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

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. Dr. Pedro da Silva Peixoto

São Paulo

May, 2024

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

Luan da Fonseca Santos

This version of the thesis includes the corrections and modifications suggested by the Examining Committee during the defense of the original version of the work, which took place on May 14, 2024.

A copy of the original version is available at the Institute of Mathematics and Statistics of the University of São Paulo.

Examining Committee:

Prof. Dr. Pedro da Silva Peixoto (advisor) – IME-USP

Prof. Dr. Fabrício Simeoni de Sousa – ICMC-USP

Prof. Dr. Pedro Leite da Silva Dias – IAG-USP

Prof. Dr. Maicon Ribeiro Correa – IMECC-Unicamp

Dr. Joseph Mouallem – Princeton University-NOAA GFDL

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

Acknowledgements

I would like to thank my lovely wife, Naira. She constantly supported me during this work, day in and day out. I am also grateful to her for bringing Nino and Lupita into my life, and I am very happy to share life with her. I love you!

Many thanks to my advisor, Prof. Dr. Pedro Peixoto, for the excellent supervision he provided me from my final year of undergraduate studies until the end of my doctoral studies. I will always be grateful for all the knowledge he kindly shared with me and he will always be a reference to me.

I would like to thank Dr. Joseph Mouallem for his valuable help every time I had a problem with the FV3 code, as well as for the insightful discussions we had and for the kind reception during my visit to the AOS-Princeton/GFDL. Additionally, I extend my thanks to Dr. Lucas Harris for providing his valuable feedback about my work.

I would like to thank Prof. Dr. Saulo Barros (*in memoriam*), with whom I had the pleasure of taking a great course in numerical PDEs and being his teaching assistant. Prof. Saulo was very important for my career development, and he will always be a reference to me as well.

I would like to thank my parents, Jeane and Reinaldo, for all the support they gave to keep me going with my studies, as well as my sister Bruna, my niece Helena and also Nick. Additionally, I would like to express my gratitude to my mother-in-law, Adriana, and also my father-in-law Marcos and his wife Denise for all the support they provided to my wife and me. A special thanks also goes to my wife's grandmother, Dona Cida, for the warm reception we received every time we visited her.

Finally, I am thankful for the financial support of FAPESP and Capes.

Resumo

Luan da Fonseca Santos. **Análise de esquemas de volumes finitos para advecção em esferas cubadas e uma alternativa precisa para ventos divergentes**. Tese (Doutorado). Instituto de Matemática e Estatística, Universidade de São Paulo, São Paulo, 2024.

O núcleo dinâmico de volumes finitos do GFDL-NOAA-EUA, originalmente projetado para grades de latitude e longitude, foi adaptado à esfera cubada para melhorar a escalabilidade em supercomputadores massivamente paralelos, resultando na criação do núcleo dinâmico FV3. O FV3 serve como núcleo dinâmico para muitos modelos globais e, em 2019, foi selecionado como o núcleo dinâmico oficial para o novo Sistema Global de Previsão do Serviço Nacional de Meteorologia dos EUA, substituindo o modelo espectral. A abordagem de volume finitos do FV3 para resolver a dinâmica horizontal consiste na aplicação de fluxos de advecção para diversas variáveis; assim, o esquema de advecção desempenha um papel fundamental no modelo. Portanto, nesta tese, propomos investigar os detalhes do esquema de advecção do FV3. Conseguimos sugerir modificações no esquema de advecção do FV3 que melhoraram significativamente a advecção para ventos divergentes com apenas um pequeno esforço computacional adicional e pequenas mudanças no código existente do FV3. Realizamos várias simulações numéricas usando as equações de advecção e águas rasas. Como o esquema de advecção do FV3 consiste na combinação de operadores de fluxo de volume finitos 1D, nossas melhorias foram obtidas ao melhorar o cálculo do ponto de partida para os operadores de fluxo 1D e modificar a forma como o termo métrico da esfera cubada é tratado ao calcular os fluxos 1D. Através de simulações, demonstramos que o esquema de advecção atual do FV3 é apenas de primeira ordem para ventos divergentes, enquanto nosso esquema é de segunda ordem. Para ventos livres de divergência, ambos os esquemas são de segunda ordem, sendo o nosso esquema ligeiramente mais preciso. Uma grande dificuldade em trabalhar na esfera cubada é lidar com a descontinuidade das coordenadas ao longo das faces do cubo, o que pode levar a erros maiores nessas regiões. No entanto, demonstramos através de simulações numéricas que o esquema de advecção proposto apresenta uma sensibilidade ligeiramente reduzida aos cantos do cubo. Em resumo, esta tese oferece uma análise abrangente da discretização do FV3 da dinâmica horizontal, fornecendo uma valiosa compreensão para aprimorar a precisão do núcleo dinâmico do FV3, especialmente para ventos divergentes.

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

Abstract

Luan da Fonseca Santos. **Analysis of finite-volume advection schemes on cubed-sphere grids and an accurate alternative for divergent winds**. Thesis (Doctorate). Institute of Mathematics and Statistics, University of São Paulo, São Paulo, 2024.

The finite-volume dynamical core developed by GFDL-NOAA-USA, originally designed for latitude-longitude grids, was adapted to the cubed-sphere to enhance scalability on massively parallel supercomputers, resulting in the creation of the FV3 dynamical core. FV3 serves as the dynamical core for many models worldwide, and in 2019, it was officially designated as the dynamical core for the new Global Forecast System of the National Weather Service in the USA, replacing the spectral model. The finite-volume approach employed by FV3 to solve horizontal dynamics involves applying advection fluxes for different variables; thus, the advection scheme plays a key role in the model. Therefore, in this thesis, we propose to investigate the details of the advection scheme of FV3. We were able to suggest modifications to the FV3 advection scheme that significantly improved advection for divergent winds with only a small extra computational effort and of simple implementation in the existing codes. We conducted several numerical simulations using the advection and shallow-water equations. Since the FV3 advection scheme utilizes a splitting strategy, combining 1D finite-volume flux operators, our improvements were obtained by improving the departure point computation for the 1D flux operators and modifying the way the metric term of the cubed-sphere is treated when computing the 1D fluxes. Through simulations, we demonstrate that the current FV3 advection scheme is only first-order accurate for divergent winds, while our scheme is second-order accurate. For divergence-free winds, both schemes are second-order, with our scheme being slightly more accurate. One major difficulty in working on the cubed-sphere is handling coordinate discontinuity along the cube faces, which may lead to larger errors in these regions. However, we demonstrate through numerical simulations that the proposed advection scheme exhibits slightly reduced sensitivity to the cube corners. In conclusion, this thesis offers a comprehensive examination of the FV3 discretization of horizontal dynamics, providing valuable insights into enhancing the accuracy of the FV3 dynamical core, particularly for divergent winds.

Keywords: Dynamical core, cubed-sphere, finite-volume, dimension splitting, departure point, mass fixer, advection equation, shallow-water equations.

List of abbreviations and acronyms

CFL	Courant–Friedrichs–Lewy
CS	Cubed-sphere
dd	Divergence damping
dg1	Lagrange interpolation on duo-grid based on geodesic distances
dg2	Lagrange interpolation on duo-grid based on the local coordinate distances
DP1	First-order departure point
DP2	Second-order departure point
dycore	Dynamical core
ENDGame	Even Newer Dynamics for General atmospheric modelling of the environment
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
IC	Initial Condition
LT	Average Lie-Trotter advection scheme
MONO	Monotonic PPM reconstruction scheme
MPI	Message Passing Interface
NOAA	National Oceanic and Atmospheric Administration
ODE	Ordinary Differential Equation
PDE	Partial Differential Equation
PL	Putman and Lin advection scheme
PPM	Piecewise Parabolic method

SHiELD System for High-resolution prediction on Earth-to-Local Domains

SL Semi-Lagrangian

SWE Shallow-water equations

TC Test case

VF Velocity field

UNLIM Unlimited PPM reconstruction scheme

Contents

Appendixes

Chapter 1

Introduction

1.1 Historical 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 1950s under the Joint Numerical Weather Prediction Unit (JNWPU), a joint project by the U.S. Air Force, Navy, and Weather Bureau, with applications to weather, seasonal, and climate forecasts (Harper et al., 2007). 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 (dycore), which is responsible for solving the PDEs that govern the atmosphere dynamics on grid-scale (Williamson, 2007). The development of numerical methods for dynamical cores has been an active research area since the 1950s.

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 (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 (Randall, 2022). The latitude-longitude grid is highly non-isotropic due to the pole problem, with a much higher resolution at the poles than elsewhere (Figure 1.1a). Additionally, polar filters may be required by explicit schemes to stabilize short waves that tend to be over-resolved at high latitudes, which may degrade the solution accuracy (Williamson, 2007).

The most successful method adopted in global atmospheric dynamical cores on latitude-

longitude grids that overcomes the CFL restriction is the Semi-Implicit Semi-Lagrangian (SI-SL) scheme (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, traditional Semi-Lagrangian schemes are inherently non-conservatives for mass, which is critical for climate forecasts (Williamson, 2007).

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

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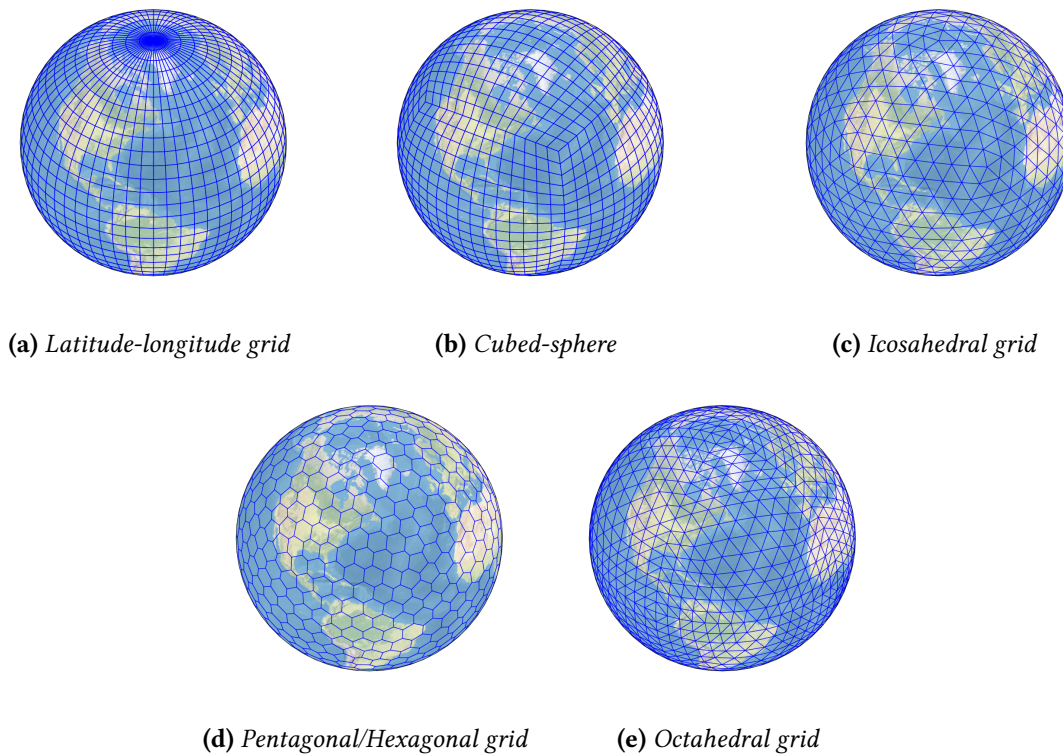
Figure 1.1: *Examples of spherical grids: latitude-longitude grid (a) and grids based on Platonic solids (b)-(d).*

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 timessteps allowed in this case. Due to these enhancements, the spectral transform dominated global atmospheric modeling (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), 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)]. The Global Forecast System (GFS) from the National Centers for Environmental Prediction (NCEP) employed the spectral method until June 2019 when it was replaced by the Finite Volume Cubed-Sphere dynamical core (FV3) from the Geophysical Fluid Dynamics Laboratory (GFDL) and the National Oceanic and Atmospheric Administration (NOAA) in June 2019 (https://www.noaa.gov/media-release/noaa-upgrades-us-global-weather-forecast-model, last accessed on March 19th, 2024).

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, classical 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 better 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 dynamical core

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) at the National Oceanic and Atmospheric Administration (NOAA) (Harris & Lin, 2013; Putman & Lin, 2007) is another example of new generation dynamical core based on the cubed-sphere.

The FV3 dycore is an extension of the Finite-Volume dycore (FVcore) from latitude-longitude grids to the cubed-sphere. The numerical methods from FVcore started to be developed with the advection 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 advection 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 disadvantage of the FVcore is its Semi-Lagrangian formulation, which introduces parallelization difficulties. Indeed, as noted by Putman and Lin (2007), it creates a load balancing problem, primarily due to a larger halo region needed in the zonal direction near the poles. Then, 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 dycore. Later, the FV3 also was improved to allow locally refinement grids through grid-nesting or grid-stretching (Harris & Lin, 2013). More recently, FV3 introduced two-way multiple same-level and telescoping grid nesting capabilities, as detailed by Mouallem et al. (2022).

Currently, the FV3 dycore is capable of performing both hydrostatic and non-hydrostatic atmospheric simulations. It was chosen as the new US GFS dynamical core, indeed replacing the spectral transform GFS in 2019, as mentioned before. 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 System for High-resolution prediction on Earth-to-Local Domain (SHiELD) model from GFDL (Harris et al., 2020).

However, a well-known problem that occurs on cubed-sphere models 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 refers to the presence of artificial behaviors in the numerical solution that are associated with the grid used. It is important to emphasize that other quasi-uniform grids may also experience grid imprinting, such as hexagonal grids (Peixoto, 2016; Peixoto & Barros, 2013; Weller et al., 2012).

Despite being chosen as the new US global weather prediction model, there is a lack of numerical studies on the FV3 discretizations in the literature. Numerical results for the advection equation on the cubed-sphere using the FV3 dynamical core were presented in Putman and Lin (2007). However, they utilized extrapolations near the cube edges instead of the duo-grid approach from Mouallem et al. (2023), which affects the convergence of this method. The current solver of FV3 solves the shallow-water equation on the so-called Lagrangian surfaces. This shallow-water solver, based on Lin and Rood (1997), utilizes the advection solver from Putman and Lin (2007) to update the pressure, vorticity, and kinetic energy fluxes. Therefore, advection is a key aspect of the FV3 dynamical core, deserving better understanding. In this thesis, we propose to thoroughly examine all the minor details of the scheme from Putman and Lin (2007) and suggest potential improvements, thereby addressing gaps in the existing literature.

1.3   Outline and contributions

This thesis is outlined as follows.

- Chapter 2 is dedicated to reviewing the Piecewise Parabolic Method (PPM) for the one-dimensional advection equation. In this Chapter, we demonstrate how the temporal component of 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. Its benefits are also observed when using the monotonic version of PPM used in FV3 and proposed by Lin (2004). 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 allows us to use one-dimensional methods, such as the PPM, to solve the two-dimensional advection equation. We review the current 2D advection scheme of FV3 on the plane proposed by Putman and Lin (2007). The main feature of this scheme is that it preserves a constant scalar field when the wind is divergence-free. We show through some numerical simulations that this scheme is second-order accurate only for divergence-free winds. When the wind is not divergence-free, we show that this scheme is only first-order accurate. On the other hand, we propose a small modification of the Putman and Lin (2007) scheme using the second-order departure point computation presented in Chapter 2, which allows us to achieve second-order accuracy and smaller errors for both divergent and divergence-free winds. Despite this scheme not preserving a constant scalar for divergence-free winds, it still exhibits second-order error in this case as well. Furthermore, when the monotonic scheme is used in the 1D solver, this scheme also has smaller errors compared to the Putman and Lin (2007) scheme.

- In Chapter 4, we introduce the cubed-sphere grid utilized in FV3, which includes the equi-edge (X. Chen, 2021) and equiangular grids (Ronchi et al., 1996), and we investigate their geometrical properties. Next, we present all the tools necessary to extend the advection schemes on the plane from Chapter 3 to the sphere. We review the contravariant/covariant wind formulation induced by the cubed-sphere mappings. Additionally, we demonstrate how stencils can be computed near the cube-edges through the duo-grid technique to generate the ghost cells required for utilizing 1D Lagrange interpolation to fill these ghost cells.

- Chapter 5 extends the ideas of Chapter 3 to the cubed-sphere grid using the tools from Chapter 4. The dimension-splitting method on each cubed-sphere panel works as in the plane, with the addition of metric terms due to the non-orthogonality of the grid and interpolation between panels to obtain ghost cell values needed for stencil computations. We show that the scheme from Putman and Lin (2007) uses a less accurate formulation of the metric term to preserve the constant scalar for a divergence-free wind, while our new scheme may use a more accurate formulation of the metric term, as it does not have this preservation constraint. The results are essentially the same as those from Chapter 3, showing that our scheme successfully extends from the plane to the cubed-sphere and is more accurate. We also demonstrate that our new scheme has smaller errors at the corners compared to the scheme from Putman and Lin (2007).

- In Chapter 6, we present in detail the shallow-water solver of FV3 based on the extension of the solver by Lin and Rood (1997) to the cubed-sphere. This solver uses finite-volume advection fluxes to compute all operators. Then, we may use our advection scheme introduced in Chapter 5 to compute the fluxes and compare

the results with the scheme by Putman and Lin (2007). We show that our scheme may help to slightly reduce the maximum errors for the geostrophic balanced case. Additionally, we analyze the runtimes and show that our scheme adds a very small extra computational cost.

In summary, the main contribution of this thesis is a modified version of the two-dimensional scheme proposed by Putman and Lin (2007), which exhibits significantly improved accuracy, especially for divergent flows, requiring minor extra computational cost and only minor modifications to the current FV3 code. For divergent free flows, our scheme is slightly better. We give some final thoughts and 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 (cf. 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 using flux advection operators to compute the fluid density, absolute vorticity, and the kinetic energy (Harris & Lin, 2013; Harris et al., 2021; 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, 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), limits 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 the 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 $\mathbb{R}_\nu^N := \mathbb{R}^{N+2\nu}$ and $\mathbb{R}_\nu^{N+1} := \mathbb{R}^{N+1+2\nu}$.

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.*
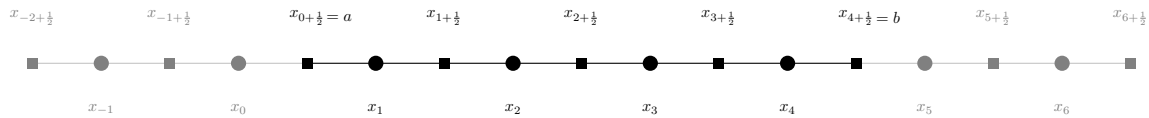

Figure 2.1: *Illustration of a Δx-grid with $N = 4$ cells in its interior (in black) and $\nu = 2$ ghost cell layers (in gray). The edges are denoted by squares and the cell centroids are denoted using circles.*

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 $Q = (Q_{-\nu+1}, \dots, Q_{N+\nu}) \in \mathbb{R}_\nu^N$.*

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

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 $q^n \in \mathbb{R}_\nu^N$ and $u^n \in \mathbb{R}_\nu^{N+1}$. Here, $q_i^n = q(x_i, t^n)$ and $u_{i+\frac{1}{2}}^n = u(x_{i+\frac{1}{2}}, t^n)$. 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:

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

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

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

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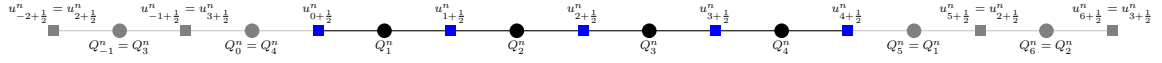
Figure 2.2: *Illustration of Δx-grid function Q (black circles) and a Δx-grid wind u (blue squares) and its ghost cell values (in gray) assuming periodicity.*

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

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

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 $S_{i+\frac{1}{2}} = \{i - r + 1, \dots, i - 1, i, i + 1, \dots, i + s\} \subset \{-\nu + 1, \dots, N + \nu\}$.*

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

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 conservative advection equation with periodic boundary conditions is 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 density (Zerroukat et al., 2006), 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 (2.4) 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.,

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

since we are assuming periodic boundary conditions. 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:

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

where

$$Q_i(t) = \frac{1}{\Delta x} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} q(x, t)\, dx. \tag{2.8}$$

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

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

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:

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

(2.10)

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 (2.10), 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:

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

(2.11)

The point $x_{i+\frac{1}{2}}^d(t^n, s)$ 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.
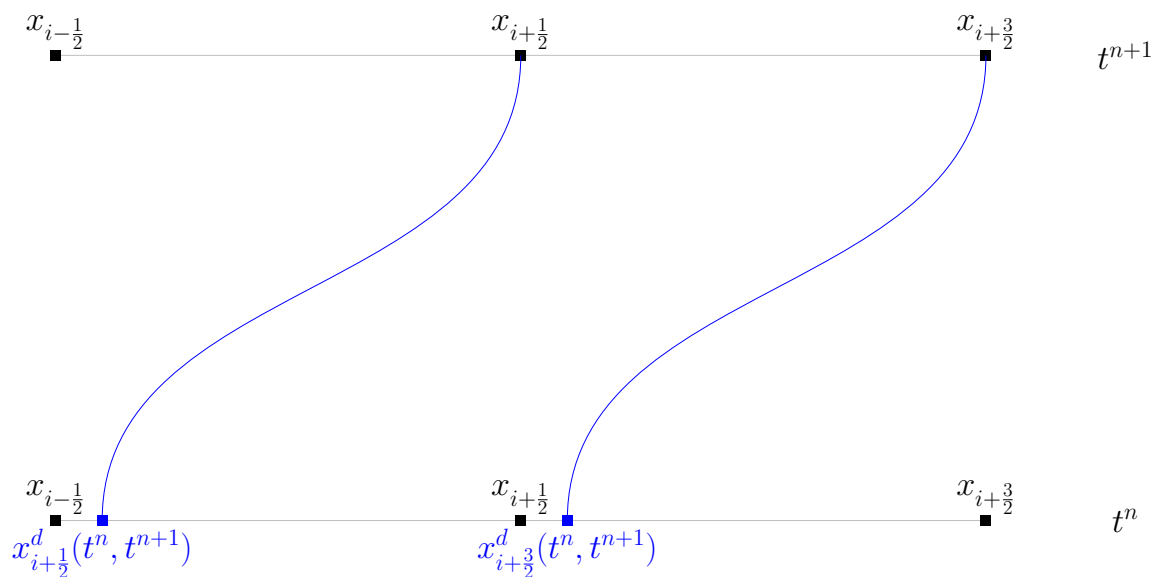

Figure 2.3: *Illustration of the departure point of the cell edges from time t^{n+1} to t^n.*

Integrating Equation (2.11) over the interval $[t, s]$, we get:

$$x_{i+\frac{1}{2}}^d(t, s) = x_{i+\frac{1}{2}} - \int_t^s u(x_{i+\frac{1}{2}}^d(\theta, s), \theta)\, d\theta. \tag{2.12}$$

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:*

$$\int_{t^n}^{t^{n+1}} (uq)(x_{i+\frac{1}{2}}, s)\, ds = \int_{x_{i+\frac{1}{2}}^d(t^n, t^{n+1})}^{x_{i+\frac{1}{2}}} q(x, t^n)\, dx. \tag{2.13}$$

Proof. Using the Leibniz rule for integration (Theorem A.3 with $f(s, \theta) = u\big(x_{i+\frac{1}{2}}^d(\theta, s), \theta\big)$), in Equation (2.12), it follows that:

$$
\begin{aligned}
\partial_s x_{i+\frac{1}{2}}^d(t, s) &= -\left(u(x_{i+\frac{1}{2}}, s) + \int_t^s \partial_s u\big(x_{i+\frac{1}{2}}^d(\theta, s), \theta\big)\, d\theta \right) \\
&= -u(x_{i+\frac{1}{2}}, s) - \int_t^s \partial_x u\big(x_{i+\frac{1}{2}}^d(\theta, s), \theta\big) \partial_s x_{i+\frac{1}{2}}^d(\theta, s)\, d\theta.
\end{aligned}
\tag{2.14}
$$

Taking the derivative with respect to t of Equation (2.14), we have:

$$\partial_t \partial_s x_{i+\frac{1}{2}}^d(t, s) = \partial_x u\big(x_{i+\frac{1}{2}}^d(t, s), t\big) \partial_s x_{i+\frac{1}{2}}^d(t, s). \tag{2.15}$$

Using standard ODE's techniques, we get that $x_{i+\frac{1}{2}}^d$ that solves Equations (2.14) and (2.15) is given by:

$$\partial_s x_{i+\frac{1}{2}}^d(t, s) = -\exp\left(\int_t^s \partial_x u\big(x_{i+\frac{1}{2}}^d(\theta, s), \theta\big)\, d\theta \right) u(x_{i+\frac{1}{2}}, s). \tag{2.16}$$

