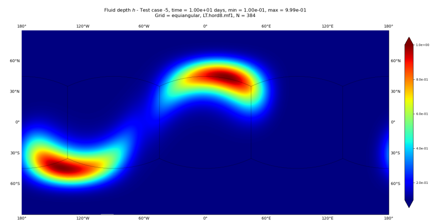
(b) $t = 4$ *days.*



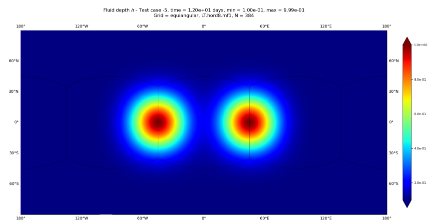
(c) $t = 6$ *days.*



(d) $t = 8$ *days.*

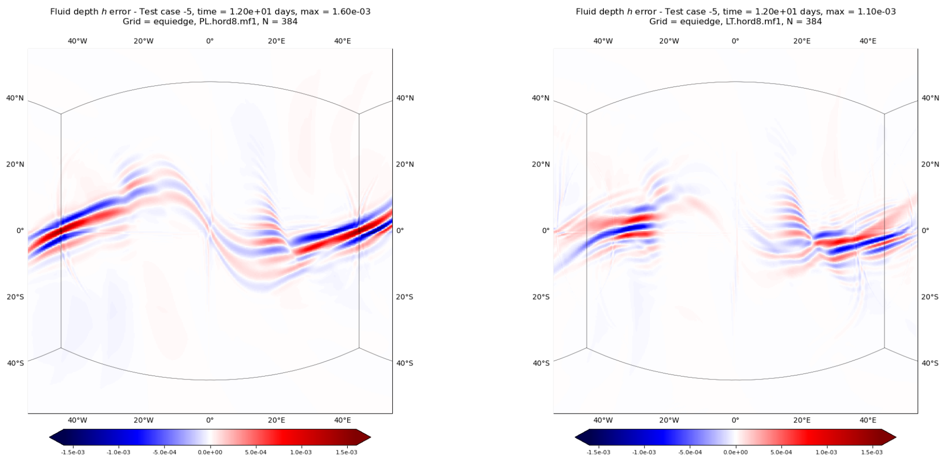


(e) $t = 10$ *days.*



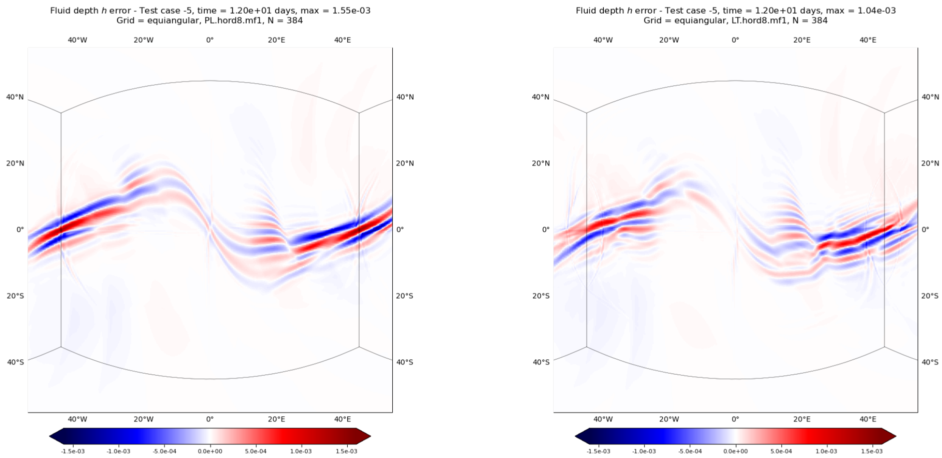
(f) $t = 12$ *days.*

Figure 5.10: *Similar to Figure 5.3 but using IC2 and VF2 from Table 5.2.*



(a) *PL scheme.* **(b)** *LT scheme.*

Figure 5.11: *Advection experiment errors using IC2 and VF2 from Table 5.2 after 12 days, using hord8 with PL (left) and LT schemes (right) on the g0 grid with* $N = 384$.



(a) *PL scheme.* **(b)** *LT scheme.*

Figure 5.12: *As Figure 5.11 but using the g2 grid.*

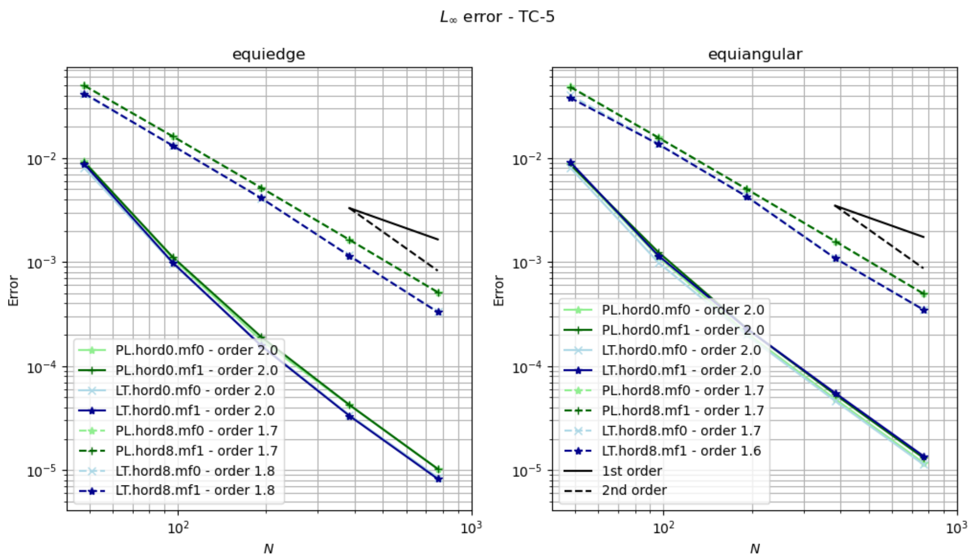


Figure 5.13: *As Figure 5.8 but using IC2 and VF2 from Table 5.2.*

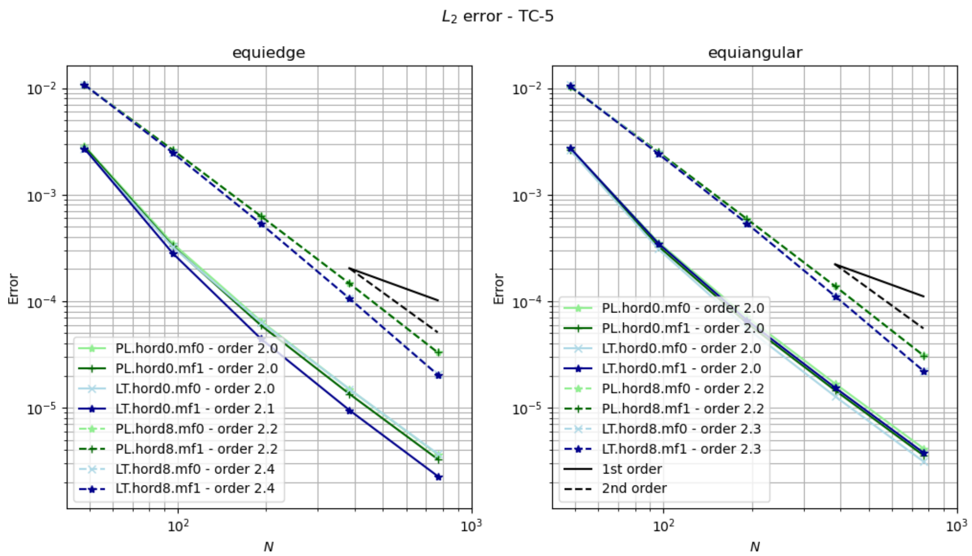
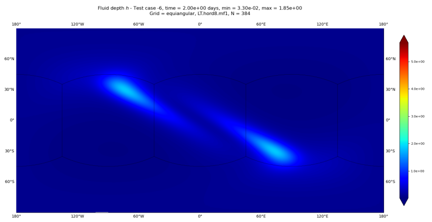


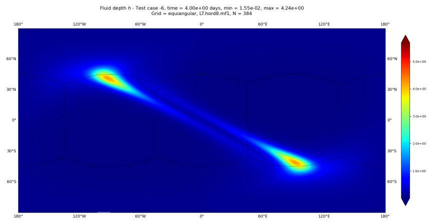
Figure 5.14: *As Figure 5.13 but considering the L_2 norm.*

5.4.3 Divergent deformational flow

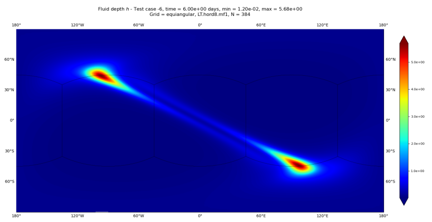
The fifth and last test case consider the divergent wind VF3 from table 5.3 along with the initial condition IC2 from table 5.2. This test is also suggested by Nair and Lauritzen (2010) and the their paper shows how the solution evolves over time.



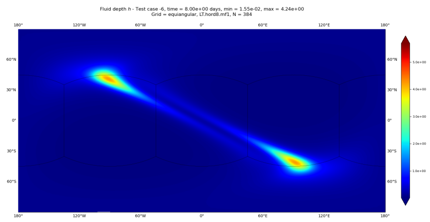
(a) *$t = 2$ days.*



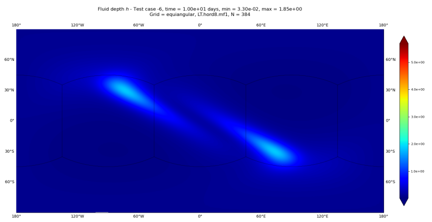
(b) *$t = 4$ days.*



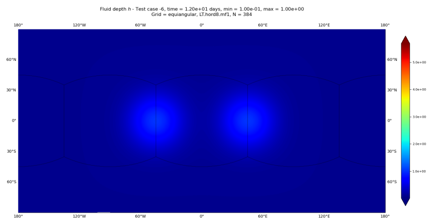
(c) *$t = 6$ days.*



(d) *$t = 8$ days.*

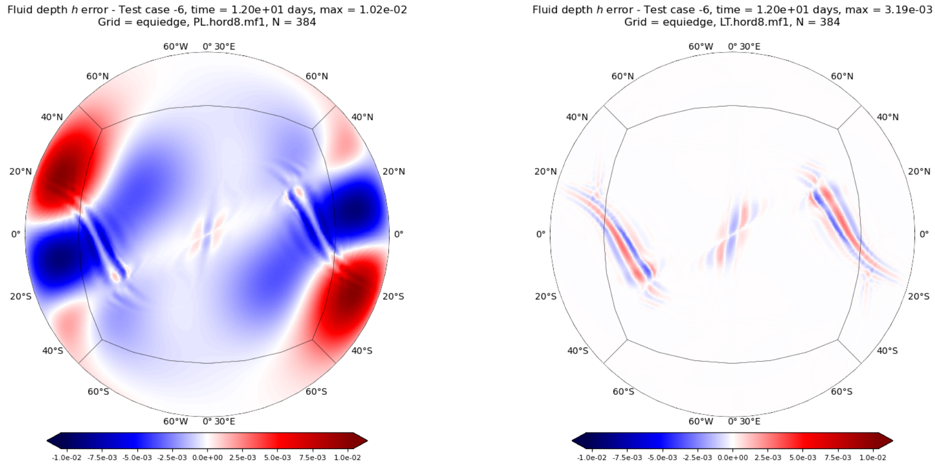


(e) *$t = 10$ days.*



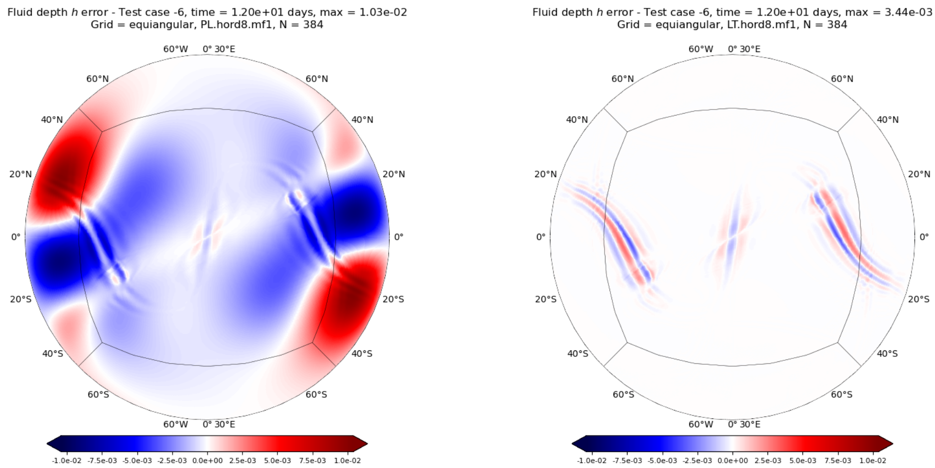
(f) *$t = 12$ days.*

Figure 5.15: *Similar to Figure 5.3 but using IC2 and VF3 from Table 5.2.*



(a) *PL scheme.* **(b)** *LT scheme.*

Figure 5.16: *Advection experiment errors using IC2 and VF3 from Table 5.2 after 12 days, using hord8 with PL (left) and LT schemes (right) on the g0 grid with* $N = 384$.