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

$$
\begin{aligned}
\frac{d}{dt} q\big(x_{i+\frac{1}{2}}^d(t, s), t\big) &= \partial_t q\big(x_{i+\frac{1}{2}}^d(t, s), t\big) + (u\partial_x q)\big(x_{i+\frac{1}{2}}^d(t, s), t\big) \\
&= -\partial_x u\big(x_{i+\frac{1}{2}}^d(t, s), t\big) q\big(x_{i+\frac{1}{2}}^d(t, s), t\big),
\end{aligned}
\tag{2.17}
$$

where we used that q satisfies the linear advection equation on its differential (2.4) form and that $x_{i+\frac{1}{2}}^d(t, s)$ solves Equation (2.11). Using again standard ODE techniques, we get that q that solves Equation (2.17) is given by:

$$q\big(x_{i+\frac{1}{2}}^d(t, s), t\big) = \exp\left(-\int_t^s \partial_x u\big(x_{i+\frac{1}{2}}^d(\theta, s), \theta\big)\, d\theta \right) q(x_{i+\frac{1}{2}}, s). \tag{2.18}$$

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

Let us consider the mapping $s \in [t^n, t^{n+1}] \to x_{i+\frac{1}{2}}^d(t^n, s)$. Integrating q over all departure

points at time t^n from $x_{i+\frac{1}{2}}$ at time s, we have

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

where we are just using the variable change integration formula. Then, it follows from Equations (2.16) and (2.18) with $t = t^n$ that:

$$\int_{x_{i+\frac{1}{2}}}^{x^d_{i+\frac{1}{2}}(t^n,t^{n+1})} q(x,t^n)\,dx = -\int_{t^n}^{t^{n+1}} (uq)(x_{i+\frac{1}{2}},s)\,ds, \tag{2.20}$$

which is the desired formula. $\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\square$

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 2.2. It follows from Proposition 2.1 that:*

$$Q_i(t^{n+1}) = Q_i(t^n) - \lambda\left(\frac{1}{\Delta t}\int_{x^d_{i+\frac{1}{2}}(t^n,t^{n+1})}^{x_{i+\frac{1}{2}}} q(x,t^n)\,dx - \frac{1}{\Delta t}\int_{x^d_{i-\frac{1}{2}}(t^n,t^{n+1})}^{x_{i-\frac{1}{2}}} q(x,t^n)\,dx\right), \tag{2.21}$$

$$\forall i = 1,\dots,N, \quad \forall n = 0,\dots,N_T - 1,$$

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:*

$$Q_i^{n+1} = Q_i^n - \lambda(F_{i+\frac{1}{2}}^n - F_{i-\frac{1}{2}}^n), \quad \forall i = 1,\dots,N, \quad \forall n = 0,\dots,N_T - 1, \tag{2.22}$$

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

$$F_{i\pm\frac{1}{2}}^n = \frac{1}{\Delta t} \int_{\tilde{x}_{i\pm\frac{1}{2}}^n}^{x_{i\pm\frac{1}{2}}} \tilde{q}(x; Q^n)\, dx, \tag{2.23}$$

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

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

$$M^n = \Delta x \sum_{i=1}^{N} Q_i^n. \tag{2.24}$$

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

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

where we are using that $F_{N+\frac{1}{2}}^n = F_{\frac{1}{2}}^n$, 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 $q \in C_P^2(\Omega)$, 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*

$$c_{i+\frac{1}{2}}^n = \frac{\Delta t}{\Delta x} u_{i+\frac{1}{2}}^n. \tag{2.25}$$

The CFL number is the maximum of the values $c_{i+\frac{1}{2}}^n$. 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 (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 (2.12) 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:

$$x_{i+\frac{1}{2}}^d(t^n, t^{n+1}) = x_{i+\frac{1}{2}} - u\Delta t. \tag{2.26}$$

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

$$\tilde{x}_{i+\frac{1}{2}}^n = x_{i+\frac{1}{2}} - \tilde{u}_{i+\frac{1}{2}}^n \Delta t, \tag{2.27}$$

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

$$\frac{1}{\Delta t} \int_{t^n}^{t^{n+1}} u(x_{i+\frac{1}{2}}^d(\theta, t^{n+1}), \theta) \, d\theta. \tag{2.28}$$

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

$$x_{i+\frac{1}{2}}^d(t^n, t^{n+1}) - \tilde{x}_{i+\frac{1}{2}}^n = \mathcal{O}(\Delta t^p). \tag{2.29}$$

2.3.1 DP1 scheme

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

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

as in FV3 papers (Lin & Rood, 1996; Putman & Lin, 2007). The idea behind DP1 is illustrated in Figure 2.4.

In this case, the time-averaged CFL is given by:

$$\tilde{c}_{i+\frac{1}{2}}^n = c_{i+\frac{1}{2}}^{n+\frac{1}{2}}. \tag{2.31}$$

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
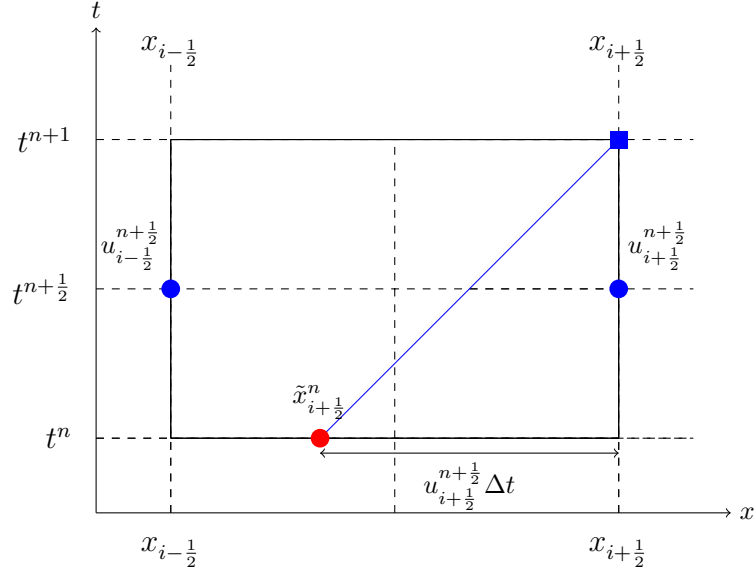
Figure 2.4: *Illustration of the scheme DP1 for computing the departure point at time level n emanating from $x_{i+\frac{1}{2}}$ (blue square) at time level $n+1$, assuming $u_{i+\frac{1}{2}}^{n+\frac{1}{2}} > 0$. The winds used are denoted by blue circles. The estimated departure point $\tilde{x}_{i+\frac{1}{2}}^n$ is illustrated by a red circle.*

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 (2.29) 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 (2.30), then the departure point from Equation (2.27) satisfies:*

$$x_{i+\frac{1}{2}}^d(t^n, t^{n+1}) - \tilde{x}_{i+\frac{1}{2}}^n = \mathcal{O}(\Delta t^2). \tag{2.32}$$

Proof. Using the midpoint rule (Theorem A.4) for the function $f(t) = u\big(x_{i+\frac{1}{2}}^d(t, t^{n+1}), t\big)$ in Equation (2.12), we obtain:

$$x_{i+\frac{1}{2}}^d(t^n, t^{n+1}) = x_{i+\frac{1}{2}} - u\big(x_{i+\frac{1}{2}}^d(t^{n+\frac{1}{2}}, t^{n+1}), t^{n+\frac{1}{2}}\big)\Delta t - \frac{1}{24}\frac{D^2u}{Dt^2}\big(x_{i+\frac{1}{2}}^d(\theta_1, t^{n+1}), \theta_1\big)\Delta t^2, \tag{2.33}$$

for $\theta_1 \in [t^n, t^{n+1}]$. Now observe that, from the intermediate value theorem for integrals and Equation (2.12), we have

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

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

$t = t^{n+\frac{1}{2}}$, we have:

$$u\left(x_{i+\frac{1}{2}}^{d}(t^{n+\frac{1}{2}}, t^{n+1}), t^{n+\frac{1}{2}}\right) = u_{i+\frac{1}{2}}^{n+\frac{1}{2}} - \left(u\frac{\partial u}{\partial x}\right)\left(x_{i+\frac{1}{2}}(\theta_3, t^{n+1}), t^{n+\frac{1}{2}}\right)u\left(x_{i+\frac{1}{2}}^{d}(\theta_2, t^{n+1}), \theta_2\right)\frac{\Delta t^2}{2},$$
(2.34)

for $\theta_3 \in [t^n, t^{n+1}]$. Substituting Equation (2.34) into Equation (2.33), 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 $\tilde{u}_{i+\frac{1}{2}}^{n}$ using the following equations (Durran, 2010):

$$\tilde{x}_{i+\frac{1}{2}}^{n+\frac{1}{2}} = x_{i+\frac{1}{2}} - u_{i+\frac{1}{2}}^{n}\frac{\Delta t}{2} = x_{i+\frac{1}{2}} - c_{i+\frac{1}{2}}^{n}\frac{\Delta x}{2},$$

$$\tilde{u}_{i+\frac{1}{2}}^{n} = u\left(\tilde{x}_{i+\frac{1}{2}}^{n+\frac{1}{2}}, t^{n} + \frac{\Delta t}{2}\right).$$
(2.35)

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

$$\tilde{u}_{i+\frac{1}{2}}^{n} = \begin{cases} \left(1 - \alpha_{i+\frac{1}{2}}^{n}\right)u_{i+\frac{1}{2}-k}^{n+\frac{1}{2}} + \alpha_{i+\frac{1}{2}}^{n}u_{i-\frac{1}{2}-k}^{n+\frac{1}{2}} & \text{if } u_{i+\frac{1}{2}}^{n} \geq 0, \\ -\alpha_{i+\frac{1}{2}}^{n}u_{i+\frac{3}{2}-k}^{n+\frac{1}{2}} + \left(1 + \alpha_{i+\frac{1}{2}}^{n}\right)u_{i+\frac{1}{2}-k}^{n+\frac{1}{2}} & \text{if } u_{i+\frac{1}{2}}^{n} < 0, \end{cases}$$
(2.36)

where $\frac{c_{i+\frac{1}{2}}^{n}}{2} = \alpha_{i+\frac{1}{2}}^{n} + k$, $k = \lfloor\frac{c_{i+\frac{1}{2}}^{n}}{2}\rfloor$, $\alpha_{i+\frac{1}{2}}^{n} \in [0, 1[$, and $\lfloor\cdot\rfloor$ is the floor function. This scheme leads to a third-order error in the departure point estimate (*cf. e.g.* Durran (2010, Section 7.1.2)). This scheme shall be referred to as **DP2** and the idea behind this scheme is depicted in Figure 2.5.

As we mention in scheme DP1, we use the exact values of the wind needed at the time step $n + \frac{1}{2}$ and cell edge indexes $i + \frac{1}{2}$. 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 $u_{-1+\frac{1}{2}}^{n}$ and $u_{N+\frac{3}{2}}^{n}$. If we consider large CFL numbers, more ghost cells are needed. In this case, it useful to work with the time-averaged CFL number:

$$\tilde{c}_{i+\frac{1}{2}}^{n} = \begin{cases} \left(1 - \frac{c_{i+\frac{1}{2}}^{n}}{2}\right)c_{i+\frac{1}{2}}^{n+\frac{1}{2}} + \frac{c_{i+\frac{1}{2}}^{n}}{2}c_{i-\frac{1}{2}}^{n+\frac{1}{2}} & \text{if } c_{i+\frac{1}{2}}^{n} \geq 0, \\ \frac{c_{i+\frac{1}{2}}^{n}}{2}c_{i+\frac{3}{2}}^{n+\frac{1}{2}} + \left(1 - \frac{c_{i+\frac{1}{2}}^{n}}{2}\right)c_{i+\frac{1}{2}}^{n+\frac{1}{2}} & \text{if } c_{i+\frac{1}{2}}^{n} < 0. \end{cases}$$
(2.37)
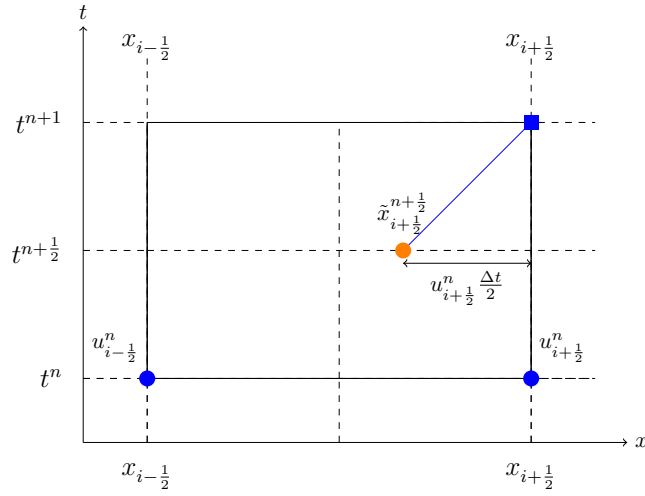
(a) *Intermediate departure point $\tilde{x}_{i+\frac{1}{2}}^{n+\frac{1}{2}}$ (orange circle). The winds used are depicted by blue circles.*
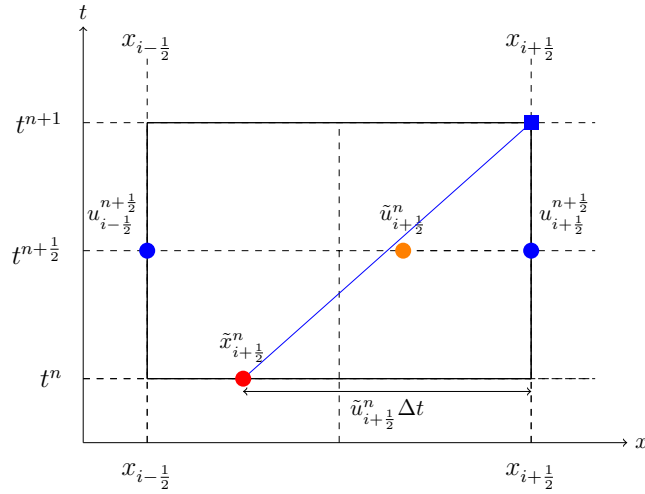
(b) *Final departure point $\tilde{x}_{i+\frac{1}{2}}^{n}$ (red circle) using the wind $\tilde{u}_{i+\frac{1}{2}}^{n}$ (orange circle) obtained using linear interpolation using the winds at time level $n + \frac{1}{2}$ (blue circles).*

Figure 2.5: *Illustration of the scheme DP2 for computing the departure point at time level n emanating from $x_{i+\frac{1}{2}}$ (blue square) at time level $n + 1$, assuming $u_{i+\frac{1}{2}}^{n} > 0$. The estimated departure point $\tilde{x}_{i+\frac{1}{2}}^{n}$ is illustrated by a red circle.*

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 dycore 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 $Q \in \mathbb{P}_\nu^N$ 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:

$$\tilde{q}(x; Q) = \sum_{i=1}^{N} \chi_i(x) q_i(x; Q), \tag{2.38}$$

where $q_i(x; Q) = \Phi\left(\frac{x - x_{i-\frac{1}{2}}}{\Delta x}; \alpha_i\right)$ 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 $q_i(x; Q) = \Phi\left(\frac{x - x_{i-\frac{1}{2}}}{\Delta x}; \alpha_i\right)$, 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:

$$Q(x) = \int_a^x q(\xi)\,d\xi, \tag{2.39}$$

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 $S_{i+\frac{1}{2}}^{(l)}$ and weights $\beta_{k,i}^{(l)}$, where $k \in S_{i+\frac{1}{2}}^{(l)}$. 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:

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

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

$$
\begin{cases}
\int_0^1 \Phi(\xi; \alpha_i) \, d\xi &= Q_i, \\
\Phi^{(l)}(0; \alpha_i) &= (\Delta x)^l \sum_{k \in S_{i-\frac{1}{2}}^{(l)}} \beta_{k,i}^{(l)} Q_k, \quad \text{for } l = 0, \dots, \frac{d}{2} - 1, \\
\Phi^{(l)}(1; \alpha_i) &= (\Delta x)^l \sum_{k \in S_{i+\frac{1}{2}}^{(l)}} \beta_{k,i}^{(l)} Q_k, \quad \text{for } l = 0, \dots, \frac{d}{2} - 1.
\end{cases}
\tag{2.41}
$$

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 (2.40) and Equation (2.41) 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:

$$
q_i(x; Q) = q_{L,i} + \Delta q_i z_i(x) + q_{6,i} z_i(x)(1 - z_i(x)), \quad \text{where } z_i(x) = \frac{x - x_{i-\frac{1}{2}}}{\Delta x}, \quad x \in X_i, \tag{2.42}
$$

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:

$$
\Delta q_i = q_{R,i} - q_{L,i}. \tag{2.43}
$$

The average value of q_i is given by:

$$
\frac{1}{\Delta x} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} q_i(x; Q) \, dx = \frac{(q_{L,i} + q_{R,i})}{2} + \frac{q_{6,i}}{6}. \tag{2.44}
$$

Under the hypothesis of mass conservation, we have:

$$q_{6,i} = 6\left(Q_i - \frac{(q_{L,i} + q_{R,i})}{2} \right). \tag{2.45}$$

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 (2.38) 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 (2.39), this is achieved using finite-differences.

2.4.1 Unlimited PPM reconstruction

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):

$$q_{i+\frac{1}{2}} = \frac{1}{2}\left(Q_{i+1} + Q_i \right) - \frac{1}{6}\left(\delta Q_{i+1} - \delta Q_i \right), \tag{2.46}$$

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

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

We notice that Formula (2.47) may be rewritten more explicitly as:

$$q_{i+\frac{1}{2}} = \frac{7}{12}\left(Q_{i+1} + Q_i \right) - \frac{1}{12}\left(Q_{i+2} + Q_{i-1} \right). \tag{2.48}$$

The Formula (2.48) 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:

$$q_{R,i} = q_{i+\frac{1}{2}} \tag{2.49}$$

$$q_{L,i} = q_{i-\frac{1}{2}}. \tag{2.50}$$

During this work, we refer to this PPM scheme as **UNLIM**. In FV3, the 1D advection solver input is named "hord" (horizontal advection method). We note that UNLIM is not implemented in FV3, but we implemented it because, since it has no limiter, it is suitable for checking order of accuracy for the schemes investigated in this work. For a list of reconstruction schemes available in FV3, refer to Harris et al. (2021). One may see that the scheme hord=5 in FV3 is the closest to UNLIM, as hord=5 only applies a filter on the UNLIM flux to remove $2\Delta x$ numerical noise.

2.4.2 Monotonic PPM reconstruction

This Subsection is dedicated to presenting a possible way of avoiding 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 (Harris et al., 2021).

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

$$\delta_m Q_i = \max(|\delta Q_i|, 2\delta Q_{\min,i}, 2\delta Q_{\max,i}) \cdot \mathrm{sgn}(\delta Q_i) \tag{2.51}$$

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 (2.46) as:

$$q_{i+\frac{1}{2}} = \frac{1}{2}\left(Q_{i+1} + Q_i\right) - \frac{1}{6}\left(\delta_m Q_{i+1} - \delta_m Q_i\right). \tag{2.52}$$

The values $q_{R,i}$ and $q_{L,i}$ are then computed using Equations (2.49) and (2.50), respectively. The monotonicity is achieved by the following scheme:

$$q_{L,i} \leftarrow Q_i - \max(|\delta_m Q_i|, |q_{L,i} - Q_i|) \cdot \mathrm{sgn}(\delta_m Q_i), \tag{2.53}$$
$$q_{R,i} \leftarrow Q_i - \max(|\delta_m Q_i|, |q_{R,i} - Q_i|) \cdot \mathrm{sgn}(\delta_m Q_i). \tag{2.54}$$

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 **MONO**. In FV3, the paramenter "hord" is set equal to 8 to use this scheme. At last, we point out again that many other PPM reconstruction schemes are available in the literature and in FV3 (Harris et al., 2021; Lin et al., 2017), but for simplicity, we are just going to consider the schemes UNLIM and MONO.

To conclude this section, we illustrate the difference between the UNLIM and MONO reconstructions using a numerical experiment. Following the approach in Trefethen (2000), we consider the periodic Gaussian profile defined as:

$$q(x) = 0.1 + 0.9 \exp\left(-10 \sin^2\left(\frac{\pi x}{L}\right)\right), \quad x \in \left[-\frac{L}{2}, \frac{L}{2}\right], \tag{2.55}$$

where $L = \frac{\pi}{2}R$, and $R = 6.371 \times 10^6$ meters is the Earth's radius. Figure 2.6 depicts the periodic Gaussian hill along with its average values (Figure 2.6a) for $N = 8$ control volumes, with unlimited (Figure 2.6b) and monotonic (Figure 2.6c) reconstructions. It is clear that the unlimited reconstruction (Figure 2.6b) creates new extrema, which are not observed in the monotonic reconstruction (Figure 2.6c).
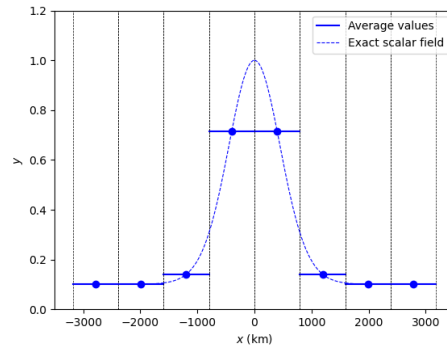
(a) *Periodic Gaussian hill (blue dashed line) and its average values (blue horizontal lines and blue circles).*
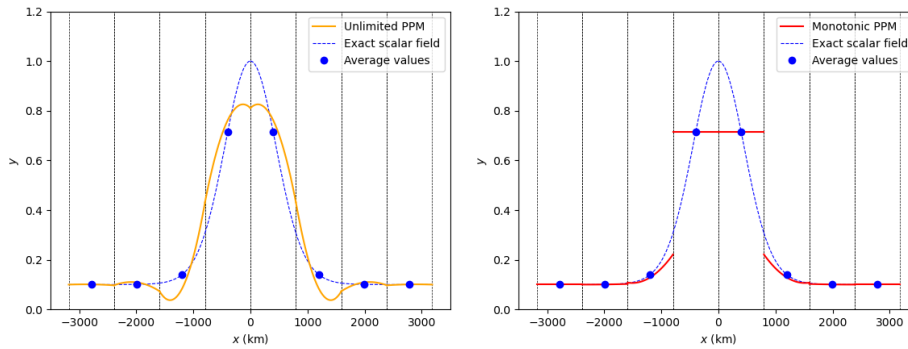
(b) *Unlimited PPM reconstruction of the periodic Gaussian hill (yellow curve).*

(c) *Monotonic PPM reconstruction of the periodic Gaussian hill (red curve).*

Figure 2.6: *Illustration of the periodic Gaussian hill from Equation* (2.55) *and its average values (a). The unlimited and monotonic PPM reconstructions are depicted in (b) and (c). We are considering N = 8 control volumes, whose boundaries are illustrated using vertical dashed lines.*

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 $\tilde{x}_{i+\frac{1}{2}}^n = x_{i+\frac{1}{2}} - \tilde{u}_{i+\frac{1}{2}}^n \Delta t$ for a time-averaged wind $\tilde{u}_{i+\frac{1}{2}}^n$ as explained in Section 2.3. The numerical flux function $F_{i+\frac{1}{2}}^n$ is then suggested in Problem 2.4:

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

Notice that if we define the averaged CFL number,

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

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):

$$F^n_{i+\frac{1}{2}}[Q^n, \tilde{u}^n] = \frac{1}{\Delta t} \begin{cases} \Delta x \sum_{l=0}^{k-1} Q_{i-l} + \int_{x_{i-k+\frac{1}{2}}-\alpha^n_{i+\frac{1}{2}}\Delta x}^{x_{i-k+\frac{1}{2}}} \tilde{q}(x; Q^n) \, dx, & \text{if } \tilde{u}^n_{i+\frac{1}{2}} \geq 0, \\ \Delta x \sum_{l=0}^{k-1} Q_{i-l} - \int_{x_{i-k+\frac{1}{2}}}^{x_{i-k+\frac{1}{2}}-\alpha^n_{i+\frac{1}{2}}\Delta x} \tilde{q}(x; Q^n) \, dx, & \text{if } \tilde{u}^n_{i+\frac{1}{2}} < 0, \end{cases} \quad (2.57)$$

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

We will provide explicit expressions for the integrals in Equation (2.57) when using the PPM method. For each control volume edge, denoted by $i = 0, \dots, N$, and $y > 0$, we define the following averages of the Piecewise-Parabolic approximation, as defined in Equation (2.38) for Q^n (Colella & Woodward, 1984):

$$F_{L,i+\frac{1}{2}}[Q^n, y] = \frac{1}{y} \int_{x_{i+\frac{1}{2}}-y}^{x_{i+\frac{1}{2}}} \tilde{q}(x; Q^n) \, dx, \quad (2.58)$$

and

$$F_{R,i+\frac{1}{2}}[Q^n, y] = \frac{1}{y} \int_{x_{i+\frac{1}{2}}}^{x_{i+\frac{1}{2}}+y} \tilde{q}(x; Q^n) \, dx. \quad (2.59)$$

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 (2.42) that:

$$F_{L,i+\frac{1}{2}}[Q^n, y] = \frac{1}{y} \int_{x_{i+\frac{1}{2}}-y}^{x_{i+\frac{1}{2}}} q_i(x; Q^n) \, dx = q_{R,i} + \frac{(q_{6,i} - \Delta q_i)}{2\Delta x} y - \frac{q_{6,i}}{3\Delta x^2} y^2, \quad (2.60)$$

and

$$F_{R,i+\frac{1}{2}}[Q^n, y] = \frac{1}{y} \int_{x_{i+\frac{1}{2}}}^{x_{i+\frac{1}{2}}+y} q_{i+1}(x; Q^n) \, dx = q_{L,i+1} + \frac{(q_{6,i+1} + \Delta q_{i+1})}{2\Delta x} y - \frac{q_{6,i+1}}{3\Delta x^2} y^2. \quad (2.61)$$

The numerical flux function for PPM is then defined by:

$$\mathfrak{F}^{PPM}_{i+\frac{1}{2}}[Q^n, \tilde{u}^n] = \begin{cases} F_{L,i+\frac{1}{2}}[Q^n, \alpha^n_{i+\frac{1}{2}}\Delta x] & \text{if } \tilde{u}^n_{i+\frac{1}{2}} \geq 0, \\ F_{R,i+\frac{1}{2}}[Q^n, -\alpha^n_{i+\frac{1}{2}}\Delta x] & \text{if } \tilde{u}^n_{i+\frac{1}{2}} < 0, \end{cases} \quad (2.62)$$

and

$$F^n_{i+\frac{1}{2}}[Q^n, \tilde{u}^n] = \frac{1}{\Delta t} \left(\Delta x \sum_{l=0}^{k-1} Q_{i-l} + \Delta x \alpha^n_{i+\frac{1}{2}} \mathfrak{F}^{PPM}_{i+\frac{1}{2}}[Q^n, \tilde{u}^n] \right). \quad (2.63)$$

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

$$\mathfrak{F}^{PPM}_{i+\frac{1}{2}}[Q^n, \tilde{c}^n] = \begin{cases} q_{R,i} + \left(\frac{q_{6,i} - \Delta q_i}{2} \right) \tilde{c}^n_{i+\frac{1}{2}} - \frac{q_{6,i}}{3} (\tilde{c}^n_{i+\frac{1}{2}})^2, & \text{if } \tilde{c}^n_{i+\frac{1}{2}} \geq 0, \\ q_{L,i+1} + \left(\frac{q_{6,i+1} + \Delta q_{i+1}}{2} \right) \tilde{c}^n_{i+\frac{1}{2}} - \frac{q_{6,i+1}}{3} (\tilde{c}^n_{i+\frac{1}{2}})^2, & \text{if } \tilde{c}^n_{i+\frac{1}{2}} < 0, \end{cases} \quad (2.64)$$

and

$$F^n_{i+\frac{1}{2}}[Q^n, \tilde{c}^n] = \tilde{u}^n_{i+\frac{1}{2}} \mathfrak{F}^{PPM}_{i+\frac{1}{2}}[Q^n, \tilde{c}^n], \quad (2.65)$$

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 unlimited PPM and monotonic PPM schemes, $F^n_{i+\frac{1}{2}}$ uses the stencil $S_{i+\frac{1}{2}} = \{i-3, i-2, i-1, i, i+1, i+2, i+3\}$, and therefore we need $\nu = 3$ layers of ghost cells.