(a) *PL scheme.* **(b)** *LT scheme.*

Figure 5.17: *As Figure 5.16 but using the g2 grid.*

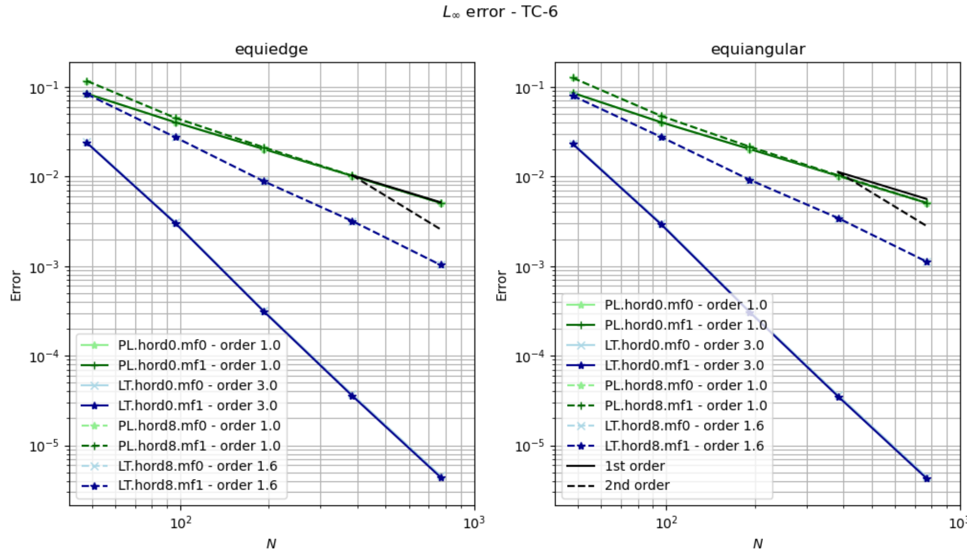


Figure 5.18: *As Figure 5.8 but using IC2 and VF3 from Table 5.2.*

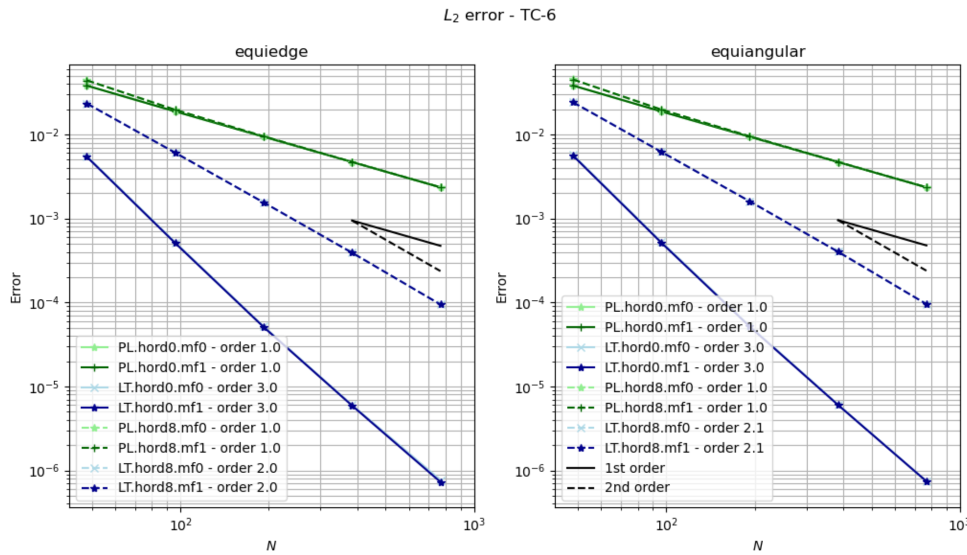


Figure 5.19: *As Figure 5.18 but considering the L_2 norm.*

5.5 Concluding remarks

Chapter 6

Cubed-sphere finite-volume shallow-water model

Chapter 7

Conclusions

Appendix A

Numerical Analysis

A.1 Lagrange interpolation

Given real numbers, called nodes, $x_0 < x_1 < \ldots < x_m$, we define the k-th Lagrange polynomial by

$$L_k(x) = \prod_{j=0, j\neq k}^{m} \frac{x - x_j}{x_k - x_j}.$$

They satisfy $L_k(x_j) = \delta_{kj}$, where δ_{kj} is the Kronecker delta. Given a function f defined at the nodes x_j, its interpolating polynomial of degree m is given by:

$$P_m(x) = \sum_{k=0}^{m} f(x_k)L_k(x).$$

Indeed, this polynomial interpolates f since $P_m(x_j) = f(x_j)$. It is well known that P_m always exists and is unique. Besides that, we have the following error formula for Lagrange interpolation.

Theorem A.1. *Let $f \in C^{m+1}(\mathbb{R})$. Then, then there is ξ in the smallest interval containing $x_0, \ldots, x_m, x$ such that:*

$$f(x) - P_m(x) = \omega(x)\frac{f^{(m+1)}(\xi)}{(m+1)!}, \tag{A.1}$$

where $\omega(x) = (x - x_0)(x - x_1)\ldots(x - x_m)$.

Proof. See Stoer and Bulirsch (2002, Theorem 2.1.4.1. on p. 49). □

A.2 Numerical integration

The following mean value theorem for integrals is a very useful tool when working with numerical integration errors.

Theorem A.2 (Mean value theorem for integrals)**.** *If $f \in C([a, b])$, and g is a integrable* *function in* $[a,b]$ *whose sign does not change in* $[a,b]$*, then there exists* $c \in]a,b[$ *such that*

$$\int_a^b f(x)g(x)\,dx = f(c)\int_a^b g(x)\,dx.$$

Proof. See Courant and John (1999, p. 143). □

Theorem A.3 (Leibniz integral rule)**.** *If* $f \in C^1$*, then*

$$\frac{d}{ds}\int_{s_0}^{s} f(s,\theta)\,d\theta = f\big(s,s\big) + \int_{s_0}^{s} \partial_s f(s,\theta)\,d\theta.$$

Proof. Let us define

$$F(s) = \int_{s_0}^{s} f(s,\theta)\,d\theta,$$

and take a sequence h_n of real numbers such that $h_n \overset{n\to\infty}{\longrightarrow} 0$. Then

$$\frac{F(s+h_n)-F(s)}{h_n} = \frac{1}{h_n}\int_{s_0}^{s+h_n} f(s+h_n,\theta)\,d\theta - \frac{1}{h_n}\int_{s_0}^{s} f(s,\theta)\,d\theta \tag{A.2}$$

$$= \frac{1}{h}\left(\int_{s}^{s+h} f(s+h_n,\theta)\,d\theta + \int_{s_0}^{s} f(s+h_n,\theta)\,d\theta - \int_{s_0}^{s} f(s,\theta)\,d\theta\right). \tag{A.3}$$

It follows from Theorem A.2 (with $g = 1$) that there exists θ_n between s and $s+h$ such that:

$$\frac{1}{h_n}\int_{s_0}^{s+h_n} f(s+h_n,\theta)\,d\theta = f(s+h_n,\theta_n) \overset{n\to\infty}{\longrightarrow} f(s,s), \tag{A.4}$$

since $\theta_n \overset{n\to\infty}{\longrightarrow} s$. From the mean value theorem, there exists s_n between s and $s+h_n$ such that:

$$\int_{s_0}^{s}\left(\frac{f(s+h_n,\theta)-f(s,\theta)}{h}\right)d\theta = \int_{s_0}^{s} \partial_s f(s_n,\theta)\,d\theta \overset{n\to\infty}{\longrightarrow} \int_{s_0}^{s} \partial_s f(s,\theta)\,d\theta, \tag{A.5}$$

where the last limit can be jusitified using the Lebesgue's dominated convergence theorem (see Folland (1999, p. 54)) . Using Equations (A.4) and (A.5) in Equation (A.3), we get the desired identity since the sequence h_n is any sequence that converges to 0. □

A.2.1 Midpoint rule

When considering finite-volume schemes, it is useful to compare the average value on a control volume of a function with its value at the control volume centroid. In the following theorems, for the one and two dimensional cases, respectively, we show that the value of a function at the centroid of a control volume given a second-order approximation to its average value on the control volume.

Theorem A.4. *If $f \in C^2([x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}])$, then*

$$\frac{1}{\Delta x}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} f(x)\,dx - f(x_i) = C_1 \Delta x^2, \tag{A.6}$$

where C_1 is a constant that depends only on f, and $x_i = \frac{x_{i+\frac{1}{2}}+x_{i-\frac{1}{2}}}{2}$, $\Delta x = x_{i+\frac{1}{2}} - x_{i-\frac{1}{2}}$.

Proof. From Taylor's expansion, it follows that, for $x \in [x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]$, we have:

$$f(x) = f(x_i) + f(x_i)(x - x_i) + f''(\xi)\frac{(x-x_i)^2}{2}, \tag{A.7}$$

for some ξ between x and x_i. Therefore:

$$\begin{aligned}\frac{1}{\Delta x}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} f(x)\,dx - f(x_i) &= \frac{1}{\Delta x}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \left(f'(x_i)(x-x_i) + f''(\xi)\frac{(x-x_i)^2}{2}\right) dx \\ &= \frac{1}{\Delta x}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} f''(\xi)\frac{(x-x_i)^2}{2}\,dx.\end{aligned}$$

Using the mean value theorem for integrals (see Theorem A.2), we have:

$$\frac{1}{\Delta x}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} f(x)\,dx - f(x_i) = f''(\eta_i)\frac{1}{\Delta x}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \frac{(x-x_i)^2}{2}\,dx = f''(\eta_i)\frac{\Delta x^2}{24}$$

for some $\eta_i \in [x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]$, from which the proposition follows with

$$C_1 = \frac{1}{24} f''(\eta_i). \tag{A.8}$$

□

Theorem A.5. *If $f \in C^2([x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}] \times [y_{j-\frac{1}{2}}, y_{j+\frac{1}{2}}])$, then*

$$\frac{1}{\Delta x \Delta y}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}}\int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}} f(x,y)\,dx\,dy - f(x_i, y_j) = C\Delta x^2, \tag{A.9}$$

where C_1 is a constants that depends only on f, where we assume $x_i = \frac{x_{i+\frac{1}{2}}+x_{i-\frac{1}{2}}}{2}$, $y_i = \frac{y_{j+\frac{1}{2}}+y_{j-\frac{1}{2}}}{2}$, $\Delta x = x_{i+\frac{1}{2}} - x_{i-\frac{1}{2}}$, $\Delta y = y_{j+\frac{1}{2}} - y_{j-\frac{1}{2}}$ and $\Delta x = \Delta y$.

Proof. Applying Theorem A.4 in the y direction, we have

$$\int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}} f(x,y)\,dy = \Delta y f(x, y_j) + \frac{\Delta y^3}{24}\partial_y^2 f(x, \eta_j),$$

for $\eta_j \in [y_{j-\frac{1}{2}}, y_{j+\frac{1}{2}}]$. Hence:

$$\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}} f(x,y)\,dx\,dy = \Delta y \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} f(x,y_j)\,dx + \frac{\Delta y^3}{24} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \partial_y^2 f(x,\eta_j)\,dx.$$

Applying Theorem A.4 in the x direction for $y = y_j$, we get

$$\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} f(x,y_j)\,dx = \Delta x f(x_i,y_j) + \frac{\Delta x^3}{24} \partial_x^2 f(\xi_i,y_j)\,dx,$$

for $\xi_i \in [x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]$. From this, we obtain

$$\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}} f(x,y)\,dx\,dy = \Delta x \Delta y f(x_i,y_j) + \frac{\Delta x^3}{24} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \partial_x^2 f(\xi_i,y_j)\,dx + \frac{\Delta y^3}{24} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \partial_y^2 f(x,\eta_j)\,dx.$$

Using Theorem A.2, we obtain the desired formula:

$$\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}} f(x,y)\,dx\,dy = \Delta x \Delta y f(x_i,y_j) + \frac{\Delta x^2}{24} \Delta x \Delta y \partial_x^2 f(\nu_i,y_j) + \frac{\Delta y^2}{24} \Delta x \Delta y \partial_y^2 f(\theta_i,\eta_j),$$

where $\nu_i, \theta_i \in [x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]$, recalling that $\Delta x = \Delta y$. □

Corollary A.1. *If $f \in C^2([a,b]\times[c,d])$, and $[a,b]\times[c,d]$ is written as the union of the uniformed-spaces control volumes $[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}] \times [y_{j-\frac{1}{2}}, y_{j+\frac{1}{2}}]$, $i,j = 1,\dots,N$, with lengths $\Delta x = \Delta y$, we have*

$$\int_a^b \int_c^d f(x,y)\,dx\,dy - \sum_{i,j=1}^{N} f(x_i,y_j)\Delta x \Delta y = C_1 \Delta x^2, \tag{A.10}$$

where C_1 depends only on f.

Proof. Using Theorem A.5, we have:

$$\begin{aligned}\frac{1}{\Delta x \Delta y} \int_a^b \int_c^d f(x,y)\,dx\,dy &= \frac{1}{\Delta x \Delta y} \sum_{i,j=1}^{N} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}} f(x,y)\,dx\,dy \\ &= \sum_{i,j=1}^{N} f(x_i,y_j) + \frac{\Delta x^2}{24} \sum_{i,j=1}^{N} \left(\partial_x^2 f(\nu_i,y_j) + \partial_y^2 f(\nu_i,y_j) \right).\end{aligned}$$

We notice that

$$\Delta x \Delta y \sum_{i,j=1}^{N} \left(\partial_x^2 f(\nu_i,y_j) + \partial_y^2 f(\nu_i,y_j) \right) = \frac{(b-a)(d-c)}{N^2} \sum_{i,j=1}^{N} \left(\partial_x^2 f(\nu_i,y_j) + \partial_y^2 f(\nu_i,y_j) \right)$$

and we also point that from the inequality

$$\min\left(\partial_x^2 f + \partial_y^2 f\right)(x,y) \le \frac{1}{N^2}\sum_{i,j=1}^{N}\left(\partial_x^2 f(v_i,y_j) + \partial_y^2 f(v_i,y_j)\right) \le \max\left(\partial_x^2 f + \partial_y^2 f\right)(x,y),$$

and with the aid of the intermediate value theorem, we have

$$\frac{1}{N^2}\sum_{i,j=1}^{N}\left(\partial_x^2 f(v_i,y_j) + \partial_y^2 f(v_i,y_j)\right) = \left(\partial_x^2 f + \partial_y^2 f\right)(\overline{x},\overline{y})$$

for some $(\overline{x},\overline{y}) \in [a,b]\times[c,d]$ from which the claim follows. □

A.3 Convergence of 1D FV-SL schemes

A.3.1 Consistency and convergence

Hereafter, we are going to use the notations introduced in Section 2.1.1. To move towards the convergence of 1D-FV schemes, for Problem 2.4 we introduce the local truncation error (LTE hereafter) τ_i^n following LeVeque (2002):

$$Q_i(t^{n+1}) = Q_i(t^n) - \lambda\left(F^n_{i+\frac{1}{2}}(Q(t^n),\tilde{u}^n_{i+\frac{1}{2}}) - F^n_{i-\frac{1}{2}}(Q(t^n),\tilde{u}^n_{i-\frac{1}{2}})\right) + \Delta t \tau_i^n. \tag{A.11}$$

We the define $\tau^n \in \mathbb{P}_v^N$, which represent the LTEs at the time-step n. Notice the LTE is obtained by replacing the exact solution in Equation (2.21). Since $Q_i(t^n)$ is the exact solution of Equation (2.9), the LTE may be rewritten as

$$\begin{aligned}\tau_i^n = \frac{1}{\Delta x}\Bigg[&\left(\frac{1}{\Delta t}\int_{t^n}^{t^{n+1}} (uq)(x_{i+\frac{1}{2}},t)\,dt - F^n_{i+\frac{1}{2}}(Q(t^n),\tilde{u}^n_{i+\frac{1}{2}})\right) + \\ &\left(\frac{1}{\Delta t}\int_{t^n}^{t^{n+1}} (uq)(x_{i-\frac{1}{2}},t)\,dt - F^n_{i-\frac{1}{2}}(Q(t^n),\tilde{u}^n_{i-\frac{1}{2}})\right)\Bigg].\end{aligned} \tag{A.12}$$

The LTE gives a measure of how well the 1D-FV scheme approximates the integral form of the considered conservation law. Another interpretation of the LTE is that the LTE gives the error obtained after applying the scheme for a single time-step using the exact solution. Now we can define consistency.

Definition A.1 (Consistency). *Let us consider the framework of Problem 2.4. A 1D-FV scheme is said to be consistency in the p-norm if for any sequence of $(\Delta x^{(k)}, \Delta t^{(k)}, \lambda)$-discretizations, $k \in \mathbb{N}$, with $\lim_{k\to\infty}\Delta x^{(k)} = \lim_{k\to\infty}\Delta t^{(k)} = 0$, we have:*

$$\lim_{k\to\infty}\left[\max_{1\le n\le N_T^{(k)}} \|\tau^n\|_{p,\Delta x^{(k)}}\right] = 0,$$

and it is said to be consistent with order P in the $p-$norm if

$$\max_{1\leq n\leq N_T^{(k)}} \|\tau^n\|_{p,\Delta x^{(k)}} = \mathcal{O}(\Delta x^P).$$

From Equation (A.12), it follows that we basically need to ensure that the numerical flux function $\mathcal{F}^n_{i+\frac{1}{2}}$ converges to the time-averaged flux at edges when $\Delta x \to 0$ in order to guarantee consistency.

At last, we define the point-wise error at time-step n by:

$$E_i^n = Q_i(t^n) - Q_i^n, \quad i = 1, \ldots, N,$$

and we define the vector of errors by $E^n \in \mathbb{P}_\nu^N$ with entries E_i^n.

Definition A.2 (Convergence). *Let us consider the framework of Problem 2.4. A 1D-FV scheme is said to be convergent in the p-norm if for any sequence of $(\Delta x^{(k)}, \Delta t^{(k)}, \lambda)$-discretizations, $k \in \mathbb{N}$, with $\lim_{k\to\infty} \Delta x^{(k)} = \lim_{k\to\infty} \Delta t^{(k)} = 0$, we have:*

$$\lim_{k\to\infty} \left[\max_{1\leq n\leq N_T^{(k)}} \|E^n\|_{p,\Delta x^{(k)}} \right] = 0,$$

and it is said to converge with order P in the $p-$norm if

$$\max_{1\leq n\leq N_T^{(k)}} \|E^n\|_{p,\Delta x^{(k)}} = \mathcal{O}(\Delta x^P).$$

Subtracting Equation (2.21) from Equation (A.11) we get the following equation for the error:

$$\begin{aligned} E_i^{n+1} = E_i^n - \lambda \Bigg[&\left(F^n_{i+\frac{1}{2}}(Q(t^n), \tilde{u}^n_{i+\frac{1}{2}}) - F^n_{i+\frac{1}{2}}(Q^n, \tilde{u}^n_{i+\frac{1}{2}}) \right) \\ &- \left(F^n_{i-\frac{1}{2}}(Q(t^n), \tilde{u}^n_{i-\frac{1}{2}}) - F^n_{i-\frac{1}{2}}(Q^n, \tilde{u}^n_{i-\frac{1}{2}}) \right) \Bigg] + \tau_i^n \Delta t. \end{aligned} \tag{A.13}$$

Notice that if $q, u \in C^3$, we can rewrite Equation (A.12) as:

$$\tau_i^n = \left[\frac{1}{\Delta x \Delta t} \int_{t^n}^{t^{n+1}} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \frac{\partial(uq)}{\partial x}(x,t)\, dx\, dt - \left(\frac{F^n_{i+\frac{1}{2}}(Q(t^n), \tilde{u}^n_{i-\frac{1}{2}}) - F^n_{i-\frac{1}{2}}(Q(t^n), \tilde{u}^n_{i-\frac{1}{2}})}{\Delta x} \right) \right].$$

Using the midpoint rule for integration (Theorem A.4) and the mean value theorem for integrals (Theorem A.2), we have:

$$\begin{aligned} \tau_i^n &= \left[\frac{1}{\Delta t} \int_{t^n}^{t^{n+1}} \left(\frac{\partial(uq)}{\partial x}(x_i, t) + \frac{\Delta x^2}{24} \frac{\partial(uq)}{\partial x}(\xi, t) \right) dt - \left(\frac{F^n_{i+\frac{1}{2}}(Q(t^n), \tilde{u}^n_{i+\frac{1}{2}}) - F^n_{i-\frac{1}{2}}(Q(t^n), \tilde{u}^n_{i-\frac{1}{2}})}{\Delta x} \right) \right] \\ &= \left[\frac{1}{\Delta t} \int_{t^n}^{t^{n+1}} \frac{\partial(uq)}{\partial x}(x_i, t)\, dt - \left(\frac{F^n_{i+\frac{1}{2}}(Q(t^n), \tilde{u}^n_{i+\frac{1}{2}}) - F^n_{i-\frac{1}{2}}(Q(t^n), \tilde{u}^n_{i-\frac{1}{2}})}{\Delta x} \right) \right] + \frac{\Delta x^2}{24} \frac{\partial^3(uq)}{\partial x^3}(\xi, \bar{t}), \end{aligned} \tag{A.14}$$

for $\xi \in X_i$ and $\bar{t} \in [t^n, t^{n+1}]$. Therefore, if $q, u \in C^3$ the scheme is consistent, if and only if, $\frac{1}{\Delta t}\int_{t^n}^{t^{n+1}} \frac{\partial(uq)}{\partial x}(x_i, t)\, dt$ is approximated by $\frac{F^n_{i+\frac{1}{2}}(Q(t^n),\tilde{u}^n_{i+\frac{1}{2}}) - F^n_{i-\frac{1}{2}}(Q(t^n),\tilde{u}^n_{i-\frac{1}{2}})}{\Delta x}$. This shall be very useful when we consider two-dimensional schemes, where we are going to use the discrete operators to estimate the divergence of velocity fields.

A.3.2 Stability

In order to define the concept of stability, it is useful to introduce an operator representation of 1D-FV schemes. In the context of Problem 2.4, we define the operators $\mathcal{H}_{\Delta x,n} : \mathbb{P}^N_v \to \mathbb{P}^N_v$ whose i-th entry is given by:

$$[\mathcal{H}_{\Delta x,n}(Q)]_i = Q_i - \lambda\left(F^n_{i+\frac{1}{2}}(Q, \tilde{u}^n_{i+\frac{1}{2}}) - F^n_{i-\frac{1}{2}}(Q, \tilde{u}^n_{i-\frac{1}{2}}) \right), \tag{A.15}$$

for $i = 1, \dots, N$, $n = 0, \dots, N_T - 1$. Notice that the dependence on n is due to the velocity that may be allowed to vary with time. As it is usual, we are assuming periodicity in the entries of Q when we apply the operator $\mathcal{H}_{\Delta x,n}$. Thus, Equation (2.21) may be rewritten in a vector form by

$$Q^{n+1} = \mathcal{H}_{\Delta x,n}(Q^n),$$

and Equation (A.11) in a vector form reads

$$Q(t^{n+1}) = \mathcal{H}_{\Delta x,n}(Q(t^n)) + \Delta t \tau^n,$$

and the error equation (A.13) is given by

$$E^{n+1} = \mathcal{H}_{\Delta x,n}(Q(t^n)) - \mathcal{H}_{\Delta x,n}(Q^n) + \Delta t \tau^n. \tag{A.16}$$

The stability theory focus on uniformly bounding the norm of $\mathcal{H}_{\Delta x,n}(Q(t^n)) - \mathcal{H}_{\Delta x,n}(Q^n)$ (LeVeque, 2002). We define stability as follows.

Definition A.3 (Stability). *In the context of Problem 2.4, a 1D-FV scheme is stable in the $p-$norm if for any $(\Delta x, \Delta t, \lambda)$-discretization of $[a, b] \times [0, T]$ we have:*

$$\|\mathcal{H}_{\Delta x,n}(Q) - \mathcal{H}_{\Delta x,n}(P)\|_{p,\Delta x} \leq (1 + \alpha\Delta t)\|Q - P\|_{p,\Delta x}, \tag{A.17}$$

for all $Q, P \in \mathbb{R}^N_v$ and α is a constant that does not depend neither on Δx nor on Δt.

Assuming that the scheme is stable in the $p-$norm, then it follows from Equation (A.16) that:

$$\begin{aligned}
\|E^{n+1}\|_{p,\Delta x} &\leq \|\mathcal{H}_{\Delta x,n}(Q(t^n)) - \mathcal{H}_{\Delta x,n}(Q^n)\|_{p,\Delta x} + \Delta t \max_{n=1,\dots,N_T} \|\tau^n\|_{p,\Delta x} \\
&\leq (1 + \alpha\Delta t)\|E^n\|_{p,\Delta x} + \Delta t \max_{n=1,\dots,N_T} \|\tau^n\|_{p,\Delta x} \\
&\leq (1 + \alpha\Delta t)^n\|E^0\|_{p,\Delta x} + \Delta t \max_{n=1,\dots,N_T} \|\tau^n\|_{p,\Delta x} \sum_{k=0}^{n-1}(1 + \alpha\Delta t)^k \\
&\leq e^{\alpha T}(\|E^0\|_{p,\Delta x} + T \max_{n=1,\dots,N_T} \|\tau^n\|_{p,\Delta x}),
\end{aligned} \tag{A.18}$$

where we used $n\Delta t \leq T$, $T = N\Delta t$ and the inequality $e^t > 1 + t$. When computing the initial average values using the value at the cell centroid, the initial error E^0 converges to zero provided q is twice continuously differentiable by Proposition 2.2. Therefore, it follows that if the scheme is stable and consistent then it is convergent. Furthermore, if it is stable and consistent with order P, then the convergence order is at least equal to $\min\{P, 2\}$. In the case where both the conservation law and $\mathcal{H}_{\Delta x,n}$ are linear, this result is a particular case of the Lax-Ritchmyer stability and the convergence is guaranteed by the Lax equivalence theorem (LeVeque, 2002). In this Chapter, we are interested only in the linear advection equation. However, as pointed in Section 2.5, the operator $\mathcal{H}_{\Delta x,n}$ may become non-linear when monotonicity constraints are activated.

Notice that, if $\mathcal{H}_{\Delta x,n}$ is linear, then stability is equivalent to require that

$$\|\mathcal{H}_{\Delta x,n}\|_{p,\Delta x} \leq 1 + \alpha\Delta t,$$

where

$$\|\mathcal{H}_{\Delta x,n}\|_{p,\Delta x} = \sup_{Q\in\mathbb{R}^{\Delta x}} \frac{\|\mathcal{H}_{\Delta x,n}(Q)\|_{p,\Delta x}}{\|Q\|_{p,\Delta x}},$$

is the operator p-norm.

For linear operators, we may use the discrete Fourier transform (Trefethen, 2000) to estimate the 2-norm of $\mathcal{H}_{\Delta x,n}$. This approach is known as Von Neumann stability analysis. We define the nodes $\theta_i = i\frac{2\pi}{N}$, $i = 1, \dots, N$, $\Delta\theta = \frac{2\pi}{N}$, $\theta = (\theta_1, \theta_2, \dots, \theta_N)$. The imaginary unit is denoted by $\imath$. We define $\mathbb{C}_\nu^N$ similarly as $\mathbb{P}_\nu^N$. The Fourier modes $e^{\imath k\theta}] \in \mathbb{C}_\nu^N$ for $k = 1, \dots, N$, have entries are given by:

$$[e^{\imath k\theta}]_i = e^{\imath k\theta_i}, \quad \text{for } i = 1, \dots, N.$$

Each k is referred to wavenumber and θ_k is called dimensionless wavenumber. The Fourier modes form an orthogonal basis of $\mathbb{C}_\nu^N$ with respect to the inner product

$$\langle Q, P\rangle = \frac{1}{N}\sum_{i=1}^{N} Q_i\overline{P_i},$$

for $P, Q \in \mathbb{C}_\nu^N$ and $\overline{z}$ denotes the complex conjugate of z. Given $Q \in \mathbb{P}_\nu^N$, we may may express it in terms of the Fourier modes

$$Q = \sum_{k=1}^{N} a_k e^{\imath k\theta},$$

where $a_k \in \mathbb{C}$. The 2-norm of Q is then given by:

$$\|Q\|_{2,\Delta x} = \sqrt{N\sum_{k=1}^{N}|a_k|^2}.$$

The idea of Von Neumann stability analysis is to apply the operator $\mathcal{H}_{\Delta x,n}$ on each Fourier mode and analyze how it modifies its amplitude. For ease of analysis, we assume that the velocity is constant, which implies that the operator $\mathcal{H}_{\Delta x,n}$ has constant coefficients and does not depend on n. For the general case, where the velocity is not constant, the stability can be ensured using the frozen coefficients method (Strikwerda, 2004, p. 59). This method boils down to performing multiple times the stability analysis with a constant velocity being equal to each one of the possible values of the velocity on the grid. If the scheme is stable for all the possible constant velocities, then stability is ensured. Since the operator is supposed to be linear with constant coefficients and we are assuming periodic boundaries conditions, we may write:

$$\mathcal{H}_{\Delta x,n}(e^{\imath k\theta}) = \rho(k)e^{\imath k\theta},$$

where the term $\rho(k)$ is called amplification factor and it is an eigenvalue of $\mathcal{H}_{\Delta x,n}$. The norm of $\mathcal{H}_{\Delta x,n}(Q)$ is bounded by:

$$\|\mathcal{H}_{\Delta x,n}(Q)\|_{2,\Delta x}^2 = N\sum_{k=1}^{N}|a_k|^2|\rho(k)|^2 \leq \max_{k=1,\dots,N}|\rho(k)|^2\|Q\|_{2,\Delta x}^2.$$

Therefore:

$$\|\mathcal{H}_{\Delta x,n}\|_{2,\Delta x} \leq \max_{k=1,\dots,N}|\rho(k)|.$$

If we show that $\max_{k=1,\dots,N}|\rho(k)| \leq 1+\alpha\Delta t$, with α independent of Δt, N and n, then we ensure the stability of $\mathcal{H}_{\Delta x,n}$.