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

$$b_{L,i} = q_{L,i} - Q^n_i, \tag{2.66}$$

$$b_{R,i} = q_{R,i} - Q^n_i. \tag{2.67}$$

Then, Equation (2.64) becomes:

$$\mathfrak{F}^{PPM}_{i+\frac{1}{2}}[Q^n, \tilde{c}^n] = \begin{cases} Q^n_i + (1 - \tilde{c}^n_{i+\frac{1}{2}})\big(b_{R,i} - \tilde{c}^n_{i+\frac{1}{2}}(b_{L,i} + b_{R,i})\big), & \text{if } \tilde{c}^n_{i+\frac{1}{2}} \geq 0, \\ Q^n_{i+1} + (1 + \tilde{c}^n_{i+\frac{1}{2}})\big(b_{L,i+1} + \tilde{c}^n_{i+\frac{1}{2}}(b_{L,i+1} + b_{R,i+1})\big), & \text{if } \tilde{c}^n_{i+\frac{1}{2}} < 0, \end{cases} \tag{2.68}$$

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

$$Q^{n+1}_i = Q^n_i - \big(\tilde{c}^n_{i+\frac{1}{2}} \mathfrak{F}^{PPM}_{i+\frac{1}{2}}[Q^n, \tilde{c}^n] - \tilde{c}^n_{i-\frac{1}{2}} \mathfrak{F}^{PPM}_{i-\frac{1}{2}}[Q^n, \tilde{c}^n]\big), \tag{2.69}$$

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 UNLIM or MONO;

3. Evalute the pertubation values (for $i = 1, \cdots, N$) using Equations (2.66) and (2.67);

4. Evaluate the fluxes $\mathfrak{F}^{PPM}_{i+\frac{1}{2}}$ (for $i = 0, \cdots, N$) using Equation (2.68);

5. Update the average values Q^{n+1} using Equation (2.69).

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 unlimited PPM (Subsection 2.4.1) and monotonic PPM (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 (2.24). 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 (2.55), 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:

$$q_0(x) = \begin{cases} 1 & \text{if } x \in [-0.1L, 0.1L], \\ 0.1 & \text{otherwise.} \end{cases} \tag{2.70}$$

for the linear advection equation with constant velocity, which we adopt as $u = \frac{L}{T}$.
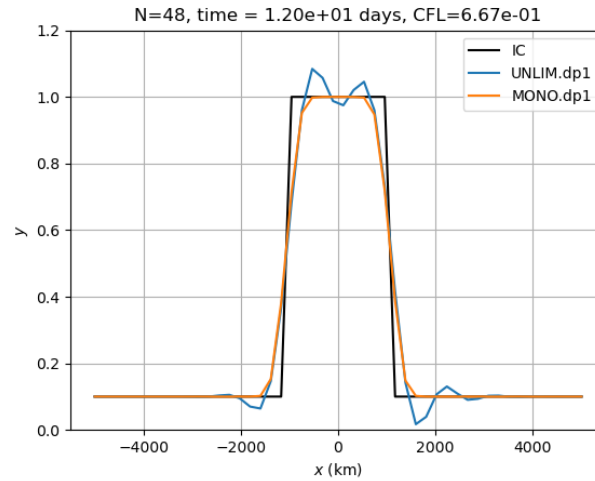
Figure 2.7: *Linear advection experiment using the IC given by Equation (2.70) (black curve) with constant velocity. These figures show the advected profile after 12 days (one time period). Reconstruction schemes employed: unlimited PPM (UNLIM - blue curve) and monotonic PPM (MONO - orange curve).*

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 $N = 48$ (therefore $\Delta x \approx 208$ km) 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 2.7, we present the obtained results. It is evident that the monotonic PPM exhibit a significant advantage. This scheme effectively prevent the strong oscillations observed in the unlimited PPM 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 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

$$u(x, t) = u_0 \cos\left(\frac{\pi t}{T}\right) \cos^2\left(\pi\left(\frac{x}{L} - \frac{t}{T}\right)\right) + u_1. \tag{2.71}$$

We adopt the parameters $T = 12$ days and $u_0 = u_1 = \frac{L}{T}$. We set the periodic Gaussian profile defined in Equation (2.55) as the initial condition. The velocity function given by Equation (2.71) 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 2.8, we show how the solution behaves using a high-resolution ($N = 768$), the MONO 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 p-norm as follows:

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

We are going to consider $p = 1$ and $p = \infty$. 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 2.9. In the L_∞ norm (Figure 2.9a), for the unlimited PPM, the DP1 scheme results in a first-order error in the departure point, which dominates the total error. This 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 the unlimited PPM. For the monotonic PPM, the DP2 slightly reduces the L_∞ error.

However, in the L_1 norm, as shown in Figure 2.9b, for both unlimited PPM and monotonic PPM, 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

(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.*
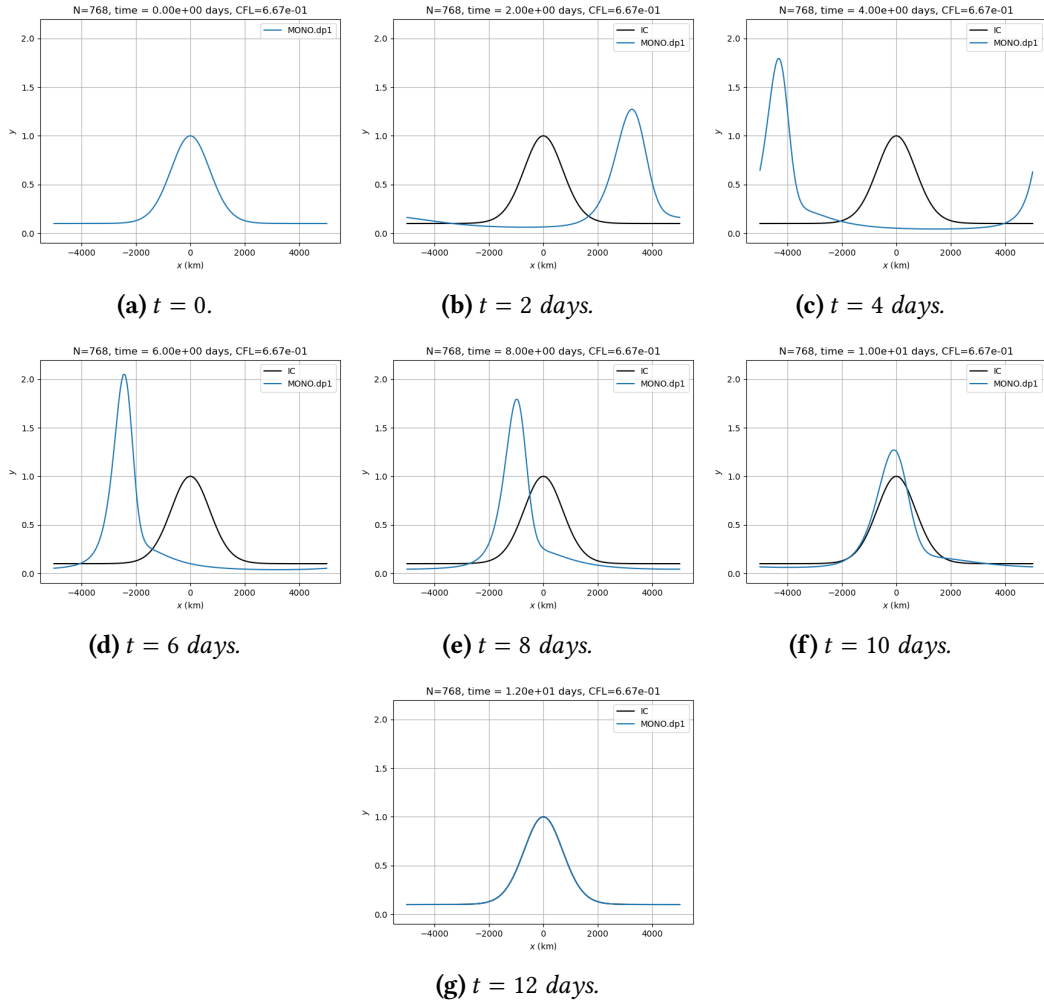
Figure 2.8: *Linear advection experiment using the velocity from Equation* (2.55), *a CFL number equal to* 0.67, $N = 768$ *cells, and the IC is given by Equation* (2.55). *These figures show the advected profile at day 0* (2.8a) *and after 2* (2.8b), *4* (2.8c), *6* (2.8d), *8* (2.8e), *10* (2.8f), *and 12* (2.8g) *days. We are using the monotonic PPM scheme with the DP1 departure point scheme.*

is the reconstruction of a function from its average values. We employed the PPM method introduced by Colella and Woodward (1984) without limiter and its monotonic variant such as the one from Lin (2004). 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 the unlimited PPM resulted in a final first-order error, despite the scheme having third-order accuracy in space. However, the DP2 scheme with the unlimited PPM preserved third-order

(a) L_∞ *error.* **(b)** L_1 *error.*
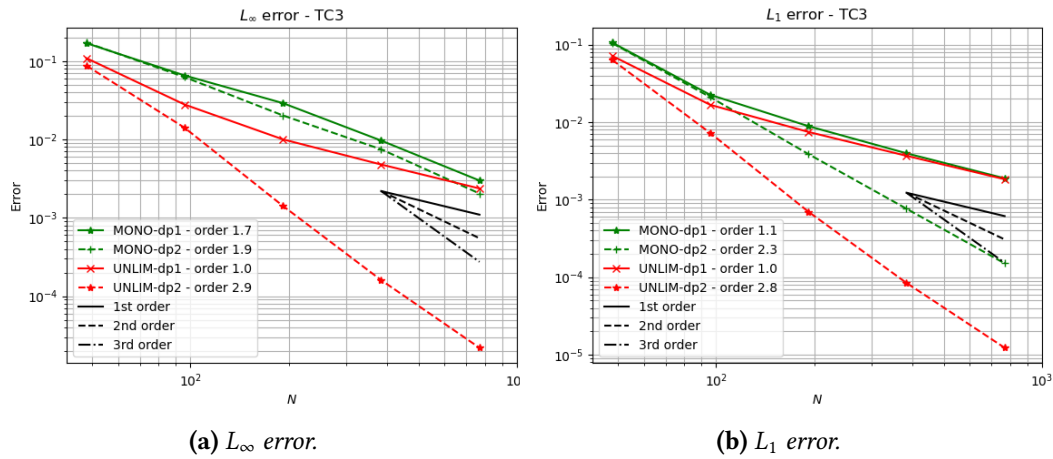
Figure 2.9: *Relative error for the unlimited PPM (UNLIM -red lines) and monotonic PPM (MONO - green lines) schemes in L_∞ (Figure 2.9a) and L_1 norms (Figure 2.9b). Results using DP1 scheme uses solid lines and DP2 results uses dashed lines. The IC given by Equation (2.55) and the variable velocity given by Equation (2.71).*

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 monotonic PPM, 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. Final thoughts are presented in Section 3.5.

3.1 Two-dimensional advection equation in the 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* $u = (u_{i+\frac{1}{2},j})_{i=-v,\dots,N+v}^{j=-v+1,\dots,M+v} \in \mathbb{R}_v^{(N+1)\times M}$, $v = (v_{i,j+\frac{1}{2}})_{i=-v+1,\dots,N+v}^{j=-v,\dots,M+v} \in \mathbb{R}_v^{N\times(M+1)}$.

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, $q_{ij}^n = q(x_i, y_j, t^n)$, $u_{i+\frac{1}{2},j}^n = u(x_{i+\frac{1}{2}}, y_j, t^n)$, $v_{i,j+\frac{1}{2}}^n = u(x_i, y_{j+\frac{1}{2}}, t^n)$. 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 $q_{i+\frac{1}{2},j}^n = q(x_{i+\frac{1}{2}}, y_j, t^n)$ and $q_{i,j+\frac{1}{2}}^n = q(x_i, y_{j+\frac{1}{2}}, t^n)$.
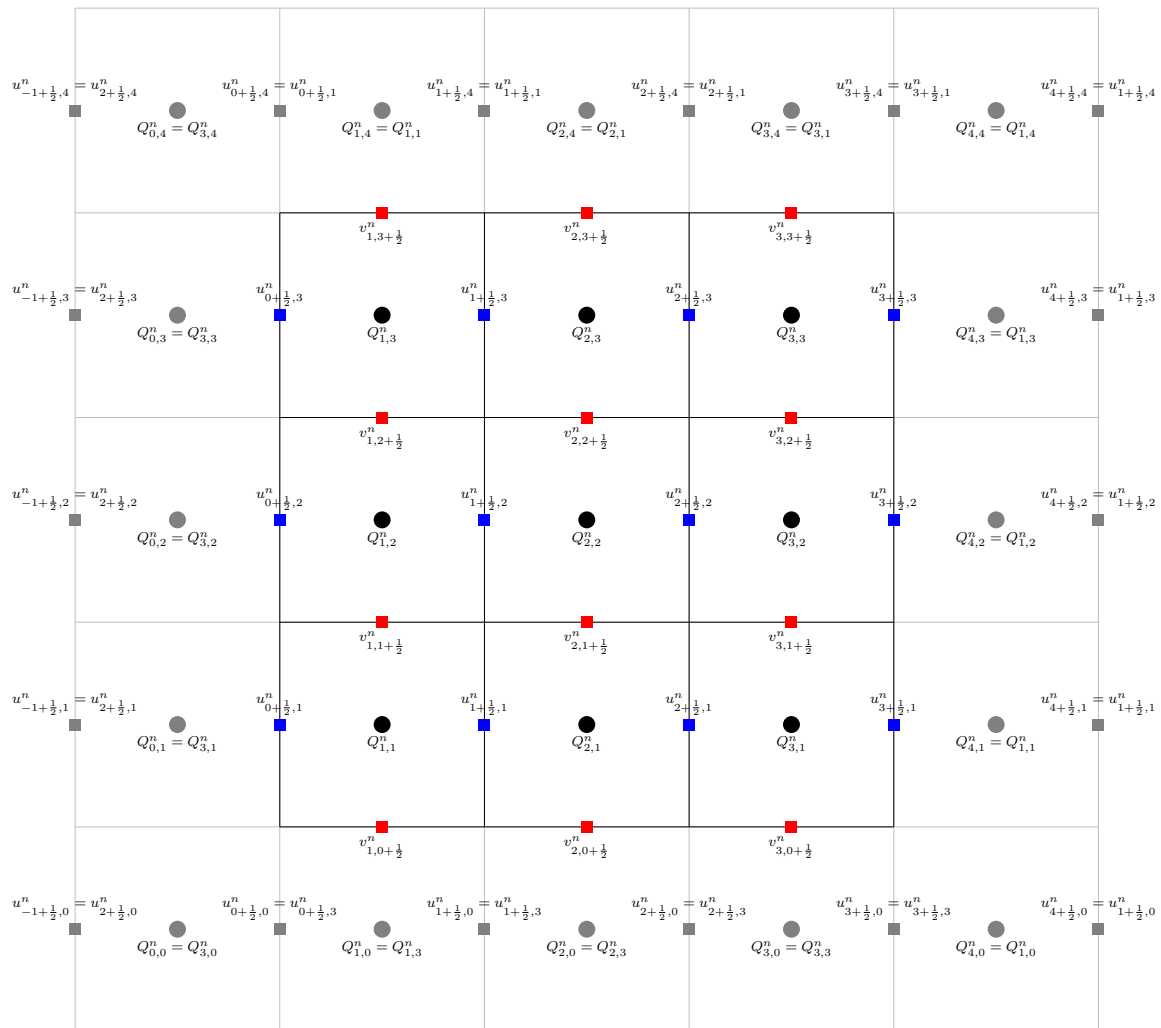
Figure 3.1: *Illustration of* ($\Delta x, \Delta y$)*-grid function Q (black circles) and a* ($\Delta x \Delta y$)*-C grid wind u (blue squares) and v (red squares) and its ghost cell values (in gray) assuming biperiodicity.*

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** or **divergent free** if $\nabla \cdot \boldsymbol{u} = 0$. We define the

$(\Delta x, \Delta y)$-grid function $\mathfrak{D}^n$ as the exact divergence of $q\boldsymbol{u}$ at the cell centers, namely

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

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:

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

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:

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

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

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

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 $Q(t) = (Q_{ij}(t))_{i=-\nu+1,\dots,N+\nu}^{j=-\nu+1,\dots,M+\nu}$.

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

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

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

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

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, $S_P(\Omega)$ 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:

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

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:

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

$$-\int_{x_1}^{x_2} \left((vq)(x, y_2, t) - (vq)(x, y_1, t) \right) dx.$$

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

$$\int_{x_1}^{x_2} \int_{y_1}^{y_2} q(x, y, t_{n+1}) \, dx \, dy = \int_{x_1}^{x_2} \int_{y_1}^{y_2} q(x, y, t_n) \, dx \, dy \tag{3.9}$$

$$- \int_{t_1}^{t_2} \int_{y_1}^{y_2} \left((uq)(x_2, y, t) - (uq)(x_1, y, t) \right) dy \, dt$$

$$- \int_{t_1}^{t_2} \int_{x_1}^{x_2} \left((vq)(x, y_2, t) - (vq)(x, y_1, t) \right) dx \, dt.$$

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:*

$$\int_{x_1}^{x_2} \int_{y_1}^{y_2} q(x, y, t) \, dx \, dy = \int_{x_1}^{x_2} \int_{y_1}^{y_2} q(x, y, t) \, dx \, dy$$

$$- \int_{t_1}^{t_2} \int_{y_1}^{y_2} \left((uq)(x_2, y, t) - (uq)(x_1, y, t) \right) dy \, dt$$

$$- \int_{t_1}^{t_2} \int_{x_1}^{x_2} \left((vq)(x, y_2, t) - (vq)(x, y_1, t) \right) dx \, dt.$$

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

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:

$$Q_{ij}(t_{n+1}) = Q_{ij}(t_n) - \frac{\Delta t}{\Delta x \Delta y} \delta_x \left(\frac{1}{\Delta t} \int_{t_1}^{t_2} \int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}} (uq)(x_i, y, t) \, dy \, dt \right) \tag{3.12}$$

$$- \frac{\Delta t}{\Delta x \Delta y} \delta_y \left(\frac{1}{\Delta t} \int_{t_1}^{t_2} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} (vq)(x, y_j, t) \, dx \, dt \right),$$

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 $Q_{ij}(t_n)$, $\forall i = 1, \ldots, N$, $\forall j = 1, \ldots, M$, $\forall n = 0, \ldots, N_T - 1$, given the initial values $Q_{ij}(0)$, $\forall i = 1, \ldots N$, $\forall j = 1, \ldots, M$. 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 3.2 consists of a finding a scheme of the form:*

$$Q_{ij}^{n+1} = Q_{ij}^n - \lambda \delta_i F_{ij}^n - \lambda \delta_j G_{ij}^n, \tag{3.13}$$

$$\forall i = 1, \ldots, N, \quad \forall j = 1, \ldots, M, \quad \forall n = 0, \ldots, N_T - 1,$$

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 $\frac{1}{\Delta t \Delta y} \int_{t_n}^{t_{n+1}} \int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}} (uq)(x_{i+\frac{1}{2}}, y, t) \, dy \, dt$, $\forall i = 0, 1, \ldots, N$, and $G_{i, j+\frac{1}{2}}^n$ is known as numerical flux in the y direction and it approximates $\frac{1}{\Delta t \Delta x} \int_{t_n}^{t_{n+1}} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} (vq)(x, y_{j+\frac{1}{2}}, t) \, dx \, dt$, $\forall j = 0, 1, \ldots, M$, 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 $\max_{i,j}\{|u_{i+\frac{1}{2}, j}^n|\} \frac{\Delta t}{\Delta x}$ and $\max_{i,j}\{|v_{i, j+\frac{1}{2}}^n|\} \frac{\Delta t}{\Delta y}$, 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:

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

$$= M^n - \Delta t \sum_{j=1}^{M} (F_{N+\frac{1}{2},j}^n - F_{\frac{1}{2},j}^n) - \Delta t \sum_{i=1}^{N} (G_{i,M+\frac{1}{2}}^n - G_{i,\frac{1}{2}}^n) = M^n,$$

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

As we mentioned in Problem 3.3, the initial condition may be assumed as q_{ij}^0 or $Q_{ij}(0)$. For two-dimensional simulations, we are going to assume q_{ij}^0 as initial data to avoid the computation of integrals. Furthermore, the errors will be calculated using the values q_{ij}^n 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 $|Q_{ij}(t^n) - q_{ij}^n| = C_1 \Delta x^2 + C_2 \Delta x \Delta y + C_3 \Delta y^2$, 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 $\mathbb{D}^n(Q^n, u^n, v^n) \in \mathbb{P}_v^{N \times M}$ given by:*

$$\mathbb{D}_{ij}^n(Q^n, u^n, v^n) = \frac{1}{\Delta t}\left(\frac{\delta_i F_{ij}^n}{\Delta x} + \frac{\delta_j G_{ij}^n}{\Delta y}\right), \quad i = 1, \dots, N, \quad j = 1, \dots, M. \quad (3.14)$$

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

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

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

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

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

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

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:

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

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

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

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:

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

where $\mathfrak{D}^n \in \mathbb{P}_v^{N\times M}$ 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 $n = 0, \ldots, N_T - 1$, 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:

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

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

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

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

We shall assume that these problems are solved using a 1D-FV scheme as in Problem 2.4 with the PPM numerical flux functions $\mathfrak{F}_{i+\frac{1}{2},j}^{PPM,x}[Q_{\times,j}^n, \tilde{c}^{x,n}]$ and $\mathfrak{F}_{i,j+\frac{1}{2}}^{PPM,y}[Q_{i,\times}^n, \tilde{c}^{y,n}]$, respectively, where $\tilde{c}_{i+\frac{1}{2},j}^{x,n}$ is the time-averaged CFL used in the departure point estimation in the x direction and $\tilde{c}_{i,j+\frac{1}{2}}^{y,n}$ 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 (2.68)). 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 $q_{L,ij}^x$, $q_{R,ij}^x$, $q_{L,ij}^y$, and $q_{R,ij}^y$, 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 unlimited PPM (UNLIM) and monotonic PPM (MONO) 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 (2.66) and (2.66), we define the perturbation values in the x direction as:

$$b_{L,i,j}^x = q_{L,i,j}^x - Q_{ij}^n, \tag{3.19}$$

$$b_{R,i,j}^x = q_{R,i,j}^x - Q_{ij}^n, \tag{3.20}$$

and the perturbation values in the y direction as:

$$b_{L,i,j}^y = q_{L,i,j}^y - Q_{ij}^n, \tag{3.21}$$

$$b_{R,i,j}^y = q_{R,i,j}^y - Q_{ij}^n. \tag{3.22}$$

Then, we may express the 1D fluxes in x direction as in Equation (2.68), namely:

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

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

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

for $i = -\nu + 1, \ldots, N + \nu$, $j = 0, \ldots, M$. For both the unlimited and monotonic PPM schemes, we set $\nu = 3$.