A.3.3 Flux accuracy analysis

With the PPM operator, we can compute the amplification factor by applying it on each Fourier mode considering the PPM and the hybrid PPM schemes, both without monotonization. We assume a constant velocity equal to one and $N = 100$ (number of control volumes). In Figure A.1 we show the amplification factor for both PPM and hybrid PPM schemes considering different CFL numbers. We can observe that both schemes damp most of the Fourier modes for larger k, regardless of the CFL number. Besides that, the hybrid scheme is more effective when reducing the Fourier modes amplitude. We point out that both schemes are exact when the CFL number is equal to 1. From this analysis, we can conclude that the PPM and hybrid PPM schemes satisfy the Von Neumann stability criteria when the CFL restriction is respected. For an analysis of stability for larger time-steps, we refer to Lauritzen (2007).

A.4 Convergence, consistency and stability of 2D-FV schemes

The notions of convergence, consistency and stability for a 2D-FV schemes are straightforward from these notions for 1D-FV schemes (see Subsections A.3.1 and A.3.2). Indeed, in the context of Problem 3.3, we define the operators $\mathcal{H}_{\Delta x,\Delta y,n} : \mathbb{R}^{N\times M} \to \mathbb{R}^{N\times M}$ whose (i,j) entry is given by:

$$[\mathcal{H}_{\Delta x,\Delta y,n}(Q)]_{ij} = Q_{ij} - \Delta t\mathbb{D}_{ij}^n$$

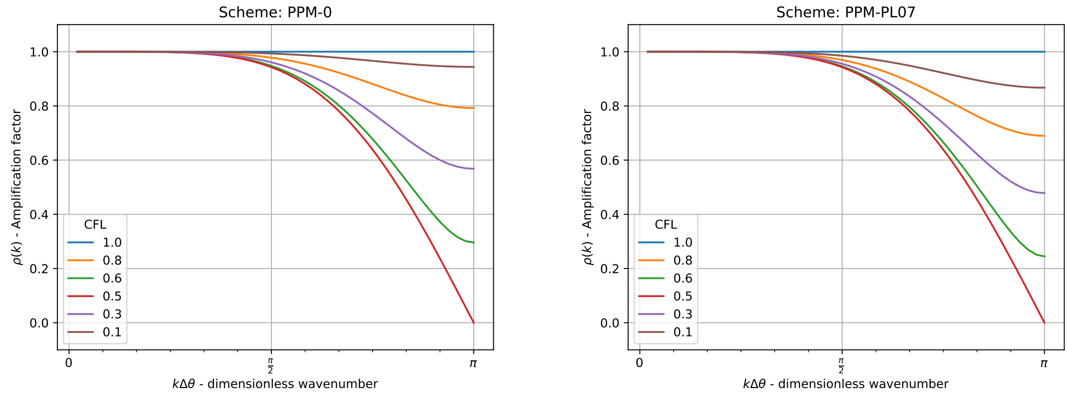


Figure A.1: *Amplification factor for the PPM (left) and hybrid PPM (right) schemes for different CFL numbers.*

for $i = 1, \dots, N$, $j = 1, \dots, M$, $n = 0, \dots, N_T - 1$. The 2D-FV is then expressed as

$$Q^{n+1} = \mathcal{H}_{\Delta x, \Delta y, n}(Q^n).$$

The local error truncation $\tau^n \in \mathbb{R}^{N \times M}$ is given by

$$Q(t^{n+1}) = \mathcal{H}_{\Delta x, \Delta y, n}(Q(t^n)) + \Delta t \tau^n.$$

The error equation is given by

$$E^{n+1} = \mathcal{H}_{\Delta x, \Delta y, n}(Q(t^n)) - \mathcal{H}_{\Delta x, \Delta y, n}(Q^n) + \Delta t \tau^n. \tag{A.19}$$

The stability in the p-norm is defined as in the 1D case.

Definition A.4. *A 2D-FV scheme is stable in the $p-$norm if*

$$\|\mathcal{H}_{\Delta x, \Delta y, n}(Q) - \mathcal{H}_{\Delta x, \Delta y, n}(P)\|_{p, \Delta x \times \Delta y} \leq (1 + \alpha \Delta t)\|Q - P\|_{p, \Delta x \times \Delta y}, \tag{A.20}$$

for all $Q, P \in \mathbb{R}^{N \times M}$ and α is a constant that does not depend neither on Δx, Δy, Δt nor on n.

If a 2D-FV scheme is stable in the $p-$norm, similarly to Equation (A.18) we have:

$$\|E^{n+1}\|_{p, \Delta x \times \Delta y} \leq e^{\alpha T}(\|E^0\|_{p, \Delta x \times \Delta y} + T \max_{n=1,\dots,N_T} \|\tau^n\|_{p, \Delta x \times \Delta y}).$$

Again, we point out that from Proposition 3.1, we have that the initial error E^0 shall be second-order accurate. Consistency is defined as in Definition A.1 and convergence is defined as in Definition A.2.

The Von Neumann analysis can be applied when $\mathcal{H}_{\Delta x, \Delta y, n}$ is linear, since we are considering periodic boundary conditions. The idea is the same as in the one-dimensional case, we just apply the operator $\mathcal{H}_{\Delta x, \Delta y, n}$ on the Fourier modes to obtain the amplification factor. We introduce the nodes $\theta_i = i\frac{2\pi}{N}$, $i = 1, \dots, N$, $\Delta\theta = \frac{2\pi}{N}$, $\theta_i = (\theta_1, \theta_2, \dots, \theta_N)$, $\phi_j = j\frac{2\pi}{M}$, $j = 1, \dots, M$, $\Delta\phi = \frac{2\pi}{M}$, $\phi = (\phi_1, \phi_2, \dots, \phi_M)$. For $k_1 = 1, \dots, N$, $k_2 = 1, \dots, M$, the two-dimensional Fourier mode $\boldsymbol{k} = (k_1, k_2)$ from $\mathbb{C}^{N\times M}$ has its (i, j) entry given by $[e^{\imath\boldsymbol{k}\theta}]_{ij} = e^{\imath k_1\theta_i}e^{\imath k_2\phi_j}$. For an analysis of stability for the dimension splitting method, we refer to Lauritzen (2007) and Lin and Rood (1996).

Notice that if $q, u, v \in C^3$, we can rewrite the LTE as:

$$\tau_{ij}^n = \left[\frac{1}{\Delta x\Delta y\Delta t}\int_{t^n}^{t^{n+1}}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}}\int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}} \nabla\cdot(\boldsymbol{u}q)(x,y,t)\,dy\,dx\,dt - \mathbb{D}_{ij}^n\right].$$

Using the midpoint rule for integration (Theorem A.5), the mean value theorem for integrals (Theorem A.2) and recalling the discrete divergence (Definition 3.5), we have:

$$\tau_{ij}^n = \frac{1}{\Delta t}\int_{t^n}^{t^{n+1}} \nabla\cdot(\boldsymbol{u}q)(x_i,y_j,t)\,dt - \mathbb{D}_{ij}^n + \mathcal{O}(\Delta x^2) + \mathcal{O}(\Delta y^2). \tag{A.21}$$

Therefore, in order to investigate the consistency, we may compare how well the discrete divergence approximates the divergence.

A.5 Finite-difference estimates

This Section aims to prove all finite-difference error estimations used throughout this appendix. All the proves are very simple and consist of applying Taylor's expansions, as it is usual when computing the accuracy order of many numerical schemes.

Lemma A.1. *Let $F \in C^5(\mathbb{R})$, $x_0 \in \mathbb{R}$ and $h > 0$. Then, the following identity holds:*

$$F'(x_0) = \frac{4}{3}\left(\frac{F(x_0+h) - F(x_0-h)}{2h}\right) - \frac{1}{3}\left(\frac{F(x_0+2h) - F(x_0-2h)}{4h}\right) + C_1 h^4, \tag{A.22}$$

where C_1 is a constant that depends only on F and h.

Proof. Given $\delta \in]0, 2h]$, then $x_0 + \delta \in]x_0, x_0 + 2h]$ and $x_0 - \delta \in]x_0 - 2h, x_0]$. Then, we get using the Taylor expansion of F:

$$F(x_0+\delta) = F(x_0) + F'(x_0)\delta + F^{(2)}(x_0)\frac{\delta^2}{2} + F^{(3)}(x_0)\frac{\delta^3}{3!} + F^{(4)}(x_0)\frac{\delta^3}{4!} + F^{(5)}(\theta_\delta)\frac{\delta^5}{5!} \quad \theta_\delta \in [x_0, x_0+\delta],$$
$$F(x_0-\delta) = F(x_0) - F'(x_0)\delta + F^{(2)}(x_0)\frac{\delta^2}{2} - F^{(3)}(x_0)\frac{\delta^3}{3!} + F^{(4)}(x_0)\frac{\delta^4}{4!} - F^{(5)}(\theta_{-\delta})\frac{\delta^5}{5!}, \quad \theta_{-\delta} \in [x_0-\delta, x_0].$$

Thus:

$$\frac{F(x_0+\delta) - F(x_0-\delta)}{2\delta} = F'(x_0) + F^{(3)}(x_0)\frac{\delta^2}{3!} + \left(F^{(5)}(\theta_\delta) + F^5(\theta_{-\delta})\right)\frac{\delta^4}{2\cdot 5!}, \tag{A.23}$$

Applying Equation (A.23) for $\delta = h$ and $\delta = 2h$, we get, respectively:

$$\frac{F(x_0+h)-F(x_0-h)}{2h} = F'(x_0)+F^{(3)}(x_0)\frac{h^2}{3!}+\left(F^{(5)}(\theta_h)+F^{5)}(\theta_{-h})\right)\frac{h^4}{2\cdot 5!}, \quad \theta_h \in [x_0, x_0+h], \quad \theta_{-h} \in [x_0-h, x \tag{A.24}$$

and

$$\frac{F(x_0+2h)-F(x_0-2h)}{4h} = F'(x_0)+F^{(3)}(x_0)\frac{4h^2}{3!}+\left(F^{(5)}(\theta_{2h})+F^{(5)}(\theta_{-2h})\right)\frac{16h^4}{2\cdot 5!}, \tag{A.25}$$
$$\theta_{2h} \in [x_0, x_0+2h], \quad \theta_{-2h} \in [x_0-2h, x_0].$$

Using Equations (A.24) and (A.25), we obtain:

$$\frac{4}{3}\left(\frac{F(x_0+h)-F(x_0-h)}{2h}\right) = \frac{4}{3}F'(x_0)+F^{(3)}(x_0)\frac{4h^2}{3\cdot 3!}+\left(F^{(5)}(\theta_h)+F^{(5)}(\theta_{-h})\right)\frac{h^4}{2\cdot 5!}, \tag{A.26}$$

$$\frac{1}{3}\left(\frac{F(x_0+2h)-F(x_0-2h)}{4h}\right) = \frac{1}{3}F'(x_0)+F^{(3)}(x_0)\frac{4h^2}{3\cdot 3!}+\left(F^{(5)}(\theta_{2h})+F^{(5)}(\theta_{-2h})\right)\frac{16h^4}{3\cdot 2\cdot 5!} \tag{A.27}$$

Subtracting Equation (A.27) from Equation (A.26) we get the desired Equation (A.22) with

$$C_1 = \frac{1}{720}\left(3F^{(5)}(\theta_h)+3F^{(5)}(\theta_{-h})-16F^{(5)}(\theta_{2h})-16F^{(5)}(\theta_{-2h})\right), \tag{A.28}$$

where $\theta_h \in [x_0, x_0+h], \theta_{-h} \in [x_0-h, x_0], \theta_{2h} \in [x_0, x_0+2h], \theta_{-2h} \in [x_0-2h, x_0]$. Using the intermediate value theorem, we can express C_1 in a more compact way as

$$C_1 = \frac{1}{720}\left(6F^{(5)}(\eta_1)-32F^{(5)}(\eta_2)\right), \tag{A.29}$$

where $\eta_1, \eta_2 \in [x_0-2h, x_0+2h]$, which concludes the proof. □

Lemma A.2. *Let $F \in C^4(\mathbb{R})$, $x_0 \in \mathbb{R}$ and $h > 0$. Then, the following identity holds:*

$$F''(x_0) = \frac{-2F(x_0-2h)+15F(x_0-h)-28F(x_0)+20F(x_0+h)-6F(x_0+2h)+F(x_0+3h)}{6h^2}+C_2h^2, \tag{A.30}$$

where C_2 is a constant that depends only on F and h.

Proof. From the Taylor's expansion, we have:

$$
\begin{aligned}
F(x_0 - 2h) &= F(x_0) - 2F'(x_0)h + 2F^{(2)}(x_0)h^2 - \frac{8}{6}F^{(3)}(x_0)h^3 + \frac{16}{24}F^{(4)}(\theta_{-2h})h^4, \\
F(x_0 - h) &= F(x_0) - F'(x_0)h + \frac{1}{2}F^{(2)}(x_0)h^2 - \frac{1}{6}F^{(3)}(x_0)h^3 + \frac{1}{24}F^{(4)}(\theta_{-h})h^4, \\
F(x_0 + h) &= F(x_0) + F'(x_0)h + \frac{1}{2}F^{(2)}(x_0)h^2 + \frac{1}{6}F^{(3)}(x_0)h^3 + \frac{1}{24}F^{(4)}(\theta_{h})h^4, \\
F(x_0 + 2h) &= F(x_0) + 2F'(x_0)h + 2F^{(2)}(x_0)h^2 + \frac{8}{6}F^{(3)}(x_0)h^3 + \frac{16}{24}F^{(4)}(\theta_{2h})h^4, \\
F(x_0 + 3h) &= F(x_0) + 3F'(x_0)h + \frac{9}{2}F^{(2)}(x_0)h^2 + \frac{27}{6}F^{(3)}(x_0)h^3 + \frac{81}{24}F^{(4)}(\theta_{3h})h^4,
\end{aligned}
$$

where $\theta_{-2h} \in [x_0 - 2h, x_0 - h]$, $\theta_{-h} \in [x_0 - h, x_0]$, $\theta_h \in [x_0, x_0 + h]$, $\theta_{2h} \in [x_0 + h, x_0 + 2h]$, $\theta_{3h} \in [x_0 + 2h, x_0 + 3h]$. Multiplying these equations by their respective coefficients given in Equation (A.30), one get:

$$
\begin{aligned}
-2F(x_0 - 2h) &= -2F(x_0) + 4F'(x_0)h - 4F^{(2)}(x_0)h^2 + \frac{16}{6}F^{(3)}(x_0)h^3 - \frac{32}{24}F^{(4)}(\theta_{-2h})h^4, \\
15F(x_0 - h) &= 15F(x_0) - 15F'(x_0)h + \frac{15}{2}F^{(2)}(x_0)h^2 - \frac{15}{6}F^{(3)}(x_0)h^3 + \frac{15}{24}F^{(4)}(\theta_{-h})h^4, \\
-28F(x_0) &= -28F(x_0), \\
20F(x_0 + h) &= 20F(x_0) + 20F'(x_0)h + 10F^{(2)}(x_0)h^2 + \frac{20}{6}F^{(3)}(x_0)h^3 + \frac{20}{24}F^{(4)}(\theta_{h})h^4, \\
-6F(x_0 + 2h) &= -6F(x_0) - 12F'(x_0)h - 12F^{(2)}(x_0)h^2 - 8F^{(3)}(x_0)h^3 - \frac{96}{24}F^{(4)}(\theta_{2h})h^4, \\
F(x_0 + 3h) &= F(x_0) + 3F'(x_0)h + \frac{9}{2}F^{(2)}(x_0)h^2 + \frac{27}{6}F^{(3)}(x_0)h^3 + \frac{81}{24}F^{(4)}(\theta_{3h})h^4.
\end{aligned}
$$

Summing all these equations, we get the desired Formula (A.30) with C_2 given by:

$$
C_2 = \frac{1}{24}\left(32F^{(4)}(\theta_{-2h}) - 15F^{(4)}(\theta_{-h}) - 20F^{(4)}(\theta_h) + 96F^{(4)}(\theta_{2h}) - 81F^{(4)}(\theta_{3h})\right). \tag{A.31}
$$

Using the intermediate value theorem, we can express C_2 in a more compact way as

$$
C_2 = \frac{1}{24}\left(128F^{(5)}(\eta_1) - 116F^{(5)}(\eta_2)\right), \tag{A.32}
$$

where $\eta_1, \eta_2 \in [x_0 - 2h, x_0 + 3h]$, which concludes the proof. □

Lemma A.3. *Let $F \in C^4(\mathbb{R})$, $x_0 \in \mathbb{R}$ and $h > 0$. Then, the following identity holds:*

$$F^{(3)}(x_0) = \frac{F(x_0 - 2h) - 7F(x_0 - h) + 16F(x_0) - 16F(x_0 + h) + 7F(x_0 + 2h) - F(x_0 + 3h)}{2h^3} + C_3 h, \tag{A.33}$$

where C_3 is a constant that depends only on F and h.

Proof. From the Taylor's expansion, we have:

$$\begin{aligned}
F(x_0 - 2h) &= F(x_0) - 2F'(x_0)h + 2F^{(2)}(x_0)h^2 - \frac{8}{6}F^{(3)}(x_0)h^3 + \frac{16}{24}F^{(4)}(\theta_{-2h})h^4,\\
F(x_0 - h) &= F(x_0) - F'(x_0)h + \frac{1}{2}F^{(2)}(x_0)h^2 - \frac{1}{6}F^{(3)}(x_0)h^3 + \frac{1}{24}F^{(4)}(\theta_{-h})h^4,\\
F(x_0 + h) &= F(x_0) + F'(x_0)h + \frac{1}{2}F^{(2)}(x_0)h^2 + \frac{1}{6}F^{(3)}(x_0)h^3 + \frac{1}{24}F^{(4)}(\theta_{h})h^4,\\
F(x_0 + 2h) &= F(x_0) + 2F'(x_0)h + 2F^{(2)}(x_0)h^2 + \frac{8}{6}F^{(3)}(x_0)h^3 + \frac{16}{24}F^{(4)}(\theta_{2h})h^4,\\
F(x_0 + 3h) &= F(x_0) + 3F'(x_0)h + \frac{9}{2}F^{(2)}(x_0)h^2 + \frac{27}{6}F^{(3)}(x_0)h^3 + \frac{81}{24}F^{(4)}(\theta_{3h})h^4,
\end{aligned}$$

where $\theta_{-2h} \in [x_0 - 2h, x_0 - h], \theta_{-h} \in [x_0 - h, x_0], \theta_h \in [x_0, x_0 + h], \theta_{2h} \in [x_0 + h, x_0 + 2h], \theta_{3h} \in [x_0 + 2h, x_0 + 3h]$. Multiplying these equations by their respective coefficients given in Equation (A.33), one get:

$$\begin{aligned}
F(x_0 - 2h) &= F(x_0) - 2F'(x_0)h + \frac{4}{2}F^{(2)}(x_0)h^2 - \frac{8}{6}F^{(3)}(x_0)h^3 + \frac{16}{24}F^{(4)}(\theta_{-2h})h^4,\\
-7F(x_0 - h) &= -7F(x_0) + 7F'(x_0)h - \frac{7}{2}F^{(2)}(x_0)h^2 + \frac{7}{6}F^{(3)}(x_0)h^3 - \frac{7}{24}F^{(4)}(\theta_{-h})h^4,\\
16F(x_0) &= 16F(x_0),\\
-16F(x_0 + h) &= -16F(x_0) - 16F'(x_0)h - \frac{16}{2}F^{(2)}(x_0)h^2 - \frac{16}{6}F^{(3)}(x_0)h^3 - \frac{16}{24}F^{(4)}(\theta_{h})h^4,\\
7F(x_0 + 2h) &= 7F(x_0) + 14F'(x_0)h + \frac{28}{2}F^{(2)}(x_0)h^2 + \frac{56}{6}F^{(3)}(x_0)h^3 + \frac{112}{24}F^{(4)}(\theta_{2h})h^4,\\
-F(x_0 + 3h) &= -F(x_0) - 3F'(x_0)h - \frac{9}{2}F^{(2)}(x_0)h^2 - \frac{27}{6}F^{(3)}(x_0)h^3 - \frac{81}{24}F^{(4)}(\theta_{3h})h^4.
\end{aligned}$$

Summing all these equations, we have:

$$F(x_0 - 2h) - 7F(x_0 - h) + 16F(x_0) - 16F(x_0 + h) + 7F(x_0 + 2h) - F(x_0 + 3h) = 2F^{(3)}(x_0)h^3 - 2C_3 h^4,$$

we get the desired Formula (A.33) with C_3 given by:

$$C_3 = \frac{1}{48}\Bigg(- 16F^{(4)}(\theta_{-2h}) + 7F^{(4)}(\theta_{-h}) + 16F^{(4)}(\theta_h) - 112F^{(4)}(\theta_{2h}) + 81F^{(4)}(\theta_{3h})\Bigg). \tag{A.34}$$

Using the intermediate value theorem, we can express C_3 in a more compact way as

$$C_3 = \frac{1}{48}\left(104F^{(5)}(\eta_1) - 128F^{(5)}(\eta_2)\right), \tag{A.35}$$

where $\eta_1, \eta_2 \in [x_0 - 2h, x_0 + 3h]$, which concludes the proof. □

A.6 PPM reconstruction accuracy analysis

In this Section, we are going to investigate the accuracy of the PPM reconstruction process. As we pointed out in Section 2.4.1, the approximation of q at the control volumes edges given by Equation (2.47) is fourth-order accurate when $q \in C^4(\mathbb{R})$. This is proved as a Corollary of the following Proposition A.1.

Proposition A.1. *Let $q \in C^4(\mathbb{R})$, $\overline{x} \in \mathbb{R}$ and $h > 0$. Then, the following identity holds:*

$$q(\overline{x}) = \frac{7}{12}\left(\frac{1}{h}\int_{\overline{x}}^{\overline{x}+h} q(x)\,dx + \frac{1}{h}\int_{\overline{x}-h}^{\overline{x}} q(x)\,dx\right) - \frac{1}{12}\left(\frac{1}{h}\int_{\overline{x}+h}^{\overline{x}+2h} q(x)\,dx + \frac{1}{h}\int_{\overline{x}-2h}^{\overline{x}-h} q(x)\,dx\right) + C_1 h^4, \tag{A.36}$$

where C_1 is a constant that depends on q and h.

Proof. We define $Q(x) = \int_a^x q(\xi)\,d\xi$ for fixed $a \in \mathbb{R}$ as in Equation (2.38). It follows that:

$$\begin{aligned}
\int_{\overline{x}}^{\overline{x}+h} q(\xi)\,d\xi + \int_{\overline{x}-h}^{\overline{x}} q(\xi)\,d\xi &= Q(\overline{x}+h) - Q(\overline{x}-h),\\
\int_{\overline{x}+h}^{\overline{x}+2h} q(\xi)\,d\xi + \int_{\overline{x}-2h}^{\overline{x}-h} q(\xi)\,d\xi &= Q(\overline{x}+2h) - Q(\overline{x}-2h) - (Q(\overline{x}+h) - Q(\overline{x}-h)).
\end{aligned}$$

Using these identities, Equation (A.36) may be rewritten as:

$$q(\overline{x}) = \frac{4}{3}\left(\frac{Q(\overline{x}+h) - Q(\overline{x}-h)}{2h}\right) - \frac{1}{3}\left(\frac{Q(\overline{x}+2h) - Q(\overline{x}-2h)}{4h}\right) + C_1 h^4, \tag{A.37}$$

which consists of finite-difference approximations. Thus, Equation (A.36) follows from Lemma A.1 with:

$$C_1 = C_1(\mu_1, \mu_2) = \frac{1}{720}\left(6q^{(4)}(\mu 1) - 32q^{(4)}(\mu_2)\right), \tag{A.38}$$

where $\mu_1, \mu_2 \in [\overline{x} - 2h, \overline{x} + 2h]$, which concludes the proof. □

Corollary A.2. *It follows from Proposition A.1 with $\overline{x} = x_{i+\frac{1}{2}}$ and $h = \Delta x$ that $q_{i+\frac{1}{2}}$ given by Equation (2.47) satisfies:*

$$q(x_{i+\frac{1}{2}}) - q_{i+\frac{1}{2}} = C_1 \Delta x^4, \tag{A.39}$$

with C_1 given by Equation (A.38), whenever $q \in C^4(\mathbb{R})$.

The parabolic function from (2.41) given with coefficients specified before approximates q with order 3 when $q \in C^4(\mathbb{R})$. In order to check this, for $x \in X_i$ we rewrite Equation (2.41) as:

$$q_i(x;Q) = q_{L,i} + \frac{(\Delta q_i + q_{6,i})}{\Delta x}(x - x_{i-\frac{1}{2}}) - \frac{q_{6,i}}{\Delta x^2}(x - x_{i-\frac{1}{2}})^2 \tag{A.40}$$

and we write q using its Taylor expansion assuming $q \in C^4(\mathbb{R})$:

$$q(x) = q(x_{i-\frac{1}{2}}) + q'(x_{i-\frac{1}{2}})(x - x_{i-\frac{1}{2}}) + \frac{q''(x_{i-\frac{1}{2}})}{2}(x - x_{i-\frac{1}{2}})^2 + \frac{q^{(3)}(\theta_i)}{6}(x - x_{i-\frac{1}{2}})^3, \tag{A.41}$$

where $\theta_i \in X_i$. Comparing Equation (A.40) with Equation (A.41), it is reasonable to seek to some bound to the expressions:

$$q'(x_{i-\frac{1}{2}}) - \frac{(\Delta q_i + q_{6,i})}{\Delta x}, \tag{A.42}$$

and:

$$\frac{q''(x_{i-\frac{1}{2}})}{2} - \left(- \frac{q_{6,i}}{\Delta x^2} \right). \tag{A.43}$$

We have seen that term $q_{L,i}$ gives a fourth-order approximation to $q(x_{i-\frac{1}{2}})$. The Corollary A.3 shall prove that the term (A.42) has a bound proportional to Δx^2, and the Corollary A.4 shall prove that the term (A.43) is bounded by a constant times Δx.

Before proving the desired bounds, it is useful to rewrite some terms explicitly as functions of the values of the Δx-grid function Q. Combining Equation (2.44) with Equations (2.48) and (2.49), we may write $q_{6,i}$ as:

$$q_{6,i} = \frac{1}{4}\left(Q_{i-2} - 6Q_{i-1} + 10Q_i - 6Q_{i+1} + Q_{i+2} \right). \tag{A.44}$$

Recalling the definition of Δq_i from Equation (2.42), and applying Equations (2.48) and (2.49), we may express Δq_i as:

$$\Delta q_i = \frac{1}{12}\left(Q_{i-2} - 8Q_{i-1} + 8Q_{i+1} - Q_{i+2} \right). \tag{A.45}$$

Finally, we combine Equations (A.44) and (A.45) and write their sum as:

$$\frac{(\Delta q_i + q_{6,i})}{\Delta x} = \frac{2Q_{i-2} - 13Q_{i-1} + 15Q_i - 5Q_{i+1} + Q_{i+2}}{6\Delta x}. \tag{A.46}$$

The next Proposition A.2 proves that Equation (A.46) approximates $q'(x_{i-\frac{1}{2}})$ with order 2.

Proposition A.2. *Let $q \in C^3(\mathbb{R})$, $\overline{x} \in \mathbb{R}$ and $h > 0$. Then, the following identity holds:*

$$\begin{aligned} q'(\overline{x}) = \frac{1}{6h}\bigg(\frac{2}{h}\int_{\overline{x}-2h}^{\overline{x}-h} q(x)\,dx - \frac{13}{h}\int_{\overline{x}-h}^{\overline{x}} q(x)\,dx + \frac{15}{h}\int_{\overline{x}}^{\overline{x}+h} q(x)\,dx \\ -\frac{5}{h}\int_{\overline{x}+h}^{\overline{x}+2h} q(x)\,dx + \frac{1}{h}\int_{\overline{x}+2h}^{\overline{x}+3h} q(x)\,dx \bigg) + C_2 h^2, \end{aligned} \tag{A.47}$$

where C_2 is a constant that depends on q and h.