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

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

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.
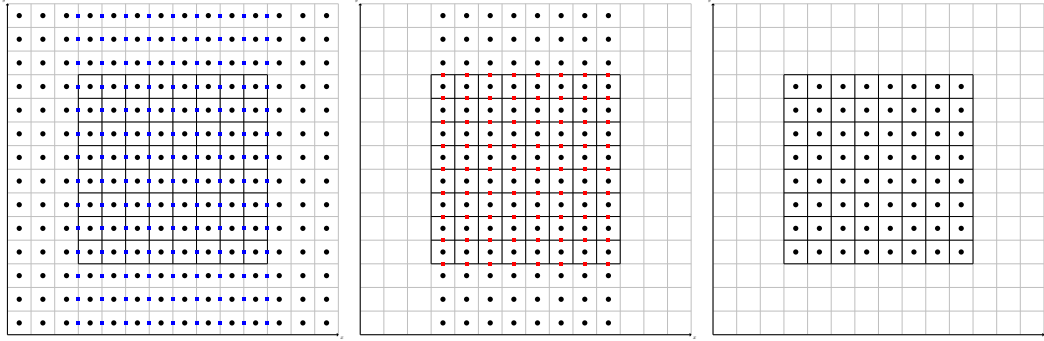
(a) Q^n (black circles) and u at edges (blue squares).

(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).*
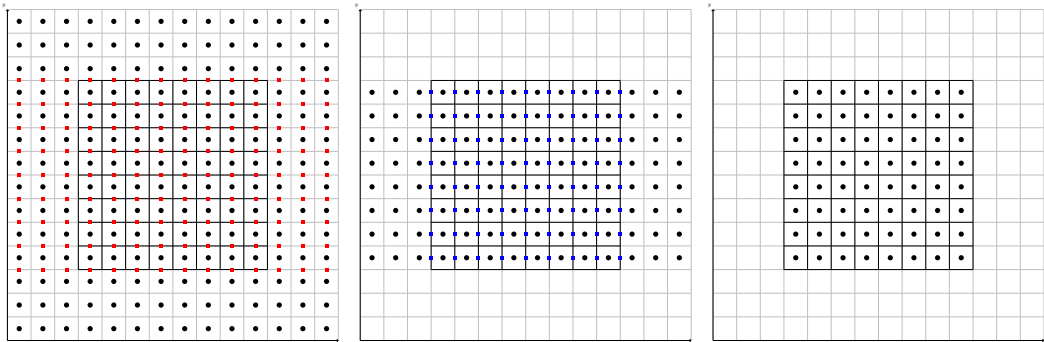
(a) Q^n (black circles) and v at edges (red squares).

(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

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

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

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

for $j = 1, \ldots, M$, $i = 1, \ldots, N$ (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

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

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

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

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

$$\begin{aligned}
Q^{n+1} = \frac{(Q^{xy,n} + Q^{yx,n})}{2} = Q^n &+ \frac{1}{2}\mathbf{F}[Q^n, \tilde{c}^{x,n}] + \frac{1}{2}\mathbf{G}[Q^n, \tilde{c}^{y,n}] \\
&+ \frac{1}{2}\mathbf{F}\left[Q^n + \mathbf{G}[Q^n, \tilde{c}^{y,n}], \tilde{c}^{x,n}\right] + \frac{1}{2}\mathbf{G}\left[Q^n + \mathbf{F}[Q^n, \tilde{c}^{x,n}], \tilde{c}^{y,n}\right].
\end{aligned}$$
(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 $\mathbf{F}$ and $\mathbf{G}$ are linear in their first input. This implies that Equation (3.25) may be rewritten as:

$$\begin{aligned}
Q^{n+1} = Q^n &+ \mathbf{F}[Q^n, \tilde{c}^{x,n}] + \mathbf{G}[Q^n, \tilde{c}^{y,n}] \\
&+ \frac{1}{2}\mathbf{F}\left[\mathbf{G}[Q^n, \tilde{c}^{y,n}], \tilde{c}^{x,n}\right] + \frac{1}{2}\mathbf{G}\left[\mathbf{F}[Q^n, \tilde{c}^{x,n}], \tilde{c}^{y,n}\right].
\end{aligned}$$
(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 the unlimited PPM, implying that $\mathbf{F}$ and $\mathbf{G}$ 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

$$u(x, y, t) = -\partial_y \psi(x, y, t),$$
$$v(x, y, t) = \partial_x \psi(x, y, t).$$

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

$$u^n_{i+\frac{1}{2}, j} = -\left(\frac{\psi^n_{i+\frac{1}{2}, j+\frac{1}{2}} - \psi^n_{i+\frac{1}{2}, j-\frac{1}{2}}}{\Delta y} \right),$$

$$v^n_{i, j+\frac{1}{2}} = \frac{\psi^n_{i+\frac{1}{2}, j+\frac{1}{2}} - \psi^n_{i-\frac{1}{2}, j+\frac{1}{2}}}{\Delta x},$$

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:

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

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

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

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

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

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

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

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

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

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.

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	$-Q_{ij}^n + \dfrac{Q_{ij}^n + \mathbf{F}_{ij}(Q^n, \tilde{c}_j^{x,n})}{1 - \frac{1}{\|\Omega_{ij}\|}\left(\mathcal{A}_{i+\frac{1}{2},j}^x - \mathcal{A}_{i-\frac{1}{2},j}^x\right)}$	$-Q_{ij}^n + \dfrac{Q_{ij}^n + \mathbf{G}_{ij}(Q^n, \tilde{c}^{y,n})}{1 - \frac{1}{\|\Omega_{ij}\|}\left(\mathcal{A}_{i,j+\frac{1}{2}}^y - \mathcal{A}_{i,j-\frac{1}{2}}^y\right)}$

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).*

It is easy to see that the PL operator 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 unlimited PPM (Subsection 2.4.1) and the monotonic PPM (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 $\Delta x^{(k)} = \Delta y^{(k)} = \frac{L}{N^{(k)}}$, $N^{(k)} = 48 \times 2^k$, $k = 0, \dots, 4$. We introduce the relative error in the p-norm:

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

We are going to consider $p = 1$ and $p = \infty$. 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:

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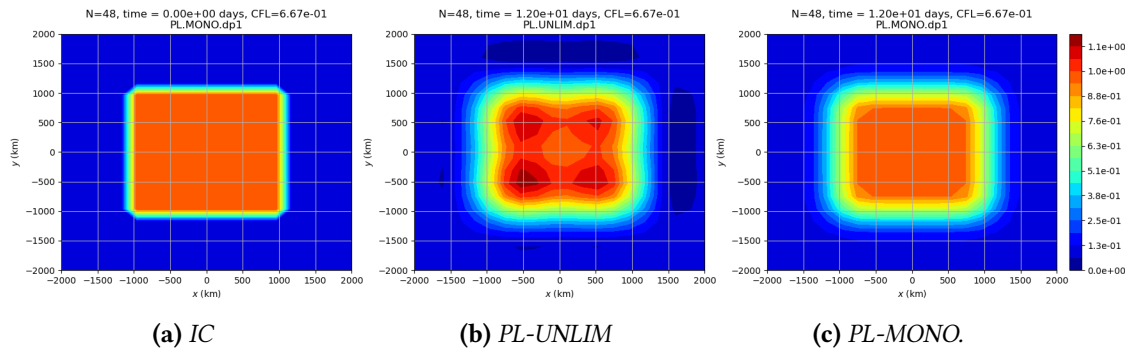

(a) *IC* (b) *PL-UNLIM* (c) *PL-MONO.*

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 unlimited PPM (UNLIM) (b) and the monotonic PPM (MONO) (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 $N = M = 48$ (therefore $\Delta x = \Delta y \approx 208$ km), 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.

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.
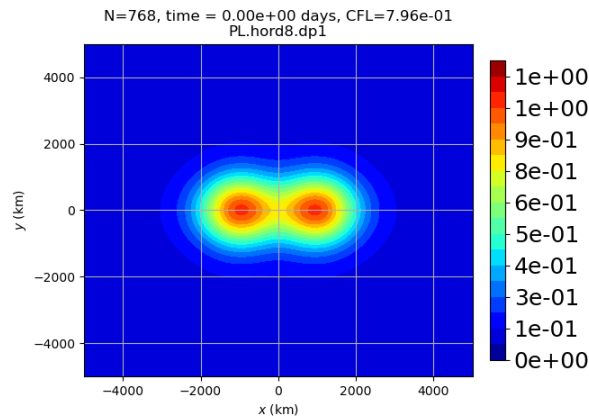

Figure 3.5: *Two Gaussian hills IC (Equation* (3.31)*).*

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-MONO 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.
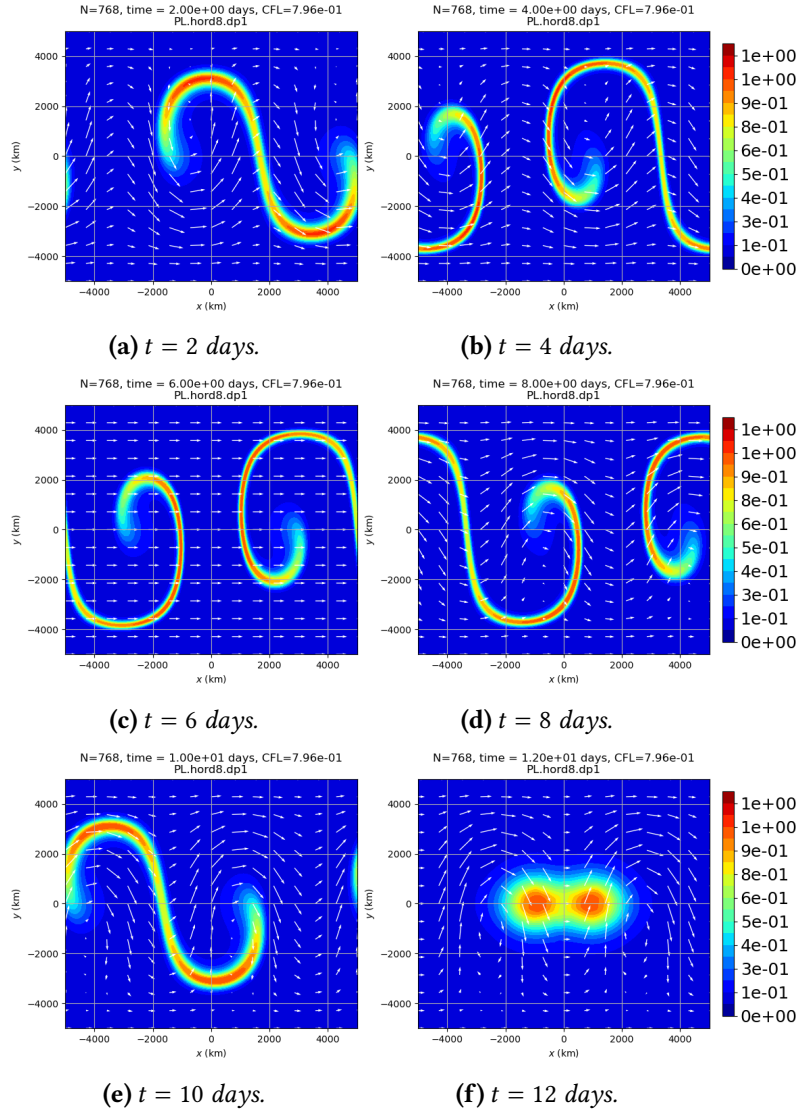

(a) $t = 2$ *days.*

(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. The white arrows illustrate the velocity field. We are using the PL-DP1-MONO scheme, where MONO is the monotonic PPM.*

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

We can observe from Figure 3.7 that for the unlimited PPM, PL-DP1 and LT-DP2 have smaller error and higher convergence order than PL-DP2 and LT-DP1. However, when considering the monotonic PPM, all the schemes have the same error in L_∞ norm. The results of the monotonic scheme have an order of approximately 1.5, which is expected, as the monotonic limiter reduces the order of accuracy. 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. The reason that justifies why improving the departure point calculation of the PL scheme, moving from PL-DP1 to PL-DP2, does not improve the overall error, is justified by the analysis provided in Section 3.3.2, where it is pointed out that PL-DP2 will not satisfy the discrete divergence-free condition (Equation (3.27)), introducing a first-order error in this case.

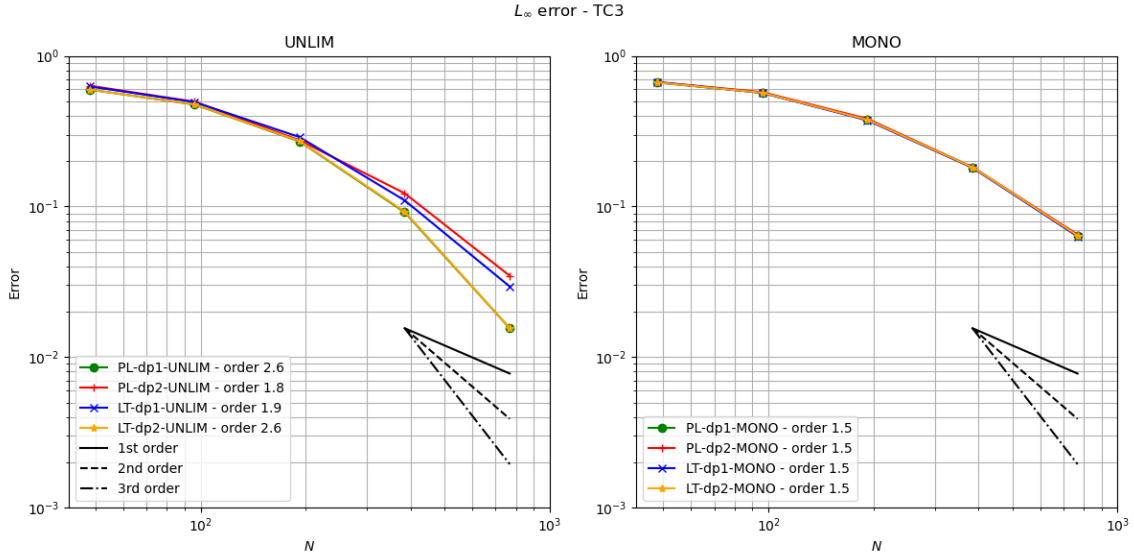**Figure 3.7:** L_∞ *error for the two Gaussian hills (Equation 3.31) with the velocity from Equation (3.32). Schemes using the unlimited PPM (UNLIM) are on the left, and the monotonic PPM (MONO) are on the right. The PL scheme with DP1 is in green, and with DP2 is in red. The LT scheme with DP1 is in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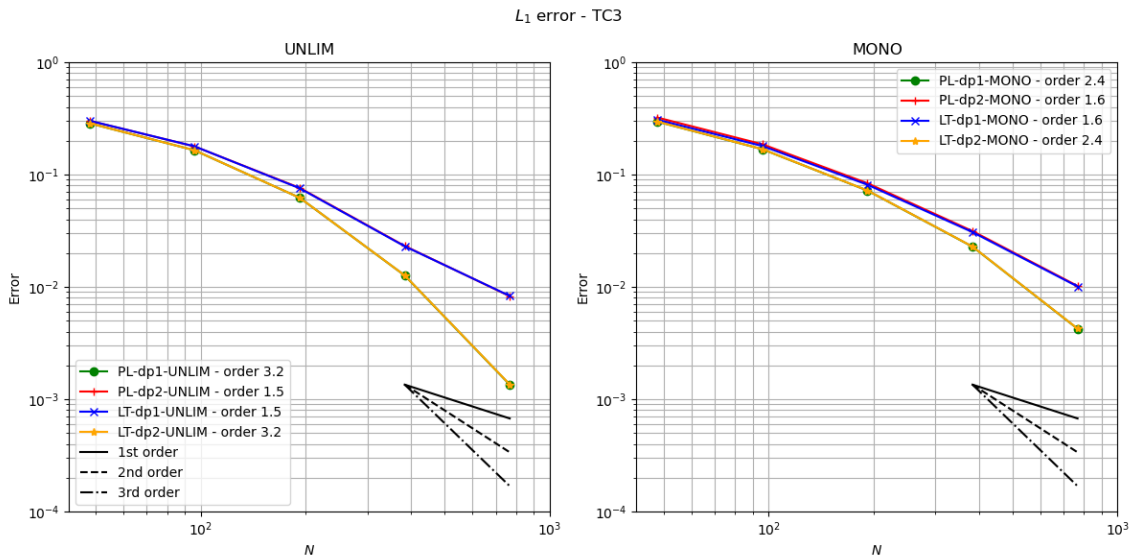

Figure 3.8: *Similar to Figure 3.7 but considering the L_1 error.*

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.

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

(c) $t = 6$ *days.*   **(d)** $t = 8$ *days.*

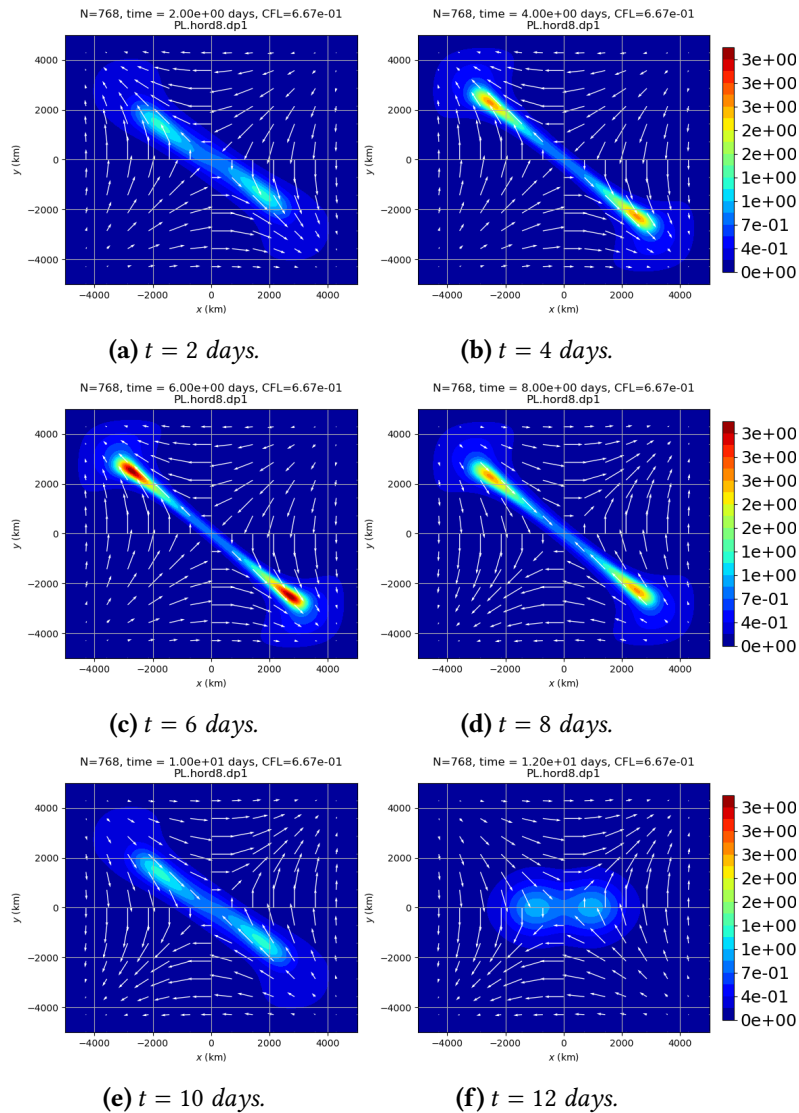**(e)** $t = 10$ *days.*   **(f)** $t = 12$ *days.*

Figure 3.9: *Similar to Figure 3.6 but using the wind from Equation (3.33). We are using the PL-DP1-MONO scheme, where MONO is the monotonic PPM.*

We employ time steps $\Delta t^{(k)} = \frac{14400}{2^k}$ for $k = 0,\dots,4$, 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.

We can observe from Figure 3.10 that for the unlimited PPM, PL-DP1 has the bigger error, while LT-DP2 has the smaller error and the highest convergence rate. However, when considering the monotonic PPM 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 monotonic PPM, 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 the unlimited and monotonic PPM, in both L_∞ and L_1 norms.

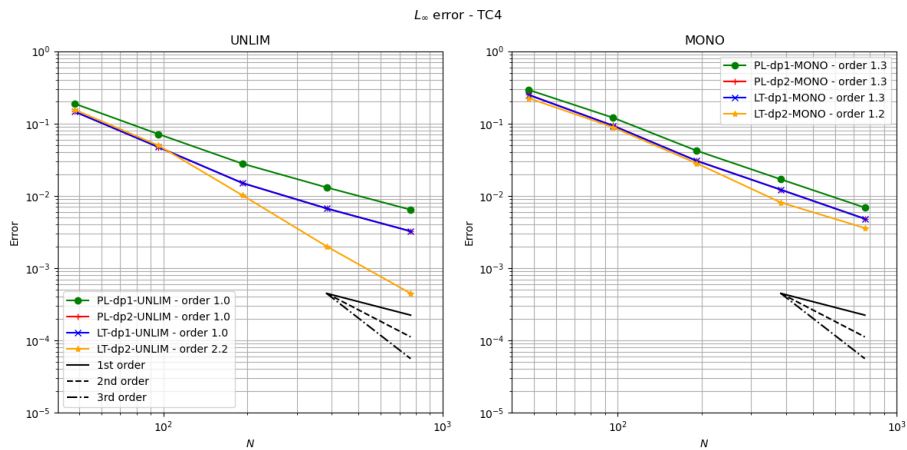**Figure 3.10:** *L_∞ error for the two Gaussian hills (Equation 3.31) with the velocity from Equation (3.33). Schemes using the unlimited PPM (UNLIM) are on the left, and monotonic PPM (MONO) are on the right. The PL scheme with DP1 is in green, and with DP2 is in red. The LT scheme with DP1 is in blue, and with DP2 is in yellow.*
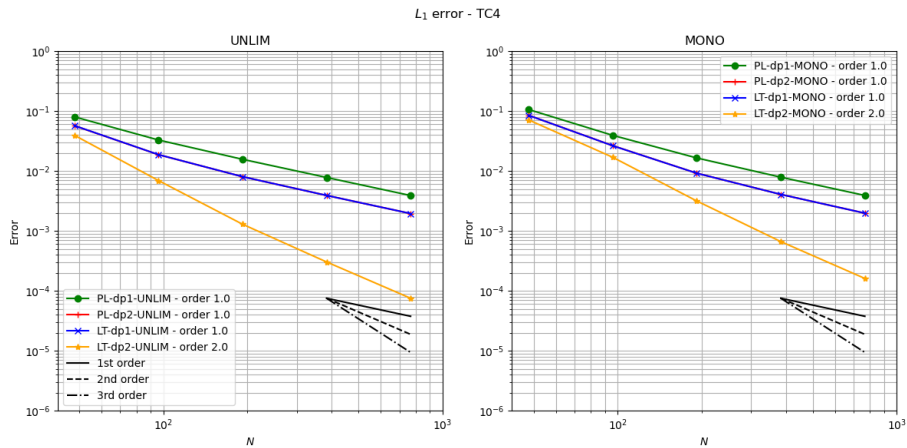

Figure 3.11: *Similar to Figure 3.10 but considering the L_1 error.*

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 unstructured 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 (Peixoto, 2016; Peixoto & Barros, 2013; Weller et al., 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 (Peixoto & Barros, 2014; 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 less 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 for 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, we 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:

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

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

$$\Gamma_1(x, y) = \frac{R}{\sqrt{1 + x^2 + y^2}}(1, x, y),$$

$$\Gamma_2(x, y) = \frac{R}{\sqrt{1 + x^2 + y^2}}(-x, 1, y),$$

$$\Gamma_3(x, y) = \frac{R}{\sqrt{1 + x^2 + y^2}}(-1, -x, y),$$

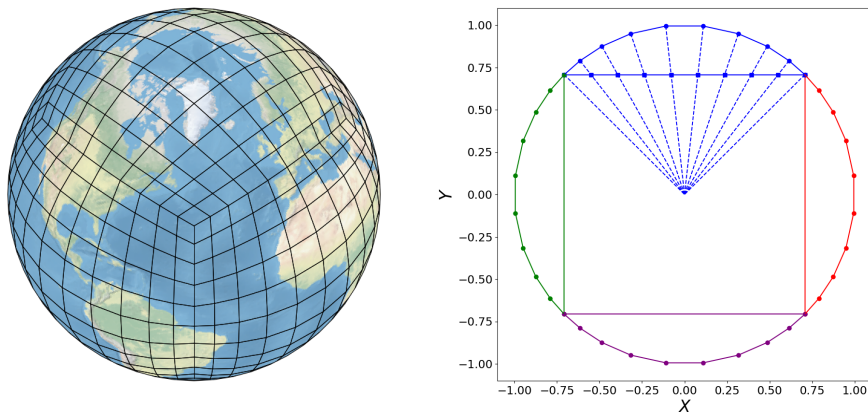$$\Gamma_4(x, y) = \frac{R}{\sqrt{1 + x^2 + y^2}}(x, -1, y),$$

$$\Gamma_5(x, y) = \frac{R}{\sqrt{1 + x^2 + y^2}}(-y, x, 1),$$

$$\Gamma_6(x, y) = \frac{R}{\sqrt{1 + x^2 + y^2}}(y, x, -1).$$

(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 $\{\Gamma_p, p = 1, \dots, 6\}$ 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).
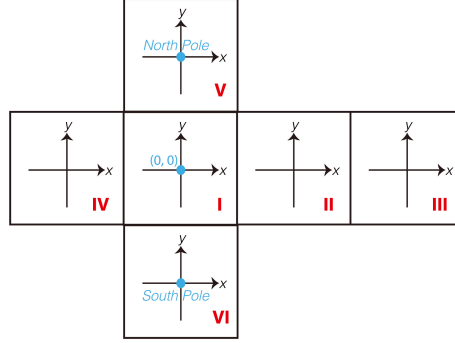

Figure 4.2: *Cubed-sphere panels definition and orientation. Figure taken from Jung et al. (2019).*

The derivative of the maps Γ_p are given by:

$$d\Gamma_1(x, y) = \frac{R}{(1 + x^2 + y^2)^{3/2}} \begin{bmatrix} -x & -y \\ 1 + y^2 & -xy \\ -xy & 1 + x^2 \end{bmatrix}, \quad d\Gamma_2(x, y) = \frac{R}{(1 + x^2 + y^2)^{3/2}} \begin{bmatrix} -(1 + y^2) & xy \\ -x & -y \\ -xy & 1 + x^2 \end{bmatrix},$$

$$d\Gamma_3(x, y) = \frac{R}{(1 + x^2 + y^2)^{3/2}} \begin{bmatrix} x & y \\ -(1 + y^2) & xy \\ -xy & 1 + x^2 \end{bmatrix}, \quad d\Gamma_4(x, y) = \frac{R}{(1 + x^2 + y^2)^{3/2}} \begin{bmatrix} 1 + y^2 & -xy \\ x & y \\ -xy & 1 + x^2 \end{bmatrix},$$

$$d\Gamma_5(x, y) = \frac{R}{(1 + x^2 + y^2)^{3/2}} \begin{bmatrix} xy & -(1 + x^2) \\ 1 + y^2 & -xy \\ -x & -y \end{bmatrix}, \quad d\Gamma_6(x, y) = \frac{R}{(1 + x^2 + y^2)^{3/2}} \begin{bmatrix} -xy & 1 + x^2 \\ 1 + y^2 & -xy \\ x & y \end{bmatrix}.$$

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

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

Notice that the matrix

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

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

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

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

called metric term, is then given by:

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

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 $\Psi_p : [-\alpha, \alpha] \times [-\alpha, \alpha] \to \mathbb{S}^2_R$, given by

$$\Psi_p(x,y) := \Gamma_p\big(\beta(x), \beta(y)\big).$$

It follows from the chain rule that:

$$d\Psi_p(x,y) = d\Gamma_p\big(\beta(x), \beta(y)\big) \cdot \text{diag}\big(\beta'(x), \beta'(y)\big),$$

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

$$\partial_x\Psi_p(x,y) = \beta'(x) \cdot \partial_x\Gamma_p\big(\beta(x), \beta(y)\big),$$
$$\partial_y\Psi_p(x,y) = \beta'(y) \cdot \partial_y\Gamma_p\big(\beta(x), \beta(y)\big).$$

The metric tensor of Ψ_p is defined as G_Γ in Equation (4.1):

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

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

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

which may also be expressed as

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

where α is the angle between $\partial_x\Psi_p$ and $\partial_y\Psi_p$ that satisfies

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

4.2 Cubed-sphere grids

Now that we have established 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 Ψ_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 Ψ_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**.
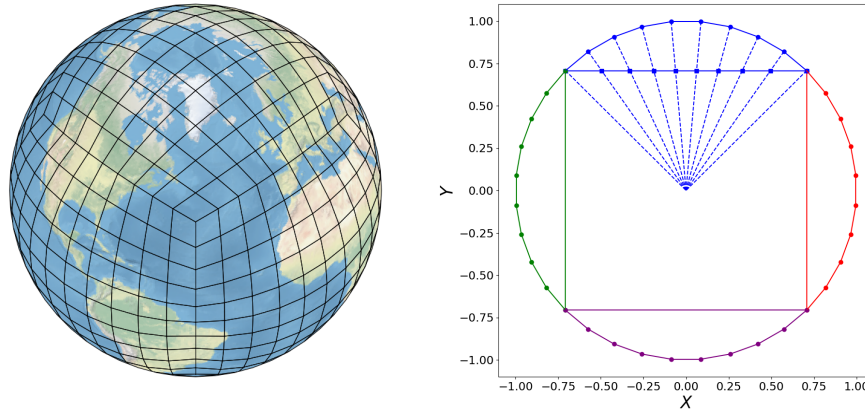
(a) *Gridlines of the cube to the sphere equian-gular mapping* **(b)** *Cube and sphere equiangular mapping for $Z = 0$.*

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

Another cubed-sphere mapping is the equi-edge reported by X. Chen (2021) using $\beta(x) = \sqrt{2} \tan x$ and $\alpha = \arcsin\left(\frac{1}{\sqrt{3}}\right)$. It is worth noting that while this mapping technique had been used previously in FV3, it was not formally documented until the work of X. Chen (2021). Figure 4.4 illustrates the equi-edge mapping.
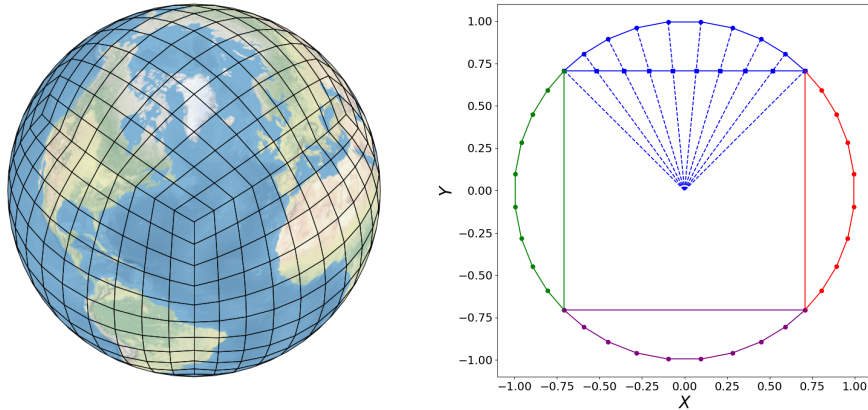
(a) *Gridlines of the cube to the sphere equi-edge mapping* **(b)** *Cube and sphere equi-edge mapping for $Z = 0$.*

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; 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 $\Omega_{\Delta x, \Delta y} = (\Omega_{ij})_{i,j=-\nu+1,\dots,N+\nu}$, 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:

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

The cubed-sphere grid refers to the collection of control volumes $(\Omega_{ijp})_{i,j=-\nu+1,\dots,N+\nu}^{p=1,\dots,6}$. 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 $|\Omega_{ij}|$. This gives a good approximation to the cell diameters, as a cell on the cubed-sphere has similar lengths. The control volume

area is given by:

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

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 equi-edge (g0), equidistant (g1), and equiangular (g2) grids 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, the equidistant grid (g1) is the less uniform, while the equiangular grid is the most uniform grid. The equi-edge grid is more uniform than the equidistant grid, but the maximum/minimum ratio of the areas is almost 2.3. Despite this, the equi-edge grid is the operational grid in some applications of FV3 (X. Chen, 2021; 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

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).*

less grid imprinting due to its greater uniformity near the cubed edges. Therefore, in this thesis, we shall constrain our attention only to the equi-edge and equiangular grids, since the equiangular grid is ideally more uniform and the equi-edge grid is currently used in FV3. In Figure 4.5, we illustrate the areas of both grids, equi-edge (g0) and equiangular (g2). We can observe that the areas of the equi-edge grid exhibit a higher gradient near the cube corners, while the equiangular grid appears to have a higher gradient near the middle of the cube edges.

There are four types of grid points on the cubed-sphere that we need to compute: the center, corners, right-left edge midpoints, and up-down edge midpoints. The locations of these points are illustrated in Figure 4.6 for the equiangular grid (g2).

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 corner points, as the name suggest, represent the corners of the control volume, namely

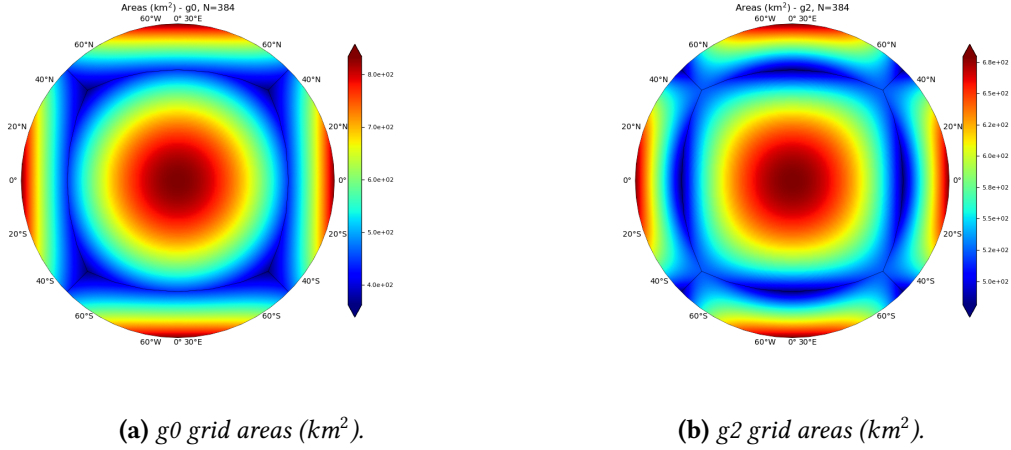$$\Psi^c_{i+\frac{1}{2},j+\frac{1}{2},p} := \Psi_p(x_{i+\frac{1}{2}}, y_{j+\frac{1}{2}}). \tag{4.4}$$

(a) *g0 grid areas (km²).*   **(b)** *g2 grid areas (km²).*

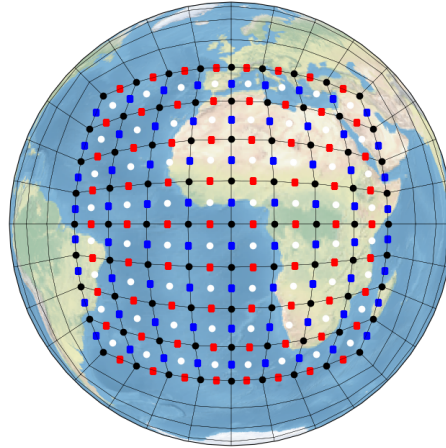**Figure 4.5:** *Areas for the grid equi-edge (g0) and equiangular grid (g2) using N = 384.*

Figure 4.6: *Illustration of the center (white), corner (black), right-left edge (blue) and up-down edge (red) points for the equiangular grid (g2) with N = 10.*

Similarly, the cell centers are:
$$\Psi^c_{ijp} := \Psi_p(x_i, y_j). \tag{4.5}$$

The right-left edge points are the midpoints of the edge in the y-direction, namely:
$$\Psi^c_{i+\frac{1}{2},j,p} := \Psi_p(x_{i+\frac{1}{2}}, y_j). \tag{4.6}$$

The up-down edge points are the midpoints of the edge in the x-direction, namely:
$$\Psi^c_{i,j+\frac{1}{2},p} := \Psi_p(x_i, y_{j+\frac{1}{2}}). \tag{4.7}$$

The grids equi-edge (g0) and equiangular (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., 2005b; Ullrich et al., 2010).

Another way to compute the grid points, which is used in FV3, is to compute the corner points as before and obtain the center, right-left edge and up-down edge points using the spherical midpoints. These points shall be denoted using a superscript 's'. The corner points in this case are:

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

The center points are computed by averaging the values of 4 corner points:

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

Similarly, the right-left edge points are obtained by averaging the values of 2 corner points.

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

and the up-down edge points are also given by the average the values of 2 corner points:

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

The grids equi-edge (g0) and equiangular (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:

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

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

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

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 center, right-left edge, and up-down edge 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 in this Subsection, we are going to omit the superscripts 's' and 'c', because what is described here has the same meaning for both midpoint formulations. Finally, we introduce the following geodesic distances in x and y directions, respectively,

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

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

where $d(P, Q) = R \arccos(\langle P, Q \rangle)$, for $P, Q \in \mathbb{S}_R^2$, 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:

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

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

Hence, their midpoint approximations are defined as:

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

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

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 corner duo-grid points are generated by computing the mappings $\mathbf{\Psi}_p$ 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 center, right-left edge, and up-down edge points are computed analogously. In Figure 4.7, we illustrate the duo-grid points obtained for both equi-edge and equiangular grids. We can observe in Figure 4.7c that the corner duo-grid points of the equiangular grid are aligned on common geodesics. This property has been known since the work of Ronchi et al. (1996), and similarly, it holds for center, right-left edge, and up-down edge 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 equi-edge 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 $g = 1, 2, \ldots, \nu$, 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 equi-edge (g0) aligns with the geodesics of neighboring panels just as the equiangular grids (g2) (Figure 4.7c). Notice, however, that the equi-edge corner points will no longer be equally spaced in terms of its x and y coordinates, in contrast to equiangular grid, where the x and y coordinates of the corner points are uniformly spaced. This may require special attention from numerical schemes using equi-edge near edges due to the loss of uniformity.

4.2.6 Tangent vectors on the sphere

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

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

We are going to consider three ways to represent an element of $\mathbb{S}_R^2$: 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 $\boldsymbol{u} \; : \; \mathbb{S}_R^2 \to \mathbb{R}^3$ is tangent on the sphere if $\boldsymbol{u}(P) \in T_P \mathbb{S}_R^2, \forall P \in \mathbb{S}_R^2$.

Conversions between latitude-longitude and contravariant coordinates

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

$$d\boldsymbol{\Pi}(\lambda, \phi) = R \begin{bmatrix} -\cos\phi \sin\lambda & -\sin\phi \cos\lambda \\ \cos\phi \cos\lambda & \sin\phi \sin\lambda \\ 0 & \cos\phi \end{bmatrix}. \tag{4.30}$$
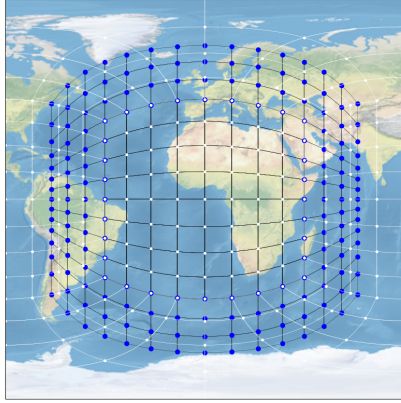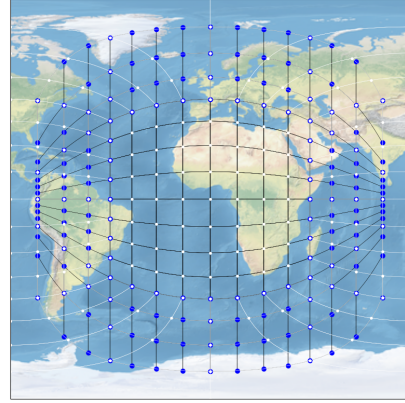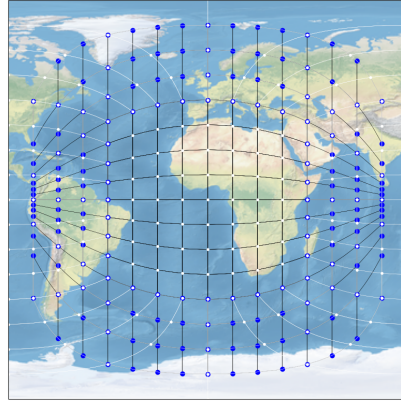
(a) *g0 duo-grid.*

(b) *Mirrored g0 duo-grid.*

(c) *g2 duo-grid.*

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

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

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

We normalize the vectors $\partial_\lambda \Pi$ and $\partial_\phi \Pi$ 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 $\boldsymbol{u} : \mathbb{S}_R^2 \to \mathbb{R}^3$ 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:

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

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

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

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

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:

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

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:

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

It is easy to see that:

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

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 (Harris et al., 2016), where explicit expressions of the exact, non-normalized tangent vectors are either not 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 $\boldsymbol{u} : \mathbb{S}_R^2 \to \mathbb{R}^3$ on the sphere. Its contravariant representation is given by Equation (4.34). The covariant components $(\mathfrak{U}, \mathfrak{V})$ are given by:

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

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

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

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

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

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

It is easy to see that:

$$U(x, y, p) = \frac{\mathfrak{U}(x, y, p)}{\|\partial_x \boldsymbol{\Psi}_p(x, y)\|}, \quad V(x, y, p) = \frac{\mathfrak{V}(x, y, p)}{\|\partial_y \boldsymbol{\Psi}_p(x, y)\|}. \tag{4.46}$$

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

$$\begin{bmatrix} U(x, y, p) \\ V(x, y, p) \end{bmatrix} = \begin{bmatrix} 1 & \langle \boldsymbol{e}_x, \boldsymbol{e}_y \rangle \\ \langle \boldsymbol{e}_x, \boldsymbol{e}_y \rangle & 1 \end{bmatrix} \begin{bmatrix} u(x, y, p) \\ v(x, y, p) \end{bmatrix}. \tag{4.47}$$

Recall that

$$\langle \boldsymbol{e}_x, \boldsymbol{e}_y \rangle(x, y, p) = \cos \alpha(x, y, p), \tag{4.48}$$

where $\alpha(x, y, p)$ is the angle between $\boldsymbol{e}_x$ and $\boldsymbol{e}_y$, 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 (4.47). Notice that combining Equation (4.47) 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 $CS_N = \mathbb{R}^{(N+\nu)\times(N+\nu)\times 6}$ to represent grid functions on the cubed-sphere at cell centers. These grid functions, as they are defined at cell centers and following the nomenclature of Arakawa and Lamb (1977), define what we call an A-grid field or function. We define the average values of a function q with the aid of the metric term $\sqrt{g}(x,y)$ at time t:

$$Q_{ijp}(t) = \frac{1}{|\Omega_{ij}|} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}} q(x,y,p,t)\sqrt{g}(x,y)\,dx\,dy. \qquad (4.49)$$

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 $q^n \in CS^N$, which represents the grid function q evaluated at the discrete points. In other words, $q_{ijp}^n = q(x_i, y_j, p, t^n)$, where $i, j = -\nu + 1, \ldots, N+\nu$, and $p = 1, \ldots, 6$. Furthermore, we use the notations $q_{i+\frac{1}{2},j,p}^n = q(x_{i+\frac{1}{2}}, y_j, t^n)$ for $i = -\nu, \ldots, N+\nu$ and $j = -\nu+1, \ldots, N+\nu$ to represent q at right-left edge points. Similarly, we use $q_{i,j+\frac{1}{2},p}^n = q(x_i, y_{j+\frac{1}{2}}, t^n)$ for $i = -\nu+1, \ldots, N+\nu$ and $j = -\nu, \ldots, N+\nu$ to represent q at the up-down edge 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:

$$Q_{\times,j,p} := (Q_{-\nu+1,j,p}, \ldots, Q_{N+\nu,j,p}) \in \mathbb{R}_\nu^N,$$
$$Q_{i,\times,p} := (Q_{i,-\nu+1,p}, \ldots, Q_{i,N+\nu,p}) \in \mathbb{R}_\nu^N.$$

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. The p-norm for $q \in CS^N$, is defined as:

$$\|q\|_{p,CS^N} = \begin{cases} \left(\sum_{k=1}^{6} \sum_{i=1}^{N} \sum_{j=1}^{N} |q_{ijk}|^p |\Omega_{ij}| \right)^{\frac{1}{p}} & \text{if } 1 \le p < \infty, \\ \max_{i=1,\ldots,N, j=1,\ldots,N, k=1,\ldots,6} |q_{ijk}| & \text{if } p = \infty. \end{cases} \qquad (4.50)$$

4.3.2   Ghost cells scalar field interpolation

Let us consider a function $q\ :\ \mathbb{S}_R^2 \to \mathbb{R}$ given at the center points denoted by q_{ijp}, where $i, j = 1 \ldots, N$ and $p = 1, \ldots, 6$. That is, we are given an A-grid scalar field. Our objective is to estimate these values at positions outside the range $1, \ldots, N$, 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.

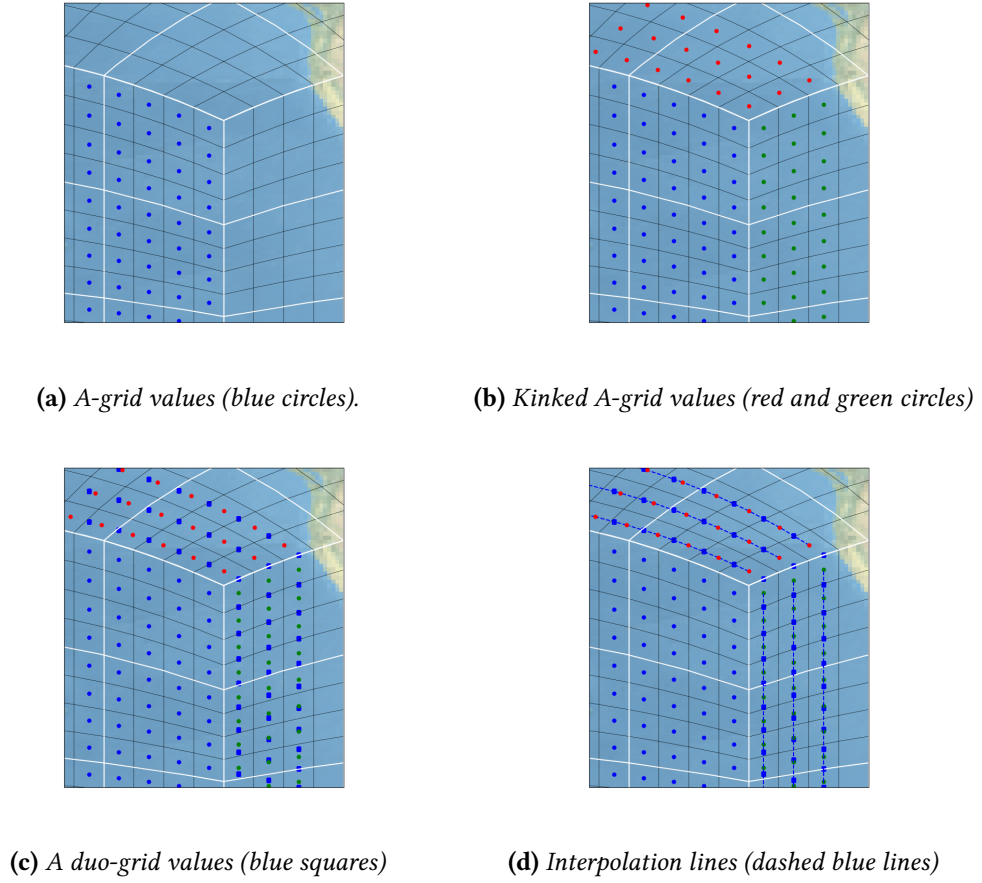To illustrate this process in Panel 1, we depict the values of q_{ijp} in Figure 4.8. The

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

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

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

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

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

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 located on a cube corner can be interpolated using the green and red points by connecting these points using geodesics and further extending the blue lines from Figure 4.8d.

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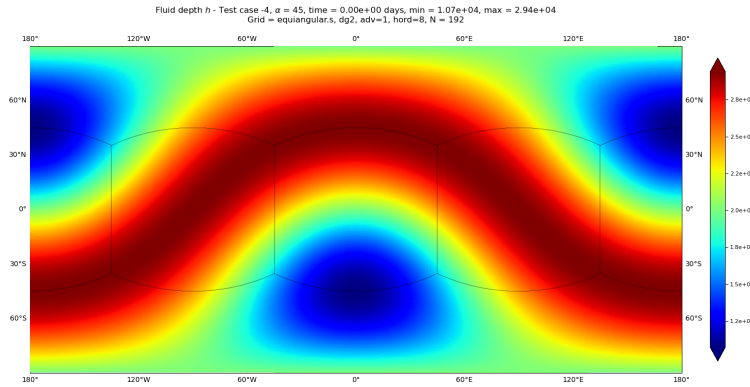
Figure 4.9: *Scalar field from Equation* (4.51).

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

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

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. However, when spherical midpoints are employed, we observe a reduction in accuracy by two orders, as indicated by the solid lines. Both dg1 and dg2 are very similar in this case. 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 equi-edge grid (g0) yields slightly better results than the equiangular grid (g2) for the cube midpoints formulation.