Proof. We consider again $Q(x) = \int_a^x q(\xi)\,d\xi$ for $a \in \mathbb{R}$ fixed as in Equation (2.38). Like in Proposition A.2, we have:

$$\begin{aligned} &\frac{1}{6h}\bigg(\frac{2}{h}\int_{\overline{x}-2h}^{\overline{x}-h} q(x)\,dx - \frac{13}{h}\int_{\overline{x}-h}^{\overline{x}} q(x)\,dx + \frac{15}{h}\int_{\overline{x}}^{\overline{x}+h} q(x)\,dx - \frac{5}{h}\int_{\overline{x}+h}^{\overline{x}+2h} q(x)\,dx + \frac{1}{h}\int_{\overline{x}+2h}^{\overline{x}+3h} q(x)\,dx \bigg) \\ &= \frac{1}{6h}\bigg(\frac{2}{h}\big(Q(\overline{x}-h) - Q(\overline{x}-2h)\big) - \frac{13}{h}\big(Q(\overline{x}) - Q(\overline{x}-h)\big) + \frac{15}{h}\big(Q(\overline{x}+h) - Q(\overline{x})\big) \\ &\qquad -\frac{5}{h}\big(Q(\overline{x}+2h) - Q(\overline{x}+h)\big) + \frac{1}{h}\big(Q(\overline{x}+3h) - Q(\overline{x}+2h)\big)\bigg) \\ &= \frac{1}{6h^2}\bigg(-2Q(\overline{x}-2h) + 15Q(\overline{x}-h) - 28Q(\overline{x}) + 20Q(\overline{x}+h) - 6Q(\overline{x}+2h) + Q(\overline{x}+3h)\bigg), \end{aligned}$$

which consists of the finite-difference scheme from Lemma A.2. Therefore, Equation (A.47) follows from Lemma A.2 with:

$$C_2 = C_2(\mu_1, \mu_2) = \frac{1}{24}\bigg(128q^{(3)}(\mu_1) - 116q^{(3)}(\mu_2)\bigg), \tag{A.48}$$

where $\mu_1, \mu_2 \in [x_0 - 2h, x_0 + 3h]$, which concludes the proof. □

Corollary A.3. *It follows from Proposition A.2 with $\overline{x} = x_{i-\frac{1}{2}}$ and $h = \Delta x$ that Δq_i given by Equation (A.45) and $q_{6,i}$ given by Equation (A.44) satisfy:*

$$q'(x_{i-\frac{1}{2}}) - \frac{(\Delta q_i + q_{6,i})}{\Delta x} = C_2 \Delta x^2, \tag{A.49}$$

with C_2 given by Equation (A.48), whenever $q \in C^3(\mathbb{R})$.

Now, we analyse the following expression:

$$-\frac{2q_{6,i}}{\Delta x^2} = -\frac{1}{2\Delta x^2}\bigg(Q_{i-2} - 6Q_{i-1} + 10Q_i - 6Q_{i+1} + Q_{i+2}\bigg). \tag{A.50}$$

deduced from Equation (A.44) and we prove in Proposition A.3 that Equation (A.50) approximates $q''(x_{i-\frac{1}{2}})$ with order 1.

Proposition A.3. *Let $q \in C^3(\mathbb{R})$, $\overline{x} \in \mathbb{R}$ and $h > 0$. Then, the following identity holds:*

$$
\begin{aligned}
q''(\overline{x}) = \frac{1}{2h^2}\Bigg(-\frac{1}{h}\int_{\overline{x}-2h}^{\overline{x}-h} q(x)\,dx + \frac{6}{h}\int_{\overline{x}-h}^{\overline{x}} q(x)\,dx - \frac{10}{h}\int_{\overline{x}}^{\overline{x}+h} q(x)\,dx \\
+\frac{6}{h}\int_{\overline{x}+h}^{\overline{x}+2h} q(x)\,dx - \frac{1}{h}\int_{\overline{x}+2h}^{\overline{x}+3h} q(x)\,dx \Bigg) + C_3 h,
\end{aligned}
\tag{A.51}
$$

where C_3 is a constant that depends on q and h.

Proof. Similarly to Proposition A.2 using the same function Q, we have:

$$
\begin{aligned}
&\frac{1}{2h^2}\Bigg(-\frac{1}{h}\int_{\overline{x}-2h}^{\overline{x}-h} q(x)\,dx + \frac{6}{h}\int_{\overline{x}-h}^{\overline{x}} q(x)\,dx - \frac{10}{h}\int_{\overline{x}}^{\overline{x}+h} q(x)\,dx + \frac{6}{h}\int_{\overline{x}+h}^{\overline{x}+2h} q(x)\,dx - \frac{1}{h}\int_{\overline{x}+2h}^{\overline{x}+3h} q(x)\,dx \Bigg) \\
&= \frac{1}{2h^2}\Bigg(-\frac{1}{h}\big(Q(\overline{x}-h) - Q(\overline{x}-2h)\big) + \frac{6}{h}\big(Q(\overline{x}) - Q(\overline{x}-h)\big) - \frac{10}{h}\big(Q(\overline{x}+h) - Q(\overline{x})\big) \\
&\qquad +\frac{6}{h}\big(Q(\overline{x}+2h) - Q(\overline{x}+h)\big) - \frac{1}{h}\big(Q(\overline{x}+3h) - Q(\overline{x}+2h)\big)\Bigg) \\
&= \frac{1}{2h^3}\Bigg(Q(\overline{x}-2h) - 7Q(\overline{x}-h) + 16Q(\overline{x}) - 16Q(\overline{x}+h) + 7Q(\overline{x}+2h) - Q(\overline{x}+3h)\Bigg),
\end{aligned}
$$

which consists of the finite-difference scheme from Lemma A.3. Therefore, Equation (A.51) follows from Lemma A.3 with:

$$
C_3 = C_3(\mu_1, \mu_2) = \frac{1}{48}\Bigg(104q^{(3)}(\mu_1) - 128q^{(3)}(\mu_2)\Bigg), \tag{A.52}
$$

where $\mu_1, \mu_2 \in [x_0 - 2h, x_0 + 3h]$, which concludes the proof. □

Corollary A.4. *It follows from Proposition A.3 with $\overline{x} = x_{i-\frac{1}{2}}$ and $h = \Delta x$ that $q_{6,i}$ given by Equation (2.47) satisfies:*

$$
q''(x_{i-\frac{1}{2}}) - \Bigg(-\frac{2q_{6,i}}{\Delta x^2}\Bigg) = C_3 \Delta x, \tag{A.53}
$$

with C_3 given by Equation (A.52), whenever $q \in C^3(\mathbb{R})$.

With the aid of Corollaries A.2, A.3, and A.4, we are able to prove that the PPM reconstruction approximates q with order 3. Indeed, we prove this on the follow up Proposition A.4.

Proposition A.4. *Let $q \in C^4([a,b])$. Then, the Piecewise-Parabolic function given by Equation (2.41) with the parameters $q_{R,i}$ and $q_{L,i}$ obeying Equations (2.48) and (2.49) gives a third-order approximation to q on the control volume X_i. Namely, there exist constants M_1 and M_2 such that*

$$
|q(x) - q_i(x; Q)| \leq M_1 \Delta x^4 + M_2 \Delta x^3, \quad \forall x \in X_i.
$$

Proof. For $x \in X_i$, from Equations (A.41) and (A.40), we have:

$$q(x) - q_i(x;Q) = (q'(x_{i-\frac{i}{2}}) - q_{L,i}) + \left(q'(x_{i-\frac{1}{2}}) - \frac{(\Delta q_i + q_{6,i})}{\Delta x}\right)(x - x_{i-\frac{1}{2}}) \\ + \left(\frac{q''(x_{i-\frac{1}{2}})}{2} + \frac{q_{6,i}}{\Delta x^2}\right)(x - x_{i-\frac{1}{2}})^2 + \frac{q^{(3)}(\theta_i)}{6}(x - x_{i-\frac{1}{2}})^3.$$

Using this fact with Corollaries A.2, A.3, and A.4, we have:

$$q(x) - q_i(x;Q) = C_1\Delta x^4 + C_2\Delta x^2(x - x_{i-\frac{1}{2}}) + \frac{C_3}{2}\Delta x(x - x_{i-\frac{1}{2}})^2 + C_4(x - x_{i-\frac{1}{2}})^3,$$

where C_1, C_2 and C_3 are given by Equations (A.38), (A.48) and (A.52), respectively, and

$$C_4 = C_4(\theta_i) = \frac{q^{(3)}(\theta_i)}{6}. \tag{A.54}$$

For $x \in X_i$, we have $|x - x_{i-\frac{i}{2}}| \leq \Delta x$, thus:

$$|q(x) - q_i(x;Q)| \leq M_1\Delta x^4 + M_2\Delta x^3,$$

where

$$M_1 = \frac{38}{720}\sup_{\xi\in[a,b]}|q^{(4)}(\xi)|,$$

$$M_2 = \left(\frac{244}{24} + \frac{232}{96} + \frac{1}{6}\right)\sup_{\xi\in[a,b]}|q^{(3)}(\xi)| = \frac{143}{12}\sup_{\xi\in[a,b]}|q^{(3)}(\xi)|,$$

which concludes the proof. □

A.7 Splitting analysis

$$x^d_{i+\frac{1}{2},j}(t^n, t^{n+1}) = x_{i+\frac{1}{2}} - \tilde{u}^n_{i+\frac{1}{2},j}\Delta t, \tag{A.55}$$

$$\tilde{u}^n_{i+\frac{1}{2}}(y) = \frac{1}{\Delta t}\int_{t^n}^{t^{n+1}} u\left(x^d_{i+\frac{1}{2},j}(\theta, t^{n+1}), y, \theta\right) d\theta. \tag{A.56}$$

$$y^d_{i,j+\frac{1}{2}}(t^n, t^{n+1}) = y_{j+\frac{1}{2}} - \tilde{v}^n_{i,j+\frac{1}{2}}\Delta t, \tag{A.57}$$

$$\tilde{v}^n_{j+\frac{1}{2}}(x) = \frac{1}{\Delta t}\int_{t^n}^{t^{n+1}} v\left(x, y^d_{i,j+\frac{1}{2}}(\theta, t^{n+1}), \theta\right) d\theta. \tag{A.58}$$

$$\mathbf{F}^E_i[q(x,y,t^n), \tilde{u}^n](y) = -\lambda\frac{1}{\Delta t}\left(\int_{x^d_{i+\frac{1}{2},j}(t^n,t^{n+1})}^{x_{i+\frac{1}{2}}} q(x,y,t^n)\,dx - \int_{x^d_{i-\frac{1}{2},j}(t^n,t^{n+1})}^{x_{i-\frac{1}{2}}} q(x,y,t^n)\,dx\right),$$

$$\mathbf{G}_j^E[q(x,y,t^n),\tilde{v}^n](x) = -\lambda\frac{1}{\Delta t}\left(\int_{y^d_{i,j+\frac{1}{2}}(t^n,t^{n+1})}^{y_{j+\frac{1}{2}}} q(x,y,t^n)\,dy - \int_{y^d_{i,j-\frac{1}{2}}(t^n,t^{n+1})}^{y_{j-\frac{1}{2}}} q(x,y,t^n)\,dy\right),$$

$$\mathbf{F}_i^E[\overline{q},\tilde{u}^n](y) = -\lambda\overline{q}\delta_i\tilde{u}_i^n(y),$$

$$\mathbf{G}_j^E[\overline{q},\tilde{v}^n](x) = -\lambda\overline{q}\delta_j\tilde{v}_j^n(x),$$

$$\begin{aligned}
&\mathbf{F}_{ij}^E[-\lambda\overline{q}\delta_j\tilde{v}_j^n(x),\tilde{u}^n] = -\lambda\overline{q}\mathbf{F}_{ij}^E[\delta_j\tilde{v}_j^n(x),\tilde{u}^n]\\
&= -\lambda^2\overline{q}\frac{1}{\Delta t}\left(\int_{x^d_{i+\frac{1}{2},j}(t^n,t^{n+1})}^{x_{i+\frac{1}{2}}} \delta_j\tilde{v}_j^n(x)\,dx - \int_{x^d_{i-\frac{1}{2},j}(t^n,t^{n+1})}^{x_{i-\frac{1}{2}}} \delta_j\tilde{v}_j^n(x)\,dx\right)\\
&= \frac{-\overline{q}}{\Delta x^2}\left(\int_{t^n}^{t^{n+1}} \left[\Psi\left(x_{i+\frac{1}{2}},y^d_{i,j+\frac{1}{2}}(\theta,t^{n+1}),\theta\right) - \Psi\left(x_{i-\frac{1}{2}},y^d_{i,j+\frac{1}{2}}(\theta,t^{n+1}),\theta\right)\right] -\right.\\
&\left.\left[\Psi\left(x^d_{i+\frac{1}{2},j},y^d_{i,j-\frac{1}{2}}(\theta,t^{n+1}),\theta\right) - \Psi\left(x^d_{i-\frac{1}{2},j},y^d_{i,j-\frac{1}{2}}(\theta,t^{n+1}),\theta\right)\right]d\theta\right)
\end{aligned}$$

$$\begin{aligned}
&\mathbf{G}_{ij}^E[-\lambda\overline{q}\delta_i\tilde{u}_i^n(y),\tilde{v}^n] = -\lambda\overline{q}\mathbf{G}_{ij}^E[\delta_i\tilde{u}_i^n(y),\tilde{v}^n]\\
&= -\lambda^2\overline{q}\frac{1}{\Delta t}\left(\int_{y^d_{i,j+\frac{1}{2}}(t^n,t^{n+1})}^{y_{j+\frac{1}{2}}} \delta_i\tilde{u}_i^n(y)\,dy - \int_{y^d_{i,j-\frac{1}{2}}(t^n,t^{n+1})}^{y_{j-\frac{1}{2}}} \delta_i\tilde{u}_i^n(y)\,dy\right)\\
&= \frac{\overline{q}}{\Delta x^2}\left(\int_{t^n}^{t^{n+1}} \left[\Psi\left(x^d_{i+\frac{1}{2},j}(\theta,t^{n+1}),y_{j+\frac{1}{2}},\theta\right) - \Psi\left(x^d_{i+\frac{1}{2},j}(\theta,t^{n+1}),y_{j-\frac{1}{2}},\theta\right)\right] -\right.\\
&\left.\left[\Psi\left(x^d_{i-\frac{1}{2},j}(\theta,t^{n+1}),y^d_{i,j+\frac{1}{2}},\theta\right) - \Psi\left(x^d_{i-\frac{1}{2},j}(\theta,t^{n+1}),y^d_{i,j-\frac{1}{2}},\theta\right)\right]d\theta\right)
\end{aligned}$$

Appendix B

Code avaibility

The codes needed for this work have been built openly at GitHub. The PPM implementation for the one-dimensional advection equation used in Chapter 2 is available at https://github.com/luanfs/FV3_adv_1D.