At last, we point out that B-grid fields, as defined by Arakawa and Lamb, 1977 represents a scalar field at the corner points, namely, denoted by $q_{i+\frac{1}{2}, j+\frac{1}{2}, p}$, where $i, j = 0 \dots, N$ and $p = 1, \dots, 6$. We can estimate these values at positions outside the range $0, \dots, N$ using an analogous procedure to the one described here for A-grid fields.
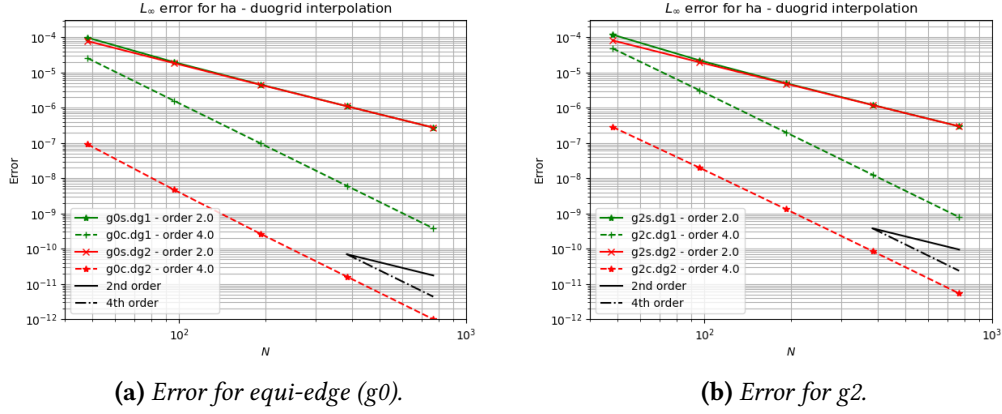
(a) *Error for equi-edge (g0).*   **(b)** *Error for g2.*

Figure 4.10: *Error for the duo-grid interpolation of the scalar field of* (4.51) *for the equi-edge grid (g0, left) and the equiangular grid (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 $\boldsymbol{u} : \mathbb{S}^2_R \to \mathbb{R}^3$. We also have its normalized contravariant normal components at the edge 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 covariant wind following Arakawa and Lamb (1977) (Figure 4.11a). The C-grid covariant wind is defined similarly using (U, V).

Our objective is to obtain the values

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

This problem arises when we apply the dimension splitting method on each panel of the cubed-sphere. We point out that the method we are going to describe considers the C-grid covariant wind, but it works very similarly for the C-grid contravariant 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 center points required for the ghost cells interpolation (Figure 4.11b). Specifically, we need the values:
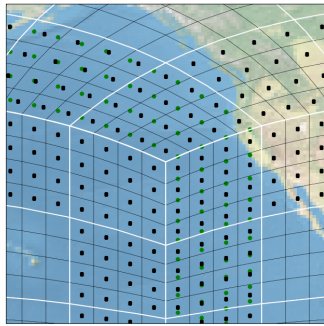
$$u_{1+k,j,p}, v_{1+k,j,p} \quad \text{for} \quad j = 1, \ldots, N, \quad k = 0, \ldots, \nu,$$
$$u_{N-k,j,p}, v_{N-k,j,p} \quad \text{for} \quad j = 1, \ldots, N, \quad k = 0, \ldots, \nu,$$
$$u_{i,1+k,p}, v_{i,1+k,p} \quad \text{for} \quad i = 1, \ldots, N, \quad k = 0, \ldots, \nu,$$
$$u_{i,N-k,p}, v_{i,N-k,p} \quad \text{for} \quad i = 1, \ldots, N, \quad k = 0, \ldots, \nu.$$
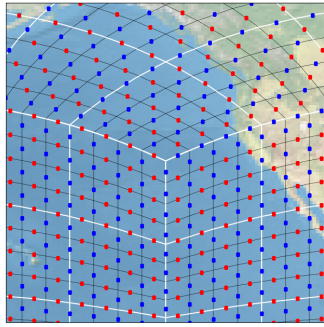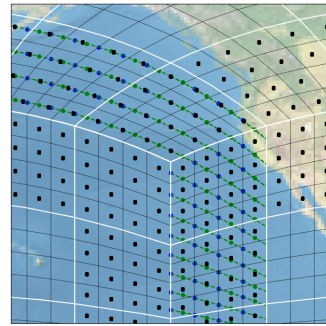
(a) *C-grid wind values (blue and red squares).*
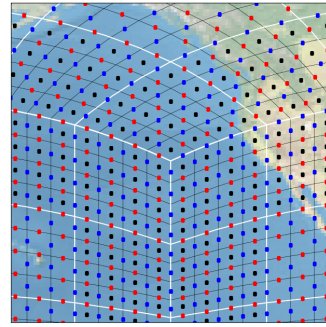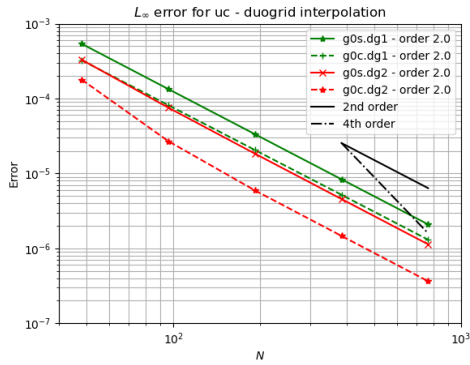

(b) *A-grid wind (black circles)*

(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.*

(a) *Error for equi-edge grid (g0).*
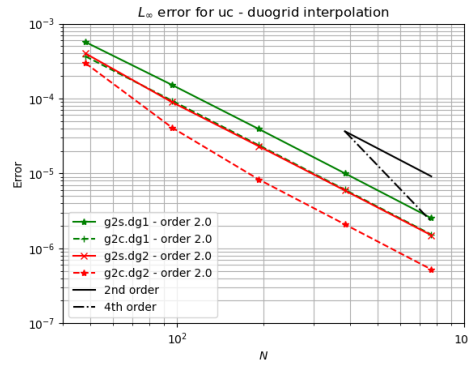
(b) *Error for equiangular grid (g2).*

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

We apply a simple linear interpolation to remap u and v to the center points (Figure 4.11). Then we obtain two A-grid scalar fields, and we may proceed as before to fill the ghost cell values. However, we are going to use one extra layer of A-grid values since they will be needed for re-interpolation to the edge points. For instance, for 3 layers of ghost cells, we need 4 layers of A duo-grid values to fill the C-grid wind at the duo-grid edge

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 (4.47) and (4.41). This conversion avoids any coordinate system discontinuity, as this interpolation is performed only close to the cube edges and far from the poles. 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 (4.47).

We will consider the following rotated zonal field, as a numerical test, based on 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 edge midpoints. The errors are presented in Figure 4.12, along with the convergence rate for different schemes employed in the ghost cell center interpolation. We emphasize that, since we utilize a linear interpolation to retrieve the A-grid wind components from the edge midpoints, as well as in the interpolation from the center duo-grid points to edge 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, equi-edge (g0) performs slightly better than the equiangular (g2) grid.

We point out that we define the D-grid contravariant wind as the grid functions namely $v_{i+\frac{1}{2},j,p}$ for $i = 0, \dots, N$ and $j = 1, \dots, N$, as well as $u_{i,j+\frac{1}{2},p}$ for $i = 1, \dots, N$ and $j = 0, \dots, N$, following again Arakawa and Lamb (1977). On the D-grid wind, we have the tangential components of the wind, whereas on the C-grid wind, we have the normal components of the wind. The D-grid wind at duo-grid points may be obtained using a very similar process as described before. The D-grid covariant wind is defined similarly using (U, V).

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 edge points denoted by

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

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:

$$q_{1,j,p}^{L,x} = \frac{1}{2}\left(3Q_{1,j,p} - Q_{2,j,p} \right),$$

$$q_{N,j,p}^{R,x} = \frac{1}{2}\left(3Q_{N,j,p} - Q_{N-1,j,p} \right),$$

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

$$q_{1,j,p}^{R,x} = \frac{1}{14}\left(3Q_{1,j,p} + 11Q_{2,j,p} - 2(Q_{3,j,p} - Q_{1,j,p}) \right),$$

$$q_{2,j,p}^{L,x} = q_{1,j,p}^{R,x},$$

$$q_{N,j,p}^{L,x} = \frac{1}{14}\left(3Q_{N,j,p} + 11Q_{N-1,j,p} - 2(Q_{N-2,j,p} - Q_{N,j,p}) \right),$$

$$q_{N-1,j,p}^{R,x} = q_{N,j,p}^{L,x},$$

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:

$$e_{i-\frac{1}{2},j,p} = (|q_{i-\frac{1}{2},j,p} - q_{ijp}^{L,x}|)/|q_{i-\frac{1}{2},j,p}|,$$

$$e_{i+\frac{1}{2},j,p} = (|q_{i+\frac{1}{2},j,p} - q_{ijp}^{R,x}|)/|q_{i+\frac{1}{2},j,p}|,$$

$$e_{i,j-\frac{1}{2},p} = (|q_{i,j-\frac{1}{2},p} - q_{ijp}^{L,y}|)/|q_{i,j-\frac{1}{2},p}|,$$

$$e_{i,j+\frac{1}{2},p} = (|q_{i,j+\frac{1}{2},p} - q_{ijp}^{R,y}|)/|q_{i,j+\frac{1}{2},p}|,$$

$$e_{ijp} = \max\{e_{i-\frac{1}{2},j,p}, e_{i+\frac{1}{2},j,p}, e_{i,j-\frac{1}{2},p}, e_{i,j+\frac{1}{2},p}\},$$

$$E = \max\{e_{ijp}\}.$$

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 the limited PPM (MONO). We depict the errors in Figure 4.13. We can observe that the error for the equi-edge grid (g0) is only slightly better than the error for the equiangular grid (g2). Also, the errors for the cube midpoint formulation and the spherical midpoint formulation are essentially the same when using dg2.
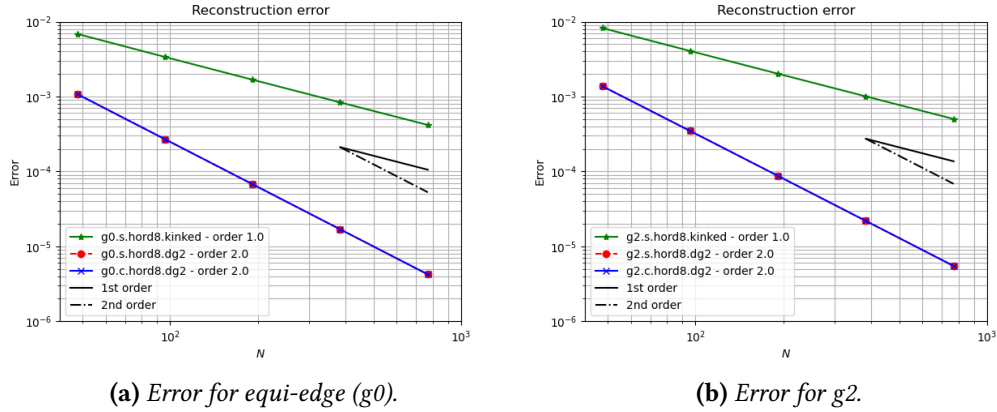

(a) *Error for equi-edge (g0).*   **(b)** *Error for g2.*

Figure 4.13: *Relative error for the PPM reconstruction using scalar field from Equation (4.51). Equi-edge (g0) grid results are on the left, and equiangular grid (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 the limited PPM (MONO).*

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 the PPM reconstruction 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 equi-edge (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 equiangular 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 equi-edge grid (Figure 4.14a) than in the equiangular 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 grid being the most uniform. The corner points are generated using the equiangular and equi-edge mappings; however, the center and edge points may be generated using a cubed-sphere mapping or using midpoints based on spherical midpoints of the corner points (Section 4.2.4).

We observed that the equiangular cubed-sphere ghost cells, obtained by extending the gridlines, have 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 the geodesic distances and the other based on local coordinate distances. Overall, the method based on local coordinate distances showed that it introduces 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 the limited PPM (MONO) 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, which shows 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 on the duo-grid interpolation step, where we attain the expected orders of interpolation only for the cube midpoints formulation. However, in the reconstruction from center to edge midpoints, both formulations have essentially the same errors. Hence, we expect that both midpoint formulations should lead to the very similar results.
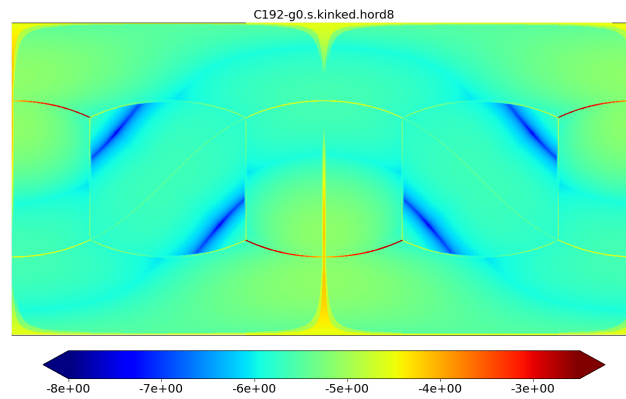
(a) *g0.s with kinked.*
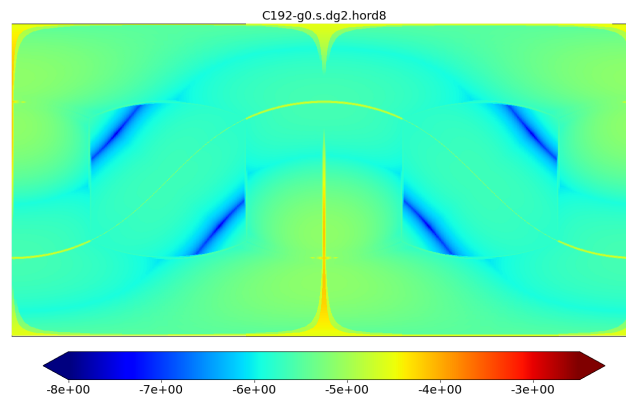
(b) *g0.s with dg2.*
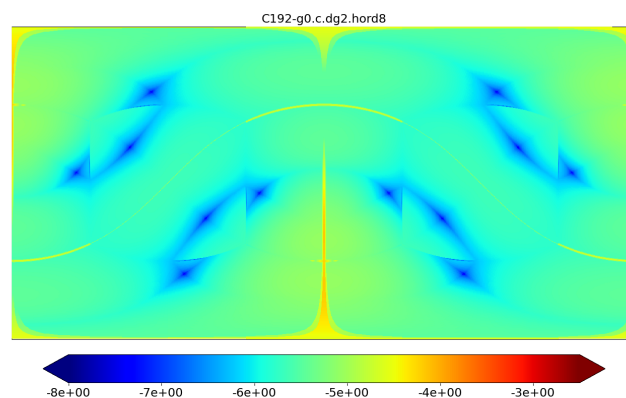
(c) *g0.c with dg2.*

Figure 4.14: *The logarithm in base 10 of the errors of the edge grid points reconstruction from the A-grid field given by Equation* (4.51), *using the equi-edge grid (g0) with N = 192.*
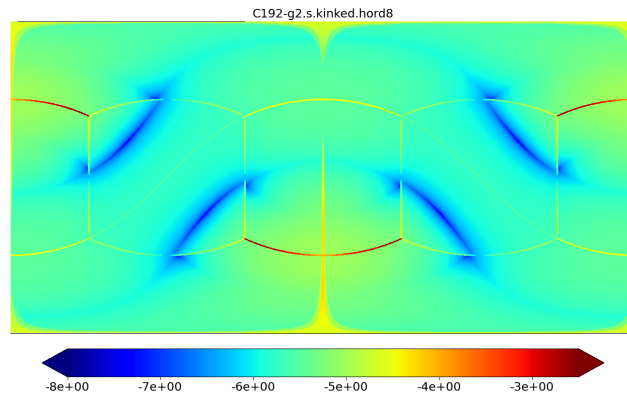
(a) *g0.s with kinked.*
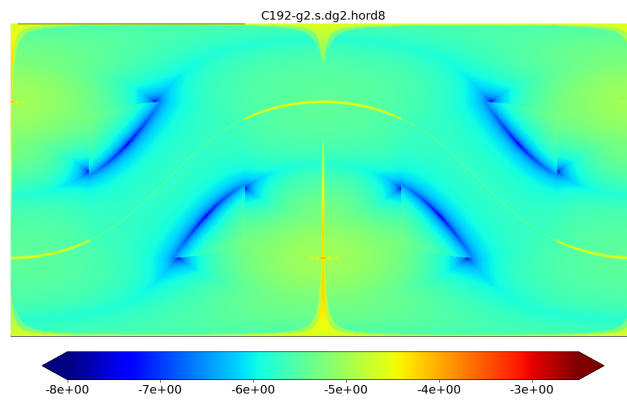
(b) *g0.s with dg2.*
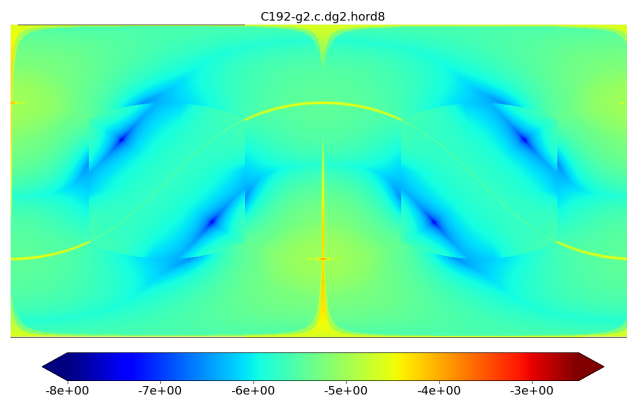
(c) *g0.c with dg2.*

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

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 solve the advection equation. Section 5.5 presents the final thoughts.

5.1 Cubed-sphere advection equation in the 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 Sections 4.1 and 4.2. 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 CS_N grid function $\mathfrak{D}^n$ as the exact divergence of $q\boldsymbol{u}$ at the cell centers, namely

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

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 (4.49)). Since the metric term does not depend on t, we may rewrite Equation (5.1) as

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

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 (u, 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:*

$$\int_{x_1}^{x_2} \int_{y_1}^{y_2} (q\sqrt{\mathfrak{g}})(x, y, p, t)\, dx\, dy = \int_{x_1}^{x_2} \int_{y_1}^{y_2} (q\sqrt{\mathfrak{g}})(x, y, p, t)\, dx\, dy$$

$$- \int_{t_1}^{t_2} \int_{y_1}^{y_2} \left((uq\sqrt{\mathfrak{g}})(x_2, y, t) - (uq\sqrt{\mathfrak{g}})(x_1, y, t) \right) dy\, dt$$

$$- \int_{t_1}^{t_2} \int_{x_1}^{x_2} \left((vq\sqrt{\mathfrak{g}})(x, y_2, t) - (vq\sqrt{\mathfrak{g}})(x, y_1, t) \right) dx\, dt.$$

$\forall [x_1, x_2] \times [y_1, y_2] \times [t_1, t_2] \subset \Omega \times [0, T]$, *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 Problem 5.2 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*

$\max_{i,j,p}\{|\mathfrak{u}^n_{i+\frac{1}{2},j,p}|\}\frac{\Delta t}{\Delta x}$ and $\max_{i,j,p}\{|\mathfrak{v}^n_{i,j+\frac{1}{2},p}|\}\frac{\Delta t}{\Delta y}$, 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). The time-averaged CFL number $\tilde{c}^{x,n}_{i+\frac{1}{2},j}$ shall be discussed in details in Section 5.3.3.

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:

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

In this case, the flux reads:

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

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 $q_{ij} = \bar{q}$, for a constant $\bar{q}$, and when the wind is divergence-free. This occurs because when we use mt1, we have

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

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 (2.68))

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

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 the unlimited PPM (Section 2.4.1) or the monotonic (Section 2.4.2). Recall that when using the mt1 scheme, we compute the flux for $\psi = \sqrt{\mathfrak{g}}q$. We point out that, in this case, the perturbation values (Equations (2.66) and (2.67)) use the exact expression of the metric term, namely:

$$b^L_{ij} = \sqrt{\mathfrak{g}}_{i+\frac{1}{2},j}q^{L,x}_{ij} - \sqrt{\mathfrak{g}}_{ij}q^n_{ij}, \tag{5.18}$$

$$b^R_{ij} = \sqrt{\mathfrak{g}}_{i,j+\frac{1}{2}}q^{R,y}_{ij} - \sqrt{\mathfrak{g}}_{ij}q^n_{ij}, \tag{5.19}$$

where the edge values reconstruction values $q^{R,x}_{ij}$ and $q^{L,x}_{ij}$ are computed using the UNLIM or MONO 1D PPM scheme in the x and y directions, respectively. Hence, when a monotonic filter is applied, it is applied only to the field q instead of the field $\sqrt{\mathfrak{g}}q$, which is important

to preserve monotonicity.

Therefore, we may rewrite Equation (5.12) as

$$q_{ij}^{x,n+1} = q_{ij}^n + \mathbf{F}_{ij}[q^n, \tilde{c}^{x,n}], \tag{5.20}$$

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

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

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

$$\mathcal{A}^x_{i+\frac{1}{2},j} = \begin{cases} \hat{\delta}x_{i+\frac{1}{2},j}\hat{\delta}y_{i+\frac{1}{2},j} \sin \alpha_{i+\frac{1}{2},j} \tilde{c}^{x,n}_{i+\frac{1}{2},j}, & \text{for mt0,} \\ \Delta x \Delta y \tilde{c}^{x,n}_{i+\frac{1}{2},j} & \text{for mt1,} \end{cases} \tag{5.21}$$

where we have made use of the definitions of $\hat{\delta}x_{i+\frac{1}{2},j}$ and $\hat{\delta}y_{i,j+\frac{1}{2}}$ (Equation (4.19)) and the metric term relation given in Equation (4.2). The PPM fluxes are

$$\mathcal{F}^{PPM,x}_{i+\frac{1}{2},j}[q^n; \tilde{c}^{x,n}] = \begin{cases} \mathfrak{F}^{PPM,x}_{i+\frac{1}{2},j}[\sqrt{\mathfrak{g}}q^n; \tilde{c}^{x,n}] & \text{for mt0,} \\ \mathfrak{F}^{PPM,x}_{i+\frac{1}{2},j}[q^n; \tilde{c}^{x,n}] & \text{for mt1.} \end{cases} \tag{5.22}$$

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

$$q_{ij}^{y,n+1} = q_{ij}^n + \mathbf{G}_{ij}[q^n, \tilde{c}^{y,n}], \tag{5.23}$$

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

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

and

$$\mathcal{A}^y_{i,j+\frac{1}{2}} = \begin{cases} \hat{\delta}x_{i,j+\frac{1}{2}}\hat{\delta}y_{i,j+\frac{1}{2}} \sin \alpha_{i,j+\frac{1}{2}} \tilde{c}^{y,n}_{i,j+\frac{1}{2}}, & \text{for mt0,} \\ \Delta x \Delta y \tilde{c}^{y,n}_{i,j+\frac{1}{2}}, & \text{for mt1,} \end{cases} \tag{5.24}$$

and the PPM fluxes are

$$\mathcal{F}^{PPM,y}_{i,j+\frac{1}{2}}[q^n; \tilde{c}^{y,n}] = \begin{cases} \mathfrak{F}^{PPM,y}_{i,j+\frac{1}{2}}[\sqrt{\mathfrak{g}}q^n; \tilde{c}^{y,n}] & \text{for mt0,} \\ \mathfrak{F}^{PPM,y}_{i+\frac{1}{2},j}[q^n; \tilde{c}^{y,n}] & \text{for mt1,} \end{cases} \tag{5.25}$$

where $\mathfrak{F}^{PPM,y}_{i,j+\frac{1}{2}}$ is the analogous of Equation (5.17) in y direction. 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).

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

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

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).*

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

$$
\begin{aligned}
q^{n+1} = q^n &+ \frac{1}{2}\mathbf{F}[q^n, \tilde{c}^{x,n}] + \frac{1}{2}\mathbf{G}[q^n, \tilde{c}^{y,n}] \\
&+ \frac{1}{2}\mathbf{F}\left[q^n + \mathbf{g}[q^n, \tilde{c}^{y,n}], \tilde{c}^{x,n}\right] + \frac{1}{2}\mathbf{G}\left[q^n + \mathbf{f}[q^n, \tilde{c}^{x,n}], \tilde{c}^{y,n}\right],
\end{aligned} \tag{5.26}
$$

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 at the edges as (recall the wind formulation in Section 4.2.6):

$$
c^{x,n}_{i+\frac{1}{2},j} = \mathfrak{u}^{x,n}_{i+\frac{1}{2},j}\frac{\Delta t}{\Delta x} = \mathfrak{u}^{x,n}_{i+\frac{1}{2},j}\frac{\Delta t}{\hat{\delta}x_{i+\frac{1}{2},j}}, \tag{5.27}
$$

$$
c^{y,n}_{i,j+\frac{1}{2}} = \mathfrak{v}^{y,n}_{i,j+\frac{1}{2}}\frac{\Delta t}{\Delta y} = v^{y,n}_{i,j+\frac{1}{2}}\frac{\Delta t}{\hat{\delta}y_{i,j+\frac{1}{2}}}. \tag{5.28}
$$

Therefore, the time-averaged CFL numbers may be computed using Equation (2.37). 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):

$$
c^{x,n}_{i+\frac{1}{2},j} = u^{x,n}_{i+\frac{1}{2},j}\frac{\Delta t}{\hat{\delta}x^*_{i+\frac{1}{2},j}}, \tag{5.29}
$$

$$
c^{y,n}_{i,j+\frac{1}{2}} = v^{y,n}_{i,j+\frac{1}{2}}\frac{\Delta t}{\hat{\delta}y^*_{i,j+\frac{1}{2}}}, \tag{5.30}
$$

where

$$
\hat{\delta}x^*_{i+\frac{1}{2},j} = \begin{cases} \hat{\delta}x_{ij}, & \text{if } u^{x,n}_{i+\frac{1}{2},j} \geq 0, \\ \hat{\delta}x_{i+1,j}, & \text{if } u^{x,n}_{i+\frac{1}{2},j} < 0, \end{cases} \tag{5.31}
$$

$$\hat{\delta}y^*_{i,j+\frac{1}{2}} = \begin{cases} \hat{\delta}y_{ij}, & \text{if } v^{y,n}_{i,j+\frac{1}{2}} \geq 0, \\ \hat{\delta}y_{i,j+1}, & \text{if } v^{y,n}_{i,j+\frac{1}{2}} < 0, \end{cases} \tag{5.32}$$

We point out that Equation (5.27) to (5.28) and Equations (5.29) to (5.30) are equivalent when the metric term is constant and equal to one, as on the Cartesian grid on the plane. Additionally, we could use Equations (5.29) to (5.30) for the DP2 scheme, but we observed that the results obtained on advection simulations using Equations (5.27) to (5.28) are much better, while Equations (5.29) to (5.30) 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.

Finally, when using DP1, the upwind CFL requires reformulation of the terms $\mathcal{A}^x_{i+\frac{1}{2},j}$ (Equation (5.21)) and $\mathcal{A}^y_{i,j\}+\frac{1}{2}}$ (Equation (5.24)) for mt0, namely:

$$\mathcal{A}^x_{i+\frac{1}{2},j} = \begin{cases} \hat{\delta}x^*_{i+\frac{1}{2},j}\hat{\delta}y_{i+\frac{1}{2},j} \sin\alpha_{i+\frac{1}{2},j}\tilde{c}^{x,n}_{i+\frac{1}{2},j}, & \text{for mt0,} \\ \Delta x\Delta y\tilde{c}^{x,n}_{i+\frac{1}{2},j} & \text{for mt1,} \end{cases} \tag{5.33}$$

and

$$\mathcal{A}^y_{i,j+\frac{1}{2}} = \begin{cases} \hat{\delta}x_{i,j+\frac{1}{2}}\hat{\delta}y^*_{i,j+\frac{1}{2}} \sin\alpha_{i,j+\frac{1}{2}}\tilde{c}^{y,n}_{i,j+\frac{1}{2}}, & \text{for mt0,} \\ \Delta x\Delta y\tilde{c}^{y,n}_{i,j+\frac{1}{2}}, & \text{for mt1.} \end{cases} \tag{5.34}$$

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:*

$$\mathbb{D}^n_{ijp}(q^n, \mathfrak{u}^n, \mathfrak{v}^n) = \frac{1}{\Delta t \sqrt{\mathfrak{g}_{ij}}} \left(\frac{\delta_i F^n_{ijp}}{\Delta x} + \frac{\delta_j G^n_{ijp}}{\Delta y} \right), \quad i = 1, \dots, N, \quad j = 1, \dots, M. \tag{5.35}$$

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

$$q^{n+1} = q^n - \Delta t \mathbb{D}^n(q^n, \mathfrak{u}^n, \mathfrak{v}^n). \tag{5.36}$$

When using the dimension splitting technique, it follows from Equation (5.26) that the discrete divergence may be expressed as:

$$\mathbb{D}^n = \frac{-1}{\Delta t} \left(\frac{1}{2}\mathbf{F}[q^n, \tilde{c}^{x,n}] + \frac{1}{2}\mathbf{G}[q^n, \tilde{c}^{y,n}] + \frac{1}{2}\mathbf{F}\left[q^n + \mathbf{g}[q^n, \tilde{c}^{y,n}], \tilde{c}^{x,n}\right] + \frac{1}{2}\mathbf{G}\left[q^n + \mathbf{f}[q^n, \tilde{c}^{x,n}], \tilde{c}^{y,n}\right] \right). \tag{5.37}$$

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

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

It follows from Equation (5.36) that:

$$M^{n+1} = M^n - \sum_{p=1}^{6} \sum_{i,j=1}^{N} \mathbb{D}_{ijp}^n |\hat{\Omega}_{ij}|.$$

Hence, to ensure mass conservation, we must ensure that

$$\sum_{p=1}^{6} \sum_{i,j=1}^{N} \mathbb{D}_{ijp}^n |\hat{\Omega}_{ij}| = 0.$$

This property is discrete version of

$$\int_{\mathbb{S}_R^2} \nabla \cdot (\boldsymbol{u}q)\, dS = 0,$$

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
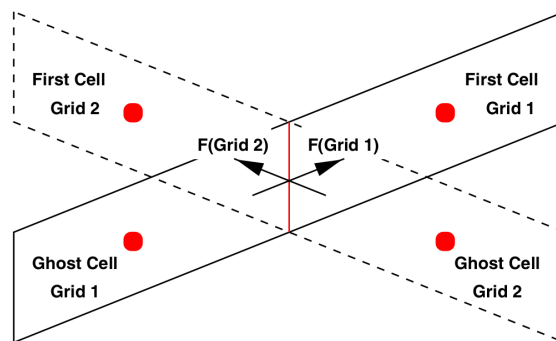
Figure 5.1: *Figure that illustrates the flux being computed twice on the cube edge, breaking the total mass conservation. Figure taken from Rossmanith (2006).*

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 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. In this section, we denote by r the geodesic distance from any point (ϕ, λ) to a fixed point (ϕ_0, λ_0), expressed as

$$r = 2R \arcsin \left(\sqrt{\sin^2 \left(\frac{\phi - \phi_0}{2} \right) + \cos \phi \cos \phi_0 \cos^2 \left(\frac{\lambda - \lambda_0}{2} \right)} \right), \qquad (5.38)$$

where R is the Earth radius. We shall made usage of the characteristic function χ_S, where $\chi_S(s) = 1$ if $s \in S$, and $\chi_S(s) = 0$ otherwise.

IC name	q_0
IC1	$\exp(b_0((X - X_0)^2 + (Y - Y_0)^2 + (Z - Z_0)^2))$
IC2	$0.5 + 0.5(1 + \cos \frac{\pi r}{r_0}) \chi_{\{r < r_0\}}(r)$
IC3	$0.1 + 0.9 \chi_{\{r < r_0\}}(r) \chi_{\{|\lambda - \lambda_0| \leq 0.05 \text{ or } \phi \leq \phi_0\}}(\lambda, \phi)$
IC4	$\exp(b_0[(X - X_1)^2 + (Y - Y_1)^2 + (Z - Z_1)^2]) + \exp(b_0[(X - X_2)^2 + (Y - Y_2)^2 + (Z - Z_2)^2])$

Table 5.2: *Initial conditions considered in the numerical experiments (Figure 5.2).*

VF name	$u_\lambda(\lambda, \phi, t)$	$v_\phi(\lambda, \phi, t)$	$\Delta t^{(0)}$	CFL
VF1	$u_0(\cos(\phi) \cos(\alpha) + \sin(\phi) \cos(\lambda) \sin(\alpha))$	$-u_0 \sin(\lambda) \sin(\alpha)$	3600	0.95
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	0.73
VF3	$-u_0 \sin^2(\frac{\lambda + \pi}{2}) \sin(2\phi) \cos^2(\phi) \cos(\frac{\pi t}{T})$	$\frac{u_0}{2} \sin(\lambda + \pi) \cos^3(\phi) \cos(\frac{\pi t}{T})$	6400	0.91

Table 5.3: *Velocity fields considered in the numerical experiments and their initial time step $\Delta t^{(0)}$ and CFL number.*

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 represent a cosine bell centered at a cube corner, where $r_0 = \frac{R}{3}$, for $\lambda_0 = \frac{\pi}{4}$ and $\phi_0 = \frac{\pi}{2} - \arccos\left(\frac{1}{\sqrt{3}}\right)$. IC3 represent a slotted cylinder, based on Nair and Lauritzen (2010), where $r_0 = \frac{R}{3}$, for $\lambda_0 = \frac{\pi}{4}$ and $\phi_0 = \frac{\pi}{2} - \arccos\left(\frac{1}{\sqrt{3}}\right)$, and therefore the slotted cylinder is also centered at a cube corner. IC4 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: $\alpha = \frac{\pi}{4}$, $\lambda_p = \lambda - \frac{2\pi t}{T}$, $T = 12$ days (12×86400 seconds) and R is the Earth radius. For VF1, VF2 and VF3, we use $u_0 = \frac{2\pi R}{T}$. In this context, VF1 represents the non-divergent rotated zonal field introduced in Williamson et al. (1992). VF2 corresponds to the non-divergent deformational flow described in Nair and Lauritzen (2010), and VF3 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. Furthermore, for VF1, we can compute
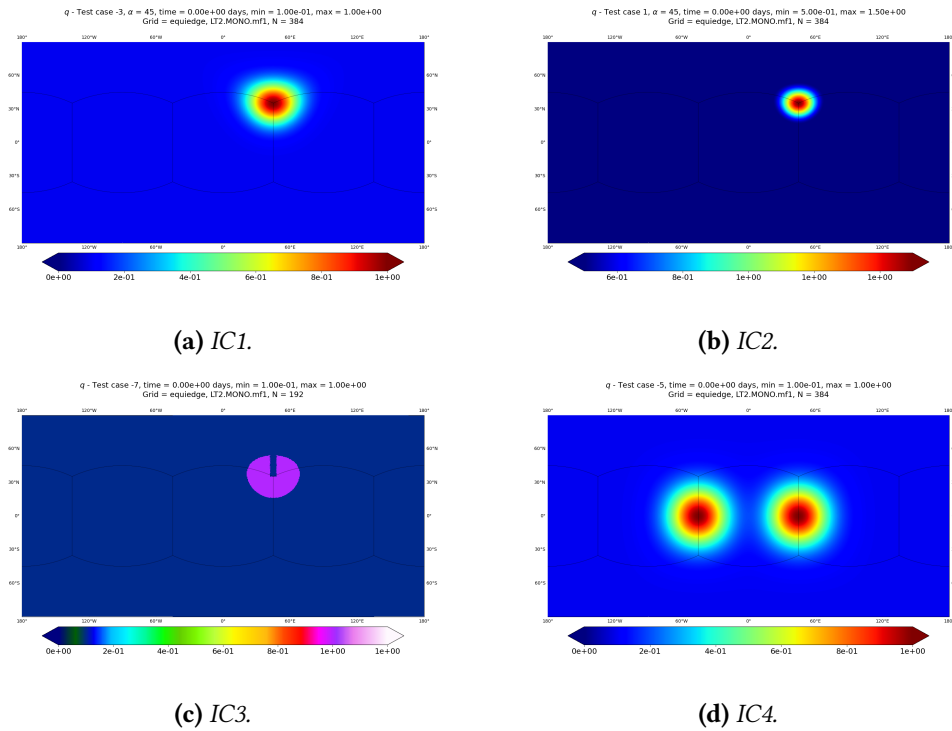
(a) *IC1.*

(b) *IC2.*

(c) *IC3.*

(d) *IC4.*

Figure 5.2: *Illustration of the initial conditions considered in this chapter (Table 5.2).*

the exact solution at any time instant for any initial condition. Therefore, we can analyze the temporal evolution of the error. For an expression of the exact solution when using VF1 and a general initial condition, refer to Brachet (2018, Theorem 5.1, p. 155).

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 the unlimited PPM and the monotonic PPM scheme. 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 using cubic polynomials. 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) using the spherical midpoints formulation (Section 4.2.4) 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 p-norm (Equation (4.50)) and the convergence rate are defined as in Section 2.6.

5.4.1 Advection of one Gaussian hill through the rotated zonal wind

As a first test case, we consider the advection of the Gaussian hill given by IC1 using the rotated zonal wind VF1. In Figure 5.3, we illustrate how the Gaussian hill is advected and passes over 4 cube corners, eventually returning to its initial position.
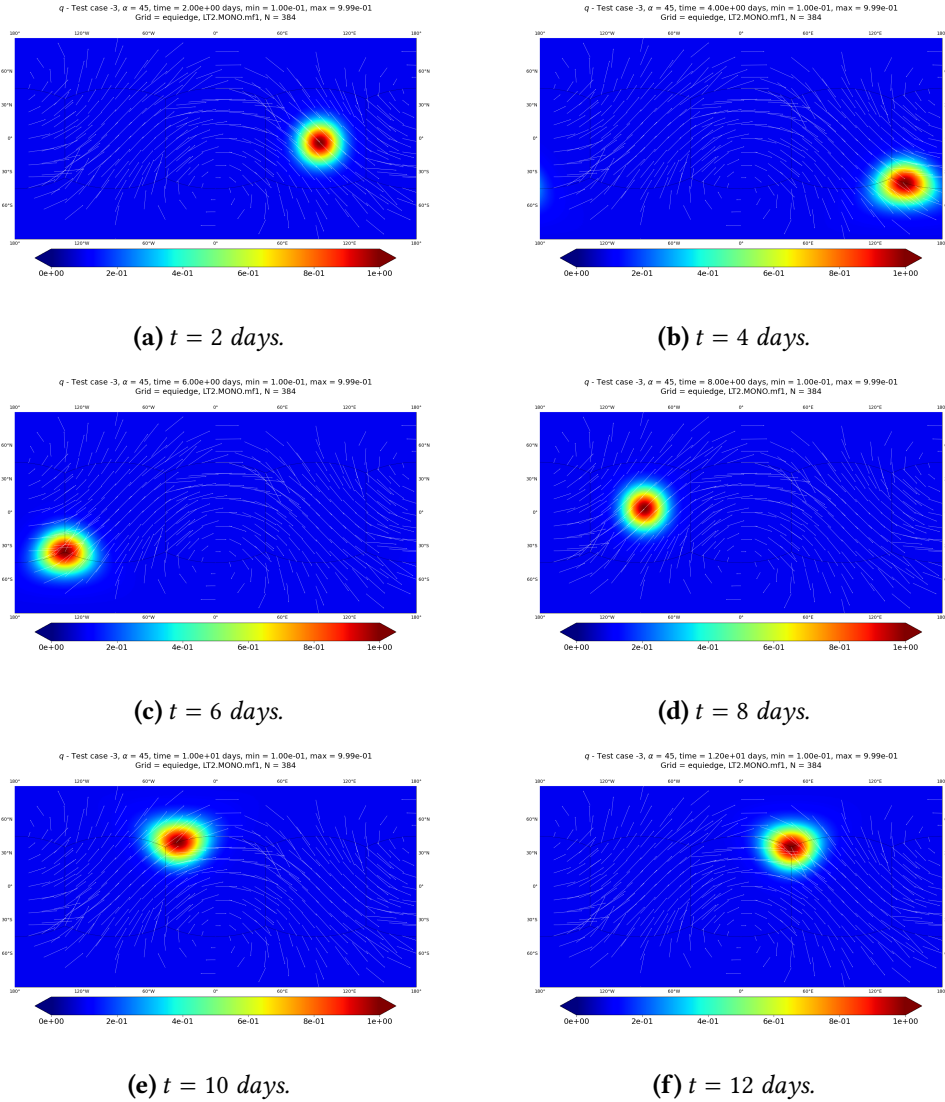

(a) $t = 2$ *days.*

(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 the Gaussian hill at a cube corner (IC1, Table 5.2) and the rotated zonal wind (VF1, Table 5.3). These figures show the advected profile after 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-MONO-mf1 scheme on the equi-edge grid (g0) with $N = 384$.*

The goal of this test is to observe the ability of all schemes and grids to perform this test without creating larger errors or grid-imprinting when the Gaussian hill reaches a corner. In fact, in Figure 5.4 we show how the error evolves with time over 12 days in the L_∞ norm for $N = 384$. Similarly, Figure 5.5 shows the error evolution over time in the L_2 norm. Both figures use green lines to represent the PL scheme and blue lines to represent

the LT scheme. Light colors denote cases where the mass fixer is not used, while dark colors represent cases where it is used. Dashed lines represent the monotonic while solid lines represent the unlimited PPM.

In terms of the L_2 norm, as shown in Figure 5.5, no spikes are observed in the graphs corresponding to the days when the Gaussian passes over a corner. Another conclusion is that the mass fixer does not have too much impact on error evolution when the monotonic scheme is used. However, from Figure 5.4 we can see some small spikes in the L_∞ error on the equi-edge grid (g0) when using the PL and LT schemes, which is less pronounced on the equiangular grid.
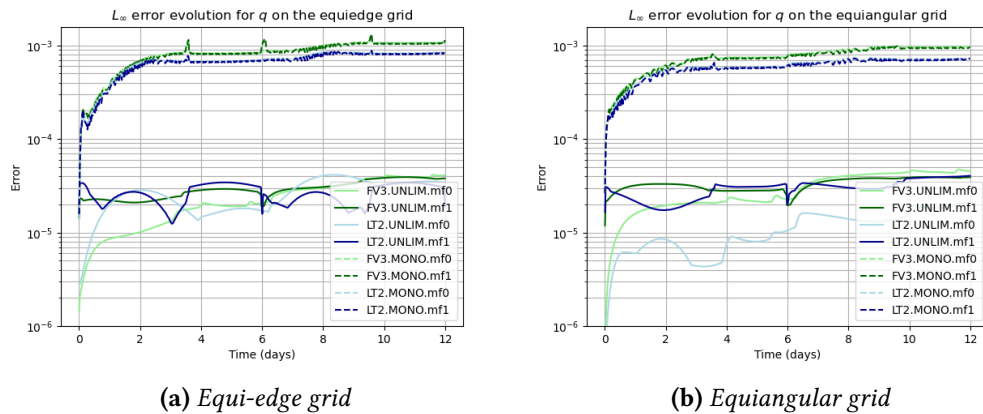

(a) *Equi-edge grid*   **(b)** *Equiangular grid*

Figure 5.4: L_∞ *error evolution for IC1 (Table 5.2) and VF1 (Table 5.3) on the equi-edge grid (a) and on the equiangular grid (b) grids for 12 days and N = 384. Blue lines indicate the use of the LT scheme, while green lines represent the PL scheme. Solid lines represent the results with the unlimited PPM (UNLIM) scheme, whereas dashed lines represent the results with the monotonic (MONO). Light colors show the result without mass fixer (mf0), whereas dark colors show the results with flux averaging (mf1).*
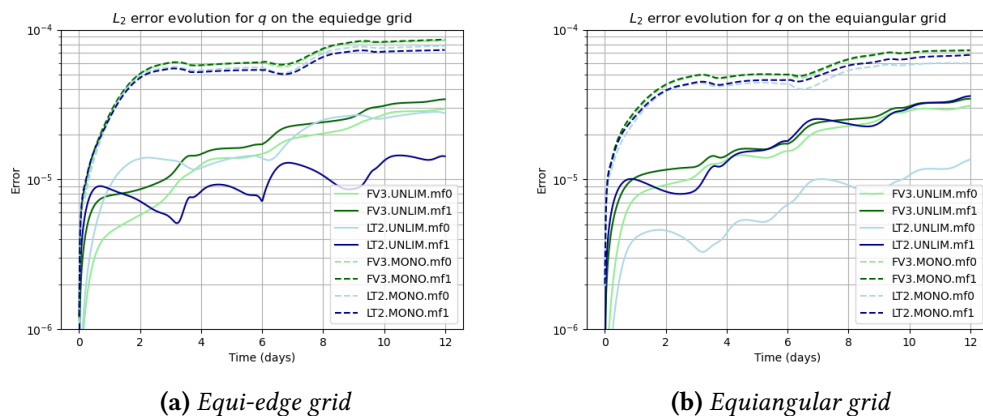

(a) *Equi-edge grid*   **(b)** *Equiangular grid*

Figure 5.5: *As Figure 5.4 but using the L_2 error.*

Indeed, Figure 5.6 shows the final error at a cube corner for the equi-edge grid (g0), and Figure 5.7 shows it for the equiangular grid (g2). The results without a mass fixer are very similar and are not shown here. We can observe that the errors for PL are larger at the corners (Figures 5.6a and 5.7a) than the corner errors of the LT scheme (Figures 5.6b and 5.7b). Additionally, the equi-edge grid (g0) and the equiangular grid (g2) yield similar results for both schemes, with the equi-edge grid resulting in smaller maximum errors.
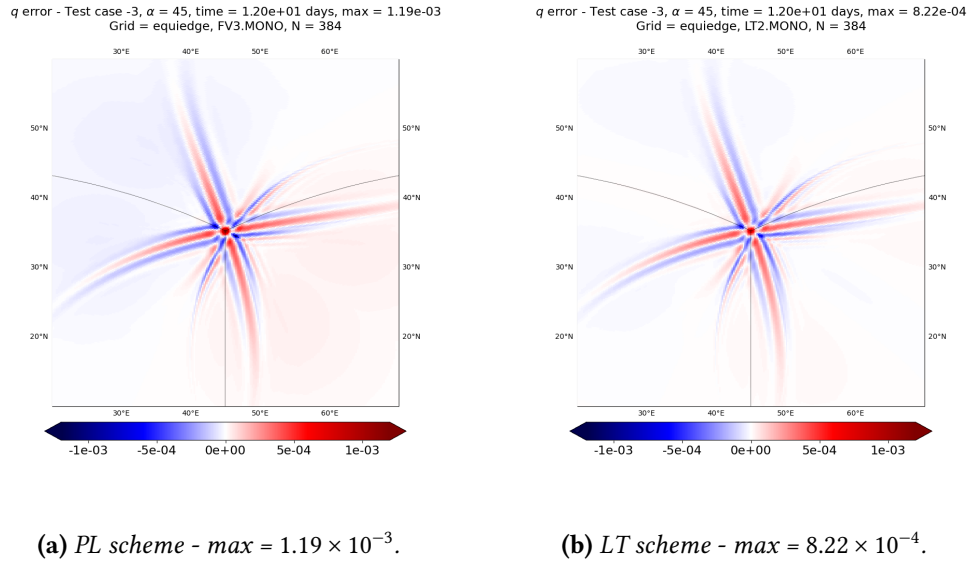

(a) *PL scheme - max = 1.19×10^{-3}*.   (b) *LT scheme - max = 8.22×10^{-4}*.

Figure 5.6: *Advection experiment errors at a cube corner using the Gaussian hill (IC1, Table 5.2) and the rotated zonal wind (VF1, Table 5.3) after 12 days, using the monotonic scheme (MONO) with PL (left) and LT schemes (right) on the equi-edge grid (g0) with N = 384.*
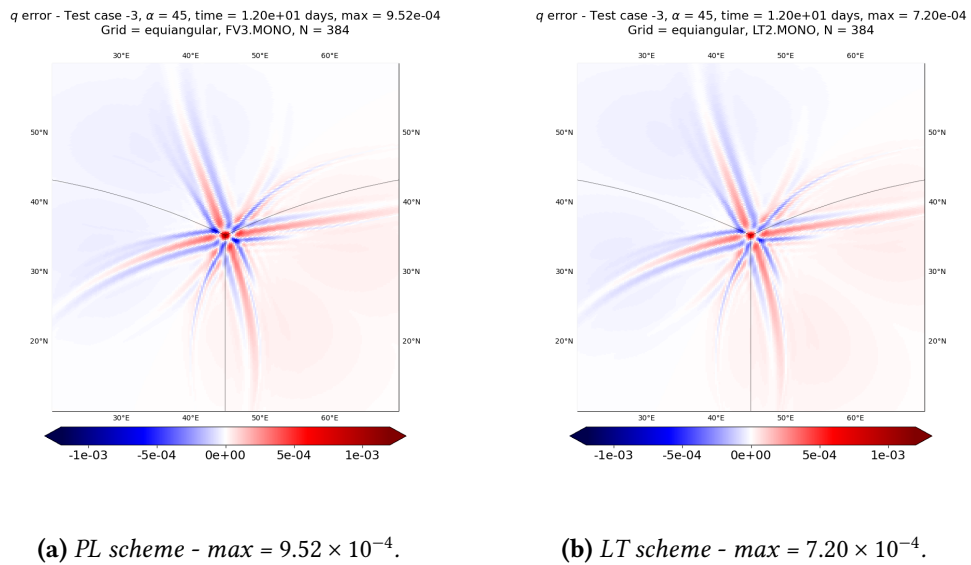

(a) *PL scheme - max = 9.52×10^{-4}*.   (b) *LT scheme - max = 7.20×10^{-4}*.