References

Arakawa, A., & Lamb, V. R. (1977). Computational design of the basic dynamical processes of the ucla general circulation model. In *General circulation models of the atmosphere* (pp. 173–265, Vol. 17). Elsevier. https://doi.org/https://doi.org/10.1016/B978-0-12-460817-7.50009-4 (cit. on pp. 4, 9, 60).

Barros, S., Dent, D., Isaksen, L., Robinson, G., Mozdzynski, G., & Wollenweber, F. (1995). The ifs model: A parallel production weather code. *Parallel Computing*, *21*(10), 1621–1638. https://doi.org/https://doi.org/10.1016/0167-8191(96)80002-0 (cit. on p. 3).

Carpenter, R. L., Droegemeier, K. K., Woodward, P. R., & Hane, C. E. (1990). Application of the piecewise parabolic method (ppm) to meteorological modeling. *Monthly Weather Review*, *118*(3), 586–612. https://doi.org/10.1175/1520-0493(1990)118<0586:AOTPPM>2.0.CO;2 (cit. on pp. 4, 8, 19).

Chen, C., & Xiao, F. (2008). Shallow water model on cubed-sphere by multi-moment finite volume method. *Journal of Computational Physics*, *227*(10), 5019–5044. https://doi.org/https://doi.org/10.1016/j.jcp.2008.01.033 (cit. on p. 84).

Chen, X. (2021). The lmars based shallow-water dynamical core on generic gnomonic cubed-sphere geometry [e2020MS002280 2020MS002280]. *Journal of Advances in Modeling Earth Systems*, *13*(1), e2020MS002280. https://doi.org/https://doi.org/10.1029/2020MS002280 (cit. on pp. 52, 56–58, 62, 65, 67, 84).

Chen, Y., Weller, H., Pring, S., & Shaw, J. (2017). Comparison of dimensionally split and multi-dimensional atmospheric transport schemes for long time steps. *Quarterly Journal of the Royal Meteorological Society*, *143*(708), 2764–2779. https://doi.org/https://doi.org/10.1002/qj.3125 (cit. on pp. 23, 29, 45).

Colella, P., & Woodward, P. R. (1984). The piecewise parabolic method (ppm) for gas-dynamical simulations. *Journal of Computational Physics*, *54*(1), 174–201. https://doi.org/https://doi.org/10.1016/0021-9991(84)90143-8 (cit. on pp. 4, 7, 19–23, 28).

Cooley, J. W., & Tukey, J. W. (1965). An algorithm for the machine calculation of complex fourier series. *Mathematics of Computation*, *19*(90), 297–301. http://www.jstor.org/stable/2003354 (cit. on p. 2).

Courant, R., & John, F. (1999). In *Introduction to calculus and analysis i*. Springer Berlin, Heidelberg. https://doi.org/https://doi.org/10.1007/978-3-642-58604-0 (cit. on p. 104).

Croisille, J.-P. (2013). Hermitian compact interpolation on the cubed-sphere grid. *Journal of Scientific Computing*, *57*. https://doi.org/10.1007/s10915-013-9702-3 (cit. on pp. 52, 62).

Csomós, P., Faragó, I., & Havasi, Á. (2005). Weighted sequential splittings and their analysis [Numerical Methods and Computational Mechanics]. *Computers and Mathematics with Applications*, *50*(7), 1017–1031. https://doi.org/https://doi.org/10.1016/j.camwa.2005.08.004 (cit. on p. 38).

Dennis, J., Edwards, J., Evans, K., Guba, O., Lauritzen, P., Mirin, A., St-Cyr, A., Taylor, M., & Worley, P. (2012). Cam-se: A scalable spectral element dynamical core for the community atmosphere model. *Internat. J. High Perf. Comput. Appl.*, *26*, 74–89. https://doi.org/10.1177/1094342011428142 (cit. on p. 4).

Durran, D. (2011). Time discretization: Some basic approaches. In *Numerical techniques for global atmospheric models* (pp. 75–104). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-11640-7_5 (cit. on p. 17).

Durran, D. R. (2010). Semi-lagrangian methods. In *Numerical methods for fluid dynamics: With applications to geophysics* (pp. 357–391). Springer New York. https://doi.org/10.1007/978-1-4419-6412-0_7 (cit. on p. 18).

Eliasen, E., Machenhauer, B., & Rasmussen, E. (1970). On a numerical method for integration of the hydrodynamical equations wirh a spectral representation of the horizontal fields. https://doi.org/10.13140/RG.2.2.13894.88645 (cit. on p. 2).

Engwirda, D., & Kelley, M. (2016). A weno-type slope-limiter for a family of piecewise polynomial methods. https://doi.org/10.48550/ARXIV.1606.08188 (cit. on pp. 8, 10, 20).

Figueroa, S., Bonatti, J., Kubota, P., Grell, G., Morrison, H., R. M. Barros, S., Fernandez, J., Ramirez-Gutierrez, E., Siqueira, L., Luzia, G., Silva, J., Silva, J., Pendharkar, J., Capistrano, V., Alvim, D., Enore, D., Diniz, F., Satyamurty, P., Cavalcanti, I., & Panetta, J. (2016). The brazilian global atmospheric model (bam): Performance for tropical rainfall forecasting and sensitivity to convective scheme and horizontal resolution. *Weather Forecast.*, *31(5)*, 1547–1572. https://doi.org/10.1175/WAF-D-16-0062.1 (cit. on p. 3).

Folland, G. B. (1999). In *Real analysis: Modern techniques and their applications*. Wiley. (Cit. on p. 104).

Giraldo, F. X., Kelly, J. F., & Constantinescu, E. M. (2013). Implicit-explicit formulations of a three-dimensional nonhydrostatic unified model of the atmosphere (numa). *SIAM Journal on Scientific Computing*, *35*(5), B1162–B1194. https://doi.org/10.1137/120876034 (cit. on p. 4).

Godunov, S. (1959). A difference method for numerical calculation of discontinuous solutions of the equations of hydrodynamics. *Mat. Sb.*, *47(89):3*, 271–306 (cit. on pp. 7, 8, 20).

Guo, W., Nair, R. D., & Qiu, J.-M. (2014). A conservative semi-lagrangian discontinuous galerkin scheme on the cubed sphere. *Monthly Weather Review*, *142*(1), 457–475. https://doi.org/10.1175/MWR-D-13-00048.1 (cit. on pp. 17, 60).

Harris, L., Chen, X., Putman, W., Zhou, L., & Chen, J.-H. (2021). A scientific description of the gfdl finite-volume cubed-sphere dynamical core. *Series : NOAA technical memorandum OAR GFDL ; 2021-001*. https://doi.org/10.25923/6nhs-5897 (cit. on pp. 7, 8, 22, 24, 58).

Harris, L., Zhou, L., Lin, S.-J., Chen, J.-H., Chen, X., Gao, K., Morin, M., Rees, S., Sun, Y., Tong, M., Xiang, B., Bender, M., Benson, R., Cheng, K.-Y., Clark, S., Elbert, O. D., Hazelton, A., Huff, J. J., Kaltenbaugh, A., … Stern, W. (2020). Gfdl shield: A unified system for weather-to-seasonal prediction [e2020MS002223 2020MS002223]. *Journal of Advances in Modeling Earth Systems*, *12*(10), e2020MS002223. https://doi.org/https://doi.org/10.1029/2020MS002223 (cit. on p. 4).

Harris, L. M., & Lin, S.-J. (2013). A two-way nested global-regional dynamical core on the cubed-sphere grid. *Monthly Weather Review*, *141*(1), 283–306. https://doi.org/10.1175/MWR-D-11-00201.1 (cit. on pp. 4, 5, 7).

Harris, L. M., Lin, S.-J., & Tu, C. (2016). High-resolution climate simulations using gfdl hiram with a stretched global grid. *Journal of Climate*, *29*(11), 4293–4314. https://doi.org/10.1175/JCLI-D-15-0389.1 (cit. on pp. 57, 65).

Herzfeld, M., & Engwirda, D. (2023). A flux-form semi-lagrangian advection scheme for tracer transport on arbitrary meshes. *Ocean Modelling*, *181*, 102140. https://doi.org/https://doi.org/10.1016/j.ocemod.2022.102140 (cit. on p. 51).

Holden, H., Karlsen, K., Lie, K.-A., & Risebro, H. (2010). *Splitting methods for partial differential equations with rough solutions: Analysis and matlab programs*. https://doi.org/10.4171/078 (cit. on pp. 38, 43).

Jia, H., & Li, K. (2011). A third accurate operator splitting method. *Mathematical and Computer Modelling*, *53*(1), 387–396. https://doi.org/https://doi.org/10.1016/j.mcm.2010.09.005 (cit. on p. 38).

Jung, J.-H., Konor, C. S., & Randall, D. (2019). Implementation of the vector vorticity dynamical core on cubed sphere for use in the quasi-3-d multiscale modeling framework. *Journal of Advances in Modeling Earth Systems*, *11*(3), 560–577. https://doi.org/https://doi.org/10.1029/2018MS001517 (cit. on p. 54).

Katta, K. K., Nair, R. D., & Kumar, V. (2015a). High-order finite volume shallow water model on the cubed-sphere: 1d reconstruction scheme. *Applied Mathematics and Computation*, *266*, 316–327. https://doi.org/https://doi.org/10.1016/j.amc.2015.04.053 (cit. on pp. 52, 60, 62).

Katta, K. K., Nair, R. D., & Kumar, V. (2015b). High-order finite-volume transport on the cubed sphere: Comparison between 1d and 2d reconstruction schemes. *Monthly Weather Review*, *143*(7), 2937–2954. https://doi.org/https://doi.org/10.1175/MWR-D-13-00176.1 (cit. on pp. 52, 60, 62).

Kent, J., Melvin, T., & Wimmer, G. A. (2023). A mixed finite-element discretisation of the shallow-water equations. *Geoscientific Model Development*, *16*(4), 1265–1276. https://doi.org/10.5194/gmd-16-1265-2023 (cit. on p. 4).

Krishnamurti, T., Hardiker, V., Bedi, H., & Ramaswamy, L. (2006). *An introduction to global spectral modeling* (Vol. 35). https://doi.org/10.1007/0-387-32962-5 (cit. on p. 2).

Lauritzen, P. H., Nair, R. D., & Ullrich, P. A. (2010). A conservative semi-lagrangian multi-tracer transport scheme (cslam) on the cubed-sphere grid. *Journal of Computational Physics*, *229*(5), 1401–1424. https://doi.org/https://doi.org/10.1016/j.jcp.2009.10.036 (cit. on p. 29).

Lauritzen, P. H., Ullrich, P. A., & Nair, R. D. (2011). Atmospheric transport schemes: Desirable properties and a semi-lagrangian view on finite-volume discretizations. In P. Lauritzen, C. Jablonowski, M. Taylor, & R. Nair (Eds.), *Numerical techniques for global atmospheric models* (pp. 185–250). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-11640-7_8 (cit. on p. 8).

Lauritzen, P. H. (2007). A stability analysis of finite-volume advection schemes permitting long time steps. *Monthly Weather Review*, *135*(7), 2658–2673. https://doi.org/https://doi.org/10.1175/MWR3425.1 (cit. on pp. 111, 113).

Leonard, B. P., Lock, A. P., & MacVean, M. K. (1996). Conservative explicit unrestricted-time-step multidimensional constancy-preserving advection schemes. *Monthly Weather Review*, *124*(11), 2588–2606. https://doi.org/https://doi.org/10.1175/1520-0493(1996)124<2588:CEUTSM>2.0.CO;2 (cit. on p. 7).

LeVeque, R. J. (1985). A large time step generalization of godunov's method for systems of conservation laws. *SIAM Journal on Numerical Analysis*, *22*(6), 1051–1073. https://doi.org/10.1137/0722063 (cit. on p. 7).

LeVeque, R. J. (1990). *Numerical methods for conservation laws*. Birkhäuser Basel. https://doi.org/10.1007/978-3-0348-5116-9 (cit. on pp. 11, 44).

LeVeque, R. J. (2002). *Finite volume methods for hyperbolic problems*. Cambridge University Press. https://doi.org/10.1017/CBO9780511791253 (cit. on pp. 11, 20, 107, 109, 110).

Lin, S.-J. (2004). A "vertically lagrangian" finite-volume dynamical core for global models. *Monthly Weather Review*, *132*(10), 2293–2307. https://doi.org/10.1175/1520-0493(2004)132<2293:AVLFDC>2.0.CO;2 (cit. on pp. 4, 8, 17, 22, 28, 51, 81, 83).

Lin, S.-J., Chao, W. C., Sud, Y. C., & Walker, G. K. (1994). A class of the van leer-type transport schemes and its application to the moisture transport in a general circulation model. *Monthly Weather Review*, *122*(7), 1575–1593. https://doi.org/10.1175/1520-0493(1994)122<1575:ACOTVL>2.0.CO;2 (cit. on pp. 4, 83).

Lin, S.-J., Harris, L. M., & Putman, W. M. (2017). *FV3: The GFDL finite-volume cubed-sphere dynamical core*. Retrieved January 13, 2024, from https://www.gfdl.noaa.gov/wp-content/uploads/2020/02/FV3-Technical-Description.pdf (cit. on p. 22).

Lin, S.-J., & Rood, R. B. (1996). Multidimensional flux-form semi-lagrangian transport schemes. *Monthly Weather Review*, *124*(9), 2046–2070. https://doi.org/10.1175/1520-0493(1996)124<2046:MFFSLT>2.0.CO;2 (cit. on pp. 4, 7, 17, 19, 23, 29, 37, 38, 43, 48, 51, 52, 113).

Lin, S.-J., & Rood, R. B. (1997). An explicit flux-form semi-lagrangian shallow-water model on the sphere. *Quarterly Journal of the Royal Meteorological Society*, *123*(544), 2477–2498. https://doi.org/https://doi.org/10.1002/qj.49712354416 (cit. on pp. 4, 7, 17).

Lu, F., Zhang, F., Wang, T., Tian, G., & Wu, F. (2022). High-order semi-lagrangian schemes for the transport equation on icosahedron spherical grids. *Atmosphere*, *13*(11). https://doi.org/10.3390/atmos13111807 (cit. on p. 17).

Martin, R. V., Eastham, S. D., Bindle, L., Lundgren, E. W., Clune, T. L., Keller, C. A., Downs, W., Zhang, D., Lucchesi, R. A., Sulprizio, M. P., Yantosca, R. M., Li, Y., Estrada, L., Putman, W. M., Auer, B. M., Trayanov, A. L., Pawson, S., & Jacob, D. J. (2022). Improved advection, resolution, performance, and community access in the new generation (version 13) of the high-performance geos-chem global atmospheric chemistry model (gchp). *Geoscientific Model Development*, *15*(23), 8731–8748. https://doi.org/10.5194/gmd-15-8731-2022 (cit. on p. 4).

Mouallem, J., Harris, L., & Chen, X. (2023). Implementation of the novel duo-grid in gfdl's fv3 dynamical core [e2023MS003712 2023MS003712]. *Journal of Advances in Modeling Earth Systems*, *15*(12), e2023MS003712. https://doi.org/https://doi.org/10.1029/2023MS003712 (cit. on pp. 5, 67, 84, 86).

Müller, A., Deconinck, W., Kühnlein, C., Mengaldo, G., Lange, M., Wedi, N., Bauer, P., Smolarkiewicz, P. K., Diamantakis, M., Lock, S.-J., Hamrud, M., Saarinen, S., Mozdzynski, G., Thiemert, D., Glinton, M., Bénard, P., Voitus, F., Colavolpe, C., Marguinaud, P., … New, N. (2019). The escape project: Energy-efficient scalable algorithms for weather prediction at exascale. *Geoscientific Model Development*, *12*(10), 4425–4441. https://doi.org/10.5194/gmd-12-4425-2019 (cit. on p. 3).

Nair, R. D., & Lauritzen, P. H. (2010). A class of deformational flow test cases for linear transport problems on the sphere. *Journal of Computational Physics*, *229*(23), 8868–8887. https://doi.org/https://doi.org/10.1016/j.jcp.2010.08.014 (cit. on pp. 26, 45, 47, 85, 86, 91, 94).

Nair, R. D., Thomas, S. J., & Loft, R. D. (2005). A discontinuous galerkin transport scheme on the cubed sphere. *Monthly Weather Review*, *133*(4), 814–828. https://doi.org/10.1175/MWR2890.1 (cit. on p. 60).

Orszag, S. A. (1970). Transform method for the calculation of vector-coupled sums: Application to the spectral form of the vorticity equation. *Journal of Atmospheric Sciences*, *27*(6), 890–895. https://doi.org/10.1175/1520-0469(1970)027<0890:TMFTCO>2.0.CO;2 (cit. on p. 2).

Peixoto, P. (2016). Accuracy analysis of mimetic finite volume operators on geodesic grids and a consistent alternative. *J. Comput. Phys.*, *310*, 127–160. https://doi.org/10.1016/j.jcp.2015.12.058 (cit. on pp. 5, 51).

Peixoto, P., & Barros, S. R. M. (2013). Analysis of grid imprinting on geodesic spherical icosahedral grids. *J. Comput. Phys.*, *237*, 61–78. https://doi.org/10.1016/j.jcp.2012.11.041 (cit. on pp. 5, 51).

Peixoto, P. S., & Barros, S. R. (2014). On vector field reconstructions for semi-lagrangian transport methods on geodesic staggered grids. *Journal of Computational Physics*, *273*, 185–211 (cit. on p. 51).

Putman, W. M. (2007). *Development of the finite-volume dynamical core on the cubed-sphere* [Doctoral dissertation, Florida State University]. Florida, US. http://purl.flvc.org/fsu/fd/FSU_migr_etd-0511 (cit. on pp. 4, 7).

Putman, W. M., & Lin, S.-J. (2007). Finite-volume transport on various cubed-sphere grids. *Journal of Computational Physics*, *227*(1), 55–78. https://doi.org/https://doi.org/10.1016/j.jcp.2007.07.022 (cit. on pp. 4, 5, 8, 17, 43, 48, 51, 52, 65, 66, 72, 74, 81–83).

Rančić, M., Purser, R. J., & Mesinger, F. (1996). A global shallow-water model using an expanded spherical cube: Gnomonic versus conformal coordinates. *Quarterly Journal of the Royal Meteorological Society*, *122*(532), 959–982. https://doi.org/https://doi.org/10.1002/qj.49712253209 (cit. on pp. 51, 52).

Rančić, M. (1992). Semi-lagrangian piecewise biparabolic scheme for two-dimensional horizontal advection of a passive scalar. *Monthly Weather Review*, *120*(7), 1394–1406. https://doi.org/10.1175/1520-0493(1992)120<1394:SLPBSF>2.0.CO;2 (cit. on p. 29).

Rančić, M., Purser, R. J., Jović, D., Vasic, R., & Black, T. (2017). A nonhydrostatic multiscale model on the uniform jacobian cubed sphere. *Monthly Weather Review*, *145*(3), 1083–1105. https://doi.org/10.1175/MWR-D-16-0178.1 (cit. on pp. 5, 52).

Randall, D. (2022). *An introduction to numerical modeling of the atmosphere*. (Cit. on p. 1).

Randall, D. A., Bitz, C. M., Danabasoglu, G., Denning, A. S., Gent, P. R., Gettelman, A., Griffies, S. M., Lynch, P., Morrison, H., Pincus, R., & Thuburn, J. (2018). 100 years of earth system model development. *Meteorological Monographs*, *59*, 12.1–12.66. https://doi.org/10.1175/AMSMONOGRAPHS-D-18-0018.1 (cit. on pp. 1, 3).

Richtmyer, R. D., & Morton, K. W. (1968). Difference methods for initial-value problems. *SIAM Review*, *10*(3), 381–383. https://doi.org/10.1137/1010073 (cit. on p. 37).

Ringler, T., Thuburn, J., Klemp, J., & Skamarock, W. (2010). A unified approach to energy conservation and potential vorticity dynamics on arbitrarily structured C-grids. *J. Comput. Phys.*, *229*, 3065–3090. https://doi.org/10.1016/j.jcp.2009.12.007 (cit. on p. 5).

Ronchi, C., Iacono, R., & Paolucci, P. (1996). The “cubed sphere”: A new method for the solution of partial differential equations in spherical geometry. *Journal of Computational Physics*, *124*(1), 93–114. https://doi.org/https://doi.org/10.1006/jcph.1996.0047 (cit. on pp. 4, 51, 52, 56, 62).

Rossmanith, J. A. (2006). A wave propagation method for hyperbolic systems on the sphere. *Journal of Computational Physics*, *213*(2), 629–658. https://doi.org/https://doi.org/10.1016/j.jcp.2005.08.027 (cit. on pp. 62, 84, 85).

Sadourny, R. (1972). Conservative finite-difference approximations of the primitive equations on quasi-uniform spherical grids. *Monthly Weather Review*, *100*(2), 136–144. https://doi.org/10.1175/1520-0493(1972)100<0136:CFAOTP>2.3.CO;2 (cit. on pp. 4, 51, 56, 72).

Skamarock, W., Klemp, J., Duda, M., Fowler, L., Park, S.-H., & Ringler, T. (2012). A multiscale nonhydrostatic atmospheric model using centroidal Voronoi tesselations and C-grid staggering. *Mon. Weather. Rev.*, *140(09)*, 3090–3105. https://doi.org/10.1175/MWR-D-11-00215.1 (cit. on p. 5).

Skamarock, W. C., & Gassmann, A. (2011). Conservative transport schemes for spherical geodesic grids: High-order flux operators for ode-based time integration. *Mon. Weather. Rev.*, *139*(9), 2962–2975. https://doi.org/10.1175/MWR-D-10-05056.1 (cit. on p. 51).

Staniforth, A., & Thuburn, J. (2012). Horizontal grids for global weather and climate prediction models: A review. *Q. J. Roy. Meteor. Soc.*, *138*, 1–26. https://doi.org/10.1002/qj.958 (cit. on p. 3).

Staniforth, A., & Wood, N. (2008). Aspects of the dynamical core of a nonhydrostatic, deep-atmosphere, unified weather and climate-prediction model [Predicting weather, climate and extreme events]. *Journal of Computational Physics*, *227*(7), 3445–3464. https://doi.org/https://doi.org/10.1016/j.jcp.2006.11.009 (cit. on p. 1).

Stoer, J., & Bulirsch, R. (2002). In *Introduction to numerical analysis*. Springer New York, NY. https://doi.org/https://doi.org/10.1007/978-0-387-21738-3 (cit. on pp. 19, 103).

Strang, G. (1968). On the construction and comparison of difference schemes. *SIAM Journal on Numerical Analysis*, *5*(3), 506–517. https://doi.org/10.1137/0705041 (cit. on p. 38).

Strikwerda, J. C. (2004). *Finite difference schemes and partial differential equations, second edition*. Society for Industrial; Applied Mathematics. https://doi.org/10.1137/1.9780898717938 (cit. on p. 111).

Thuburn, J. (2011). Conservation in dynamical cores: What, how and why? In *Numerical techniques for global atmospheric models* (pp. 345–355). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-11640-7_11 (cit. on p. 3).

Thuburn, J., Ringler, T., Skamarock, W., & Klemp, J. (2009). Numerical representation of geostrophic modes on arbitrarily structured C-grids. *J. Comput. Phys.*, *228*, 8321–8335. https://doi.org/10.1016/j.jcp.2009.08.006 (cit. on p. 5).

Trefethen, L. N. (2000). *Spectral methods in matlab*. Society for Industrial; Applied Mathematics. https://doi.org/10.1137/1.9780898719598 (cit. on pp. 26, 110).

Tumolo, G. (2011). *A semi-implicit, semi-lagrangian, p-adaptative discontinuous galerkin method for the rotating shallow-water equations: Analysis and numerical experiments* [Doctoral dissertation, University of Trieste]. https://core.ac.uk/download/pdf/41173373.pdf (cit. on p. 17).

Ullrich, P. A., Jablonowski, C., Kent, J., Lauritzen, P. H., Nair, R., Reed, K. A., Zarzycki, C. M., Hall, D. M., Dazlich, D., Heikes, R., Konor, C., Randall, D., Dubos, T., Meurdesoif, Y., Chen, X., Harris, L., Kühnlein, C., Lee, V., Qaddouri, A., … Viner, K. (2017). Dcmip2016: A review of non-hydrostatic dynamical core design and intercomparison of participating models. *Geoscientific Model Development*, *10*(12), 4477–4509. https://doi.org/10.5194/gmd-10-4477-2017 (cit. on p. 3).

Ullrich, P. A., Jablonowski, C., & van Leer, B. (2010). High-order finite-volume methods for the shallow-water equations on the sphere. *Journal of Computational Physics*, *229*(17), 6104–6134. https://doi.org/https://doi.org/10.1016/j.jcp.2010.04.044 (cit. on p. 60).

Van Leer, B. (1977). Towards the ultimate conservative difference scheme. iv. a new approach to numerical convection. *Journal of Computational Physics*, *23*(3), 276–299. https://doi.org/https://doi.org/10.1016/0021-9991(77)90095-X (cit. on pp. 4, 8, 19, 20).

Weller, H. (2012). Controlling the computational modes of the arbitrarily structured c grid, *Mon. Weather. Rev.*, *140(10)*, 3220–3234. https://doi.org/doi.org/10.1175/MWR-D-11-00221.1 (cit. on pp. 5, 51).

Wesseling, P. (2001). Scalar conservation laws. In *Principles of computational fluid dynamics* (pp. 339–396). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-05146-3_9 (cit. on p. 8).

Whitaker, J. (2015). *Hiwpp non-hydrostatic dynamical core tests: Results from idealized test cases*. Retrieved November 5, 2022, from https://www.weather.gov/media/sti/nggps/HIWPP_idealized_tests-v8%20revised%2005212015.pdf/ (cit. on p. 5).

White, L., & Adcroft, A. (2008). A high-order finite volume remapping scheme for nonuniform grids: The piecewise quartic method (pqm). *Journal of Computational Physics*, *227*(15), 7394–7422. https://doi.org/https://doi.org/10.1016/j.jcp.2008.04.026 (cit. on p. 8).

Williamson, D., Drake, J., Hack, J., Jakob, R., & Swarztrauber, P. (1992). A standard test set for numerical approximations to the shallow water equations in spherical geometry. *J. Comput. Phys.*, *102*, 211–224. https://doi.org/10.1016/S0021-9991(05)80016-6 (cit. on pp. 5, 68, 71, 85).

Williamson, D. L. (2007). The evolution of dynamical cores for global atmospheric models. *Journal of the Meteorological Society of Japan. Ser. II*, *85B*, 241–269. https://doi.org/10.2151/jmsj.85B.241 (cit. on pp. 1, 2).

Wilson, R. J., Kahre, M. A., Urata, R. A., Brecht, A. S., Kling, A., Batterson, C. M., Steakley, K., Hartwick, V., & Harman, C. E. (2022). The New NASA Ames Mars Global Climate Model: Comparing a Reference Simulation to MRO/MCS Temperatures. *AGU Fall Meeting Abstracts*, *2022*, Article P42D-2439, P42D–2439 (cit. on p. 4).

Wood, N., Staniforth, A., White, A., Allen, T., Diamantakis, M., Gross, M., Melvin, T., Smith, C., Vosper, S., Zerroukat, M., & Thuburn, J. (2014). An inherently mass-conserving semi-implicit semi-lagrangian discretization of the deep-atmosphere global non-hydrostatic equations. *Quarterly Journal of the Royal Meteorological Society*, *140*(682), 1505–1520. https://doi.org/https://doi.org/10.1002/qj.2235 (cit. on p. 1).

Woodward, P. R. (1986). Piecewise-parabolic methods for astrophysical fluid dynamics. In K.-H. A. Winkler & M. L. Norman (Eds.), *Astrophysical radiation hydrodynamics* (pp. 245–326). Springer Netherlands. https://doi.org/10.1007/978-94-009-4754-2_8 (cit. on p. 8).

Zerroukat, M., & Allen, T. (2022). On the corners of the cubed-sphere grid. *Quarterly Journal of the Royal Meteorological Society*, *148*(743), 778–783. https://doi.org/https://doi.org/10.1002/qj.4230 (cit. on p. 67).

Zheng, Y., & Marguinaud, P. (2018). Simulation of the performance and scalability of message passing interface (mpi) communications of atmospheric models running on exascale supercomputers. *Geoscientific Model Development*, *11*(8), 3409–3426. https://doi.org/10.5194/gmd-11-3409-2018 (cit. on p. 3).

Zhou, L., Lin, S.-J., Chen, J.-H., Harris, L. M., Chen, X., & Rees, S. L. (2019). Toward convective-scale prediction within the next generation global prediction system. *Bulletin of the American Meteorological Society*, *100*(7), 1225–1243. https://doi.org/10.1175/BAMS-D-17-0246.1 (cit. on p. 58).