Figure 5.7: *As Figure 5.6 but using the equiangular grid (g2).*

Finally, in Figures 5.8 and 5.9 we show the error convergence in L_∞ and L_2 norms. We can observe that all schemes with the unlimited PPM (UNLIM) achieve second-order accuracy as expected. However, for MONO, the order is reduced, which is also expected. Additionally, we can see that MONO with LT has smaller errors when comparing the blue dashed lines with the green dashed lines, for both L_∞ and L_2 norms on both equi-edge grid (g0) and the equiangular grid (g2), indicating that LT is slightly more accurate. In general, the errors of g0 are slightly smaller than those of g2.
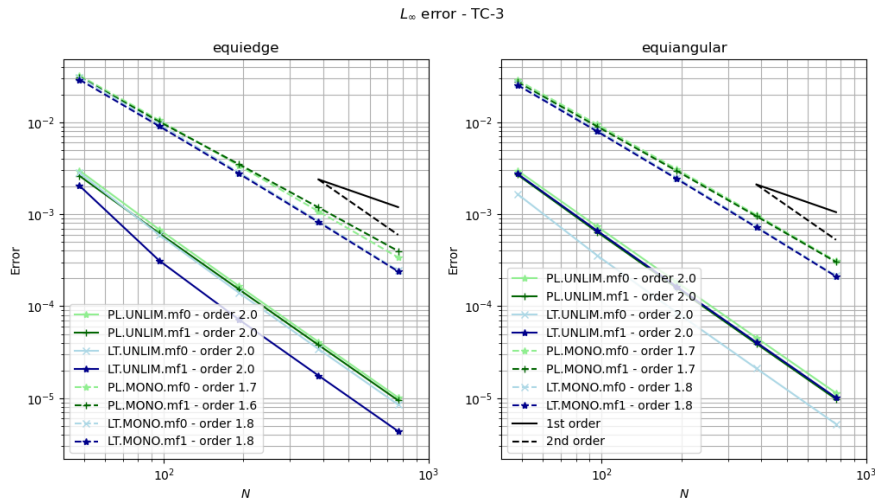

Figure 5.8: L_∞ *error convergence for the advection on the sphere test using the Gaussian hill at cube a corner (IC1, Table 5.2) and the rotated zonal wind (VF1, Table 5.3) on the equi-edge grid (g0, left) and on the equiangular grid (g2, right) after 12 days. Blue lines indicate the use of the LT scheme, while green lines represent the PL scheme. Solid lines represent the results with the unlimited PPM (UNLIM) scheme, whereas dashed lines represent the results with the monotonic PPM (MONO). Light colors show the result without mass fixer (mf0), whereas dark colors show the results with flux averaging (mf1).*
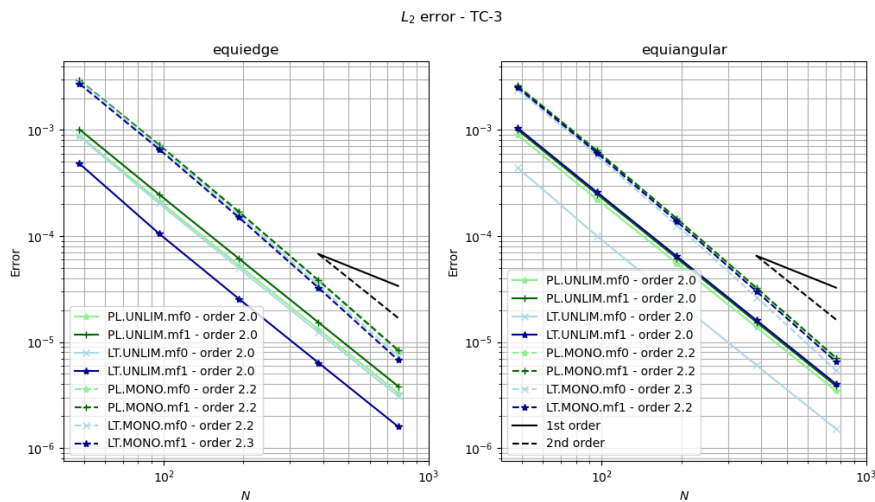

Figure 5.9: *As Figure 5.8 but considering the L_2 norm.*

5.4.2 Advection of a cosine bell hill through the rotated zonal wind

As the second test case, we consider the advection of the cosine bell given by IC2 using the rotated zonal wind VF1. The cosine bell is advected and passes over 4 cube corners, very similarly to the Gaussian hill, as shown in Figure 5.3. The major difference between IC1 and IC2 is that IC1 is a smooth function while IC2 is only continuous. Then, we may compare how both schemes handle a non-differentiable function. For $N = 384$, the temporal evolution of the errors is very similar to that of IC1 (Figures 5.4 and 5.5) and is omitted here.
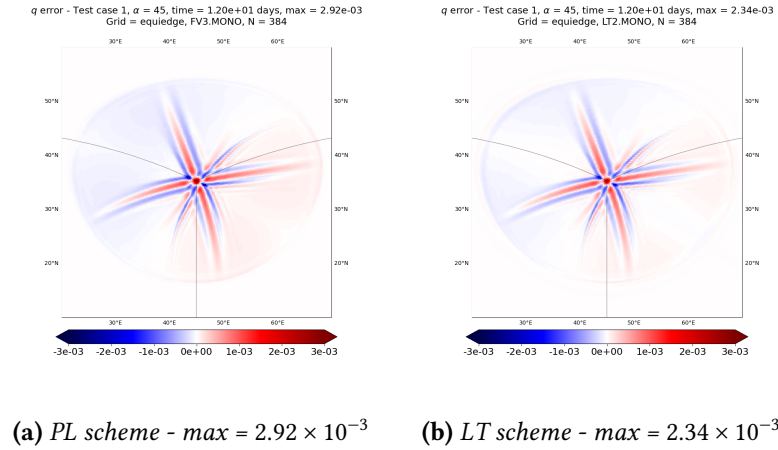

(a) *PL scheme - max = 2.92×10^{-3}*   **(b)** *LT scheme - max = 2.34×10^{-3}*

Figure 5.10: *Advection experiment errors at a cube corner using the cosine bell (IC2, Table 5.2) and the rotated zonal wind (VF1, Table 5.3) after 12 days, using the monotonic scheme (MONO) with PL (left) and LT schemes (right) on the equi-edge grid (g0) grid with N = 384.*

(a) *PL scheme - max = 2.35×10^{-3}*   **(b)** *LT scheme - max = 1.91×10^{-3}*

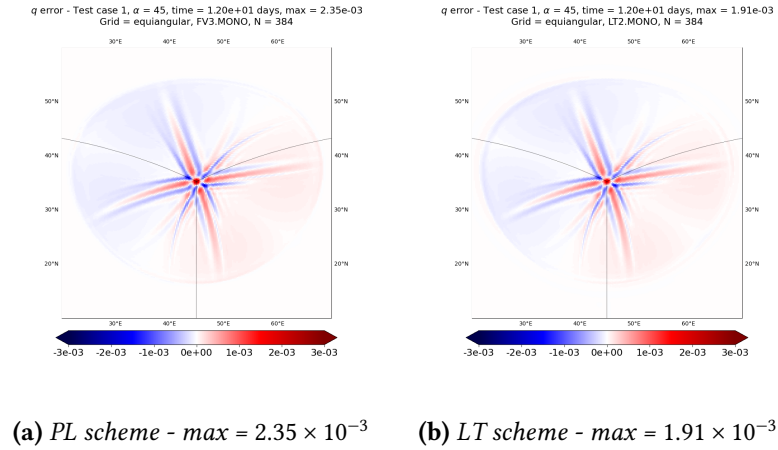**Figure 5.11:** *As Figure 5.6 but using the equiangular grid (g2).*

In Figure 5.10, we show the final error at the cube corner for the equi-edge grid (g0), and Figure 5.11 show the same for the equiangular grid (g2). We can observe that the results are similar to IC1 with VF1 shown in Figures 5.6 and 5.7. We conclude again that LT has a smaller error at the corner, especially for the equi-edge grid (g0).

In Figures 5.12 and 5.13, we show the error convergence for the L_∞ and L_2 norms, respectively. Note that the errors in the L_∞ norm (Figure 5.12) for the unlimited PPM (solid lines) are not achieving second order as they should because the solution is not differentiable. For MONO (dashed lines), we can see that the L_∞ errors for LT (blue lines) are smaller than the errors of PL (green lines), especially for the equi-edge grid (g0). Finally, the L_2 errors are very similar (Figure 5.13).
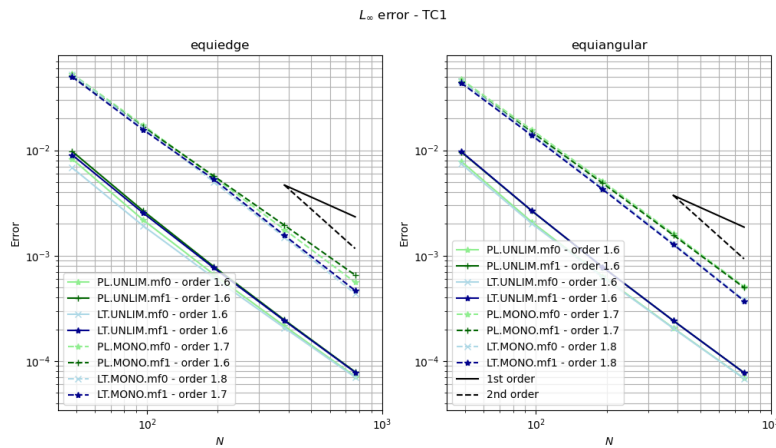
Figure 5.12: L_∞ *error convergence for the advection on the sphere test using the cosine bell at cube a corner (IC2, Table 5.2) and the rotated zonal wind (VF1, Table 5.3) on the equi-edge grid (g0, left) and on the equiangular grid (g2, right) after 12 days. Blue lines indicate the use of the LT scheme, while green lines represent the PL scheme. Solid lines represent the results with the unlimited PPM (UNLIM) scheme, whereas dashed lines represent the results with monotonic (MONO). Light colors show the result without mass fixer (mf0), whereas dark colors show the results with flux averaging (mf1).*

Figure 5.13: *As Figure 5.19 but considering the L_2 norm.*

5.4.3 Advection of a slotted cylinder through the rotated zonal wind

The third test case here is the slotted cylinder advection, given by IC3 from Table 5.2 and using again the rotated zonal wind VF1 (Table 5.3). We show how the solution evolves with time in Figure 5.14.

(a) $t = 2$ days.

(b) $t = 4$ days.

(c) $t = 6$ days.
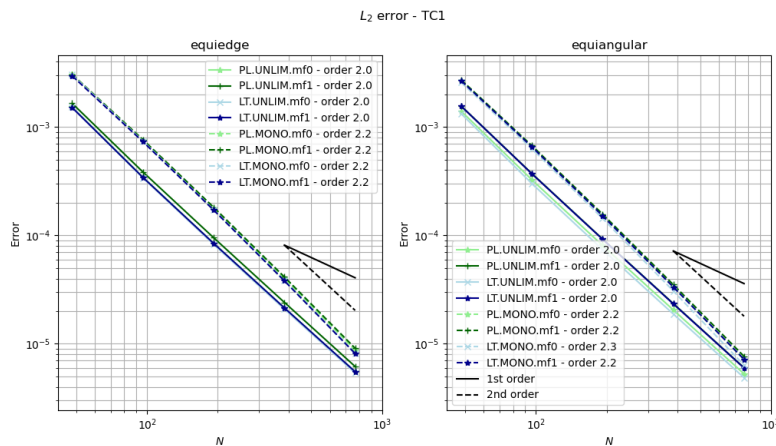
(d) $t = 8$ days.

(e) $t = 10$ days.

(f) $t = 12$ days.
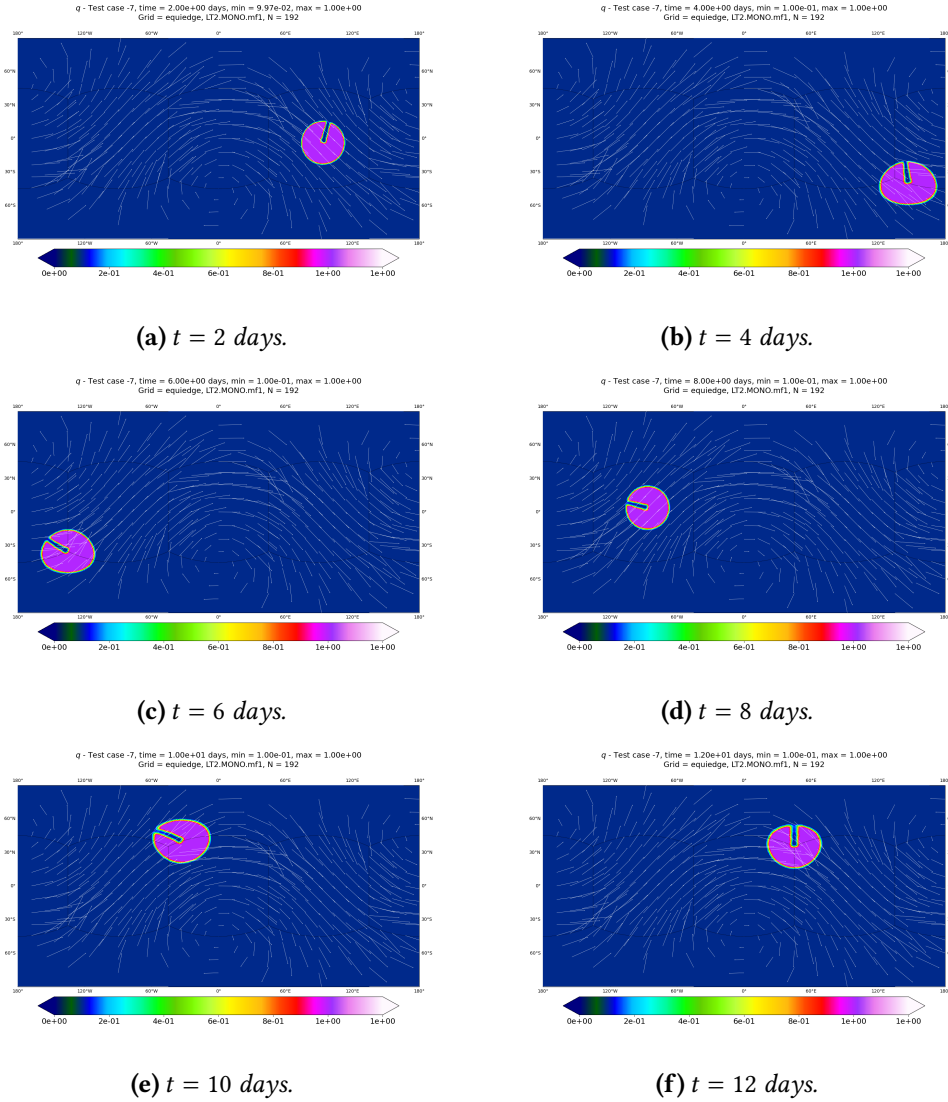
Figure 5.14: *Advection experiment results using the slotted cylinder at a cube corner (IC3, Table 5.2) and the rotated zonal wind (VF1, Table 5.3). These figures show the advected profile after 2 (5.14a), 4 (5.14b), 6 (5.14c), 8 (5.14d), 10 (5.14e), and 12 (5.14f) days. We are using the LT-MONO-mf1 scheme on the equi-edge grid (g2) with N = 192.*

The goal of this test is to assess the qualitative behavior of the solution, especially to see if the limiter prevents oscillations that are expected since the slotted cylinder has a discontinuous profile. Also, as the slotted cylinder is located at a cube corner, we would like to see if the corner affects the solution. In Figure 5.15, we present the final solutions for $N = 192$ as well the reference solution. It is evident that all the schemes yield similar

results, and we cannot observe any interference of the corner despite using a discontinuous
initial condition.

(a) *PL-MONO at g0.*

(b) *LT-MONO at g0.*

(c) *PL-MONO at g2.*

(d) *LT-MONO at g2.*

(e) *IC and exact solution at day
12.*

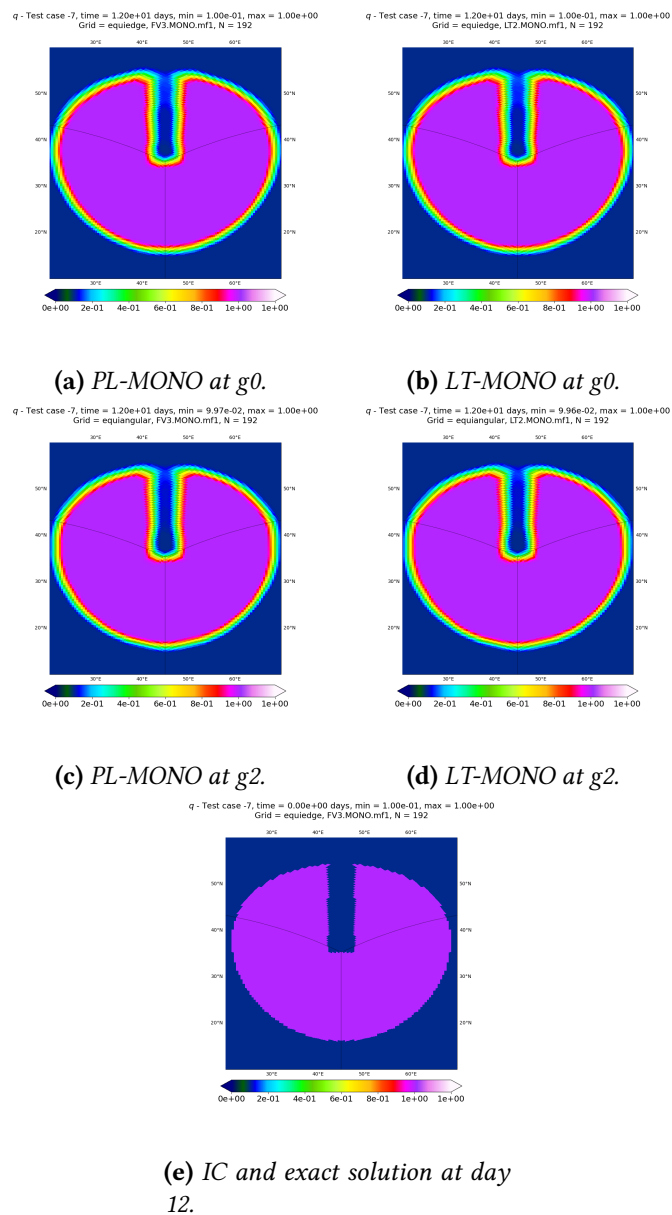**Figure 5.15:** *Slotted cylinder at corner test with N = 192 after 12 days for the schemes PL-MONO at
the equi-edge grid (g0) (a), LT-MONO at the equi-edge grid (g0) (b), PL-MONO at the equiangular
grid (g2) (c) and LT-MONO at the equiangular grid (g2) (d). (e) depicts the reference solution. The
monotonic scheme is denoted by MONO.*

5.4.4 Non-divergent deformational flow

The fourth test case considers the divergence free wind VF2 from Table 5.3, along with the initial condition IC4 from Table 5.2, where the velocity is time-dependent. This test is suggested by Nair and Lauritzen (2010), and Figure 5.16 shows how the solution evolves over time. Since the wind is divergence free, we observe that it deforms the two Gaussian hills, without creating new extrema. Eventually, the final solution is equal to the initial condition after 12 days. This test is the spherical analogous of the planar divergence free deformational flow test presented in Section 3.4.2.

(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 5.16: *Advection experiment results using the two Gaussian hills (IC4, Table 5.2) and the variable in time divergent free wind (VF2, Table 5.3). These figures show the advected profile after 2 (5.16a), 4 (5.16b), 6 (5.16c), 8 (5.16d), 10 (5.16e), and 12 (5.16f) days. We are using the LT-MONO-mf1 scheme on the equi-edge grid (g0) with N = 384.*

Figures 5.17 and 5.18 show the final error at a cube face for the equi-edge grid (g0) and the equiangular grid (g2), respectively. The results without a mass fixer are very similar and are not shown here. We can observe that the errors for both PL and LT are very similar, and also the type of grid does not have a significant impact.

(a) *PL scheme.*   **(b)** *LT scheme.*
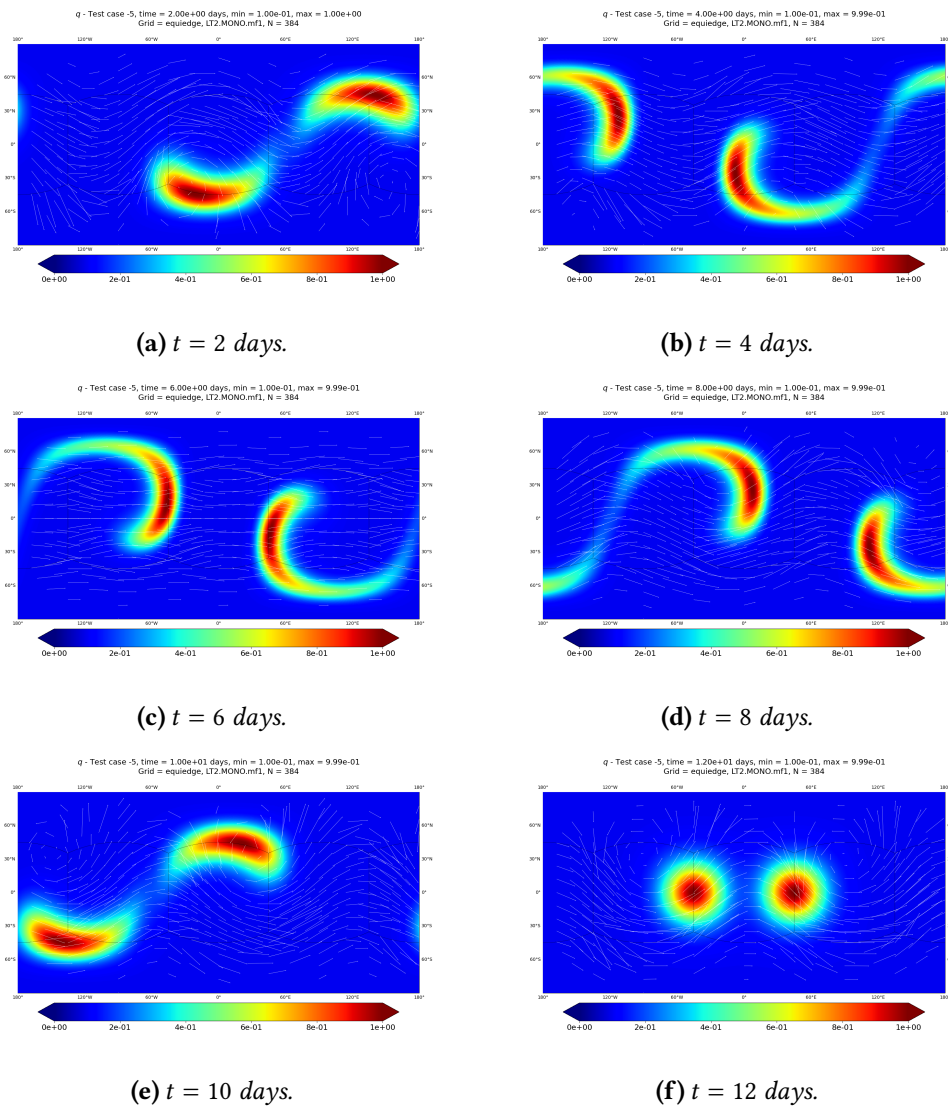
Figure 5.17: *Advection experiment results using the two Gaussian hills (IC4, Table 5.2) and the variable in time divergence free wind (VF2, Table 5.3). These figures show the advected profile after 2 (5.16a), 4 (5.16b), 6 (5.16c), 8 (5.16d), 10 (5.16e), and 12 (5.16f) days. We are using the LT-MONO-mf1 scheme on the equi-edge grid (g0) with N = 384.*

(a) *PL scheme.*   **(b)** *LT scheme.*
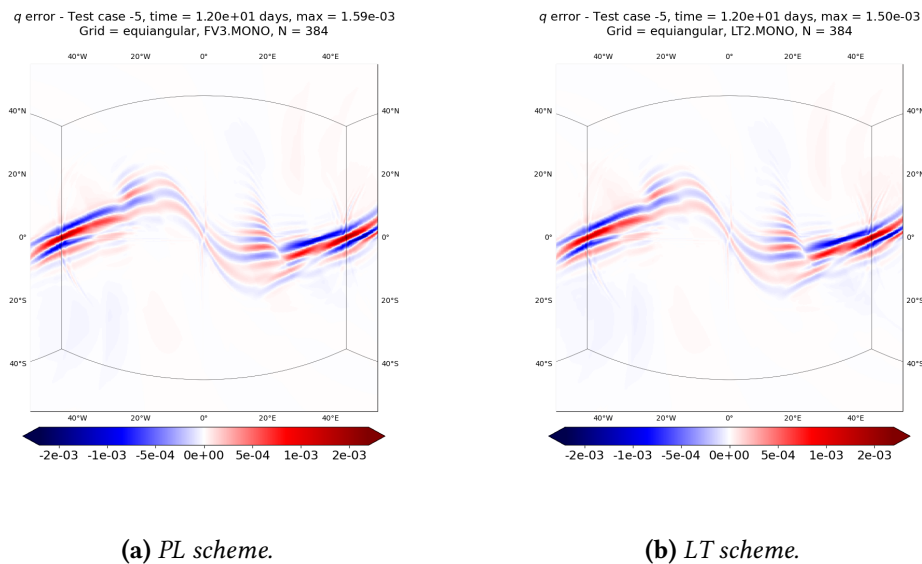
Figure 5.18: *As Figure 5.17 but using the equiangular grid (g2).*

Figures 5.19 and 5.20 we show the error convergence in L_∞ and L_2 norms. Once more, it is evident that all schemes with the unlimited PPM (UNLIM) achieve second-order accuracy as expected, while those with the monotonic PPM (MONO) experience a reduced order in L_∞ norm. In L_2 norm, the order is 2 for MONO. Furthermore, LT and PL demonstrate

almost the same errors when utilizing MONO, with the LT scheme being slightly smaller. We also notice that the mass fixer does not impact the errors.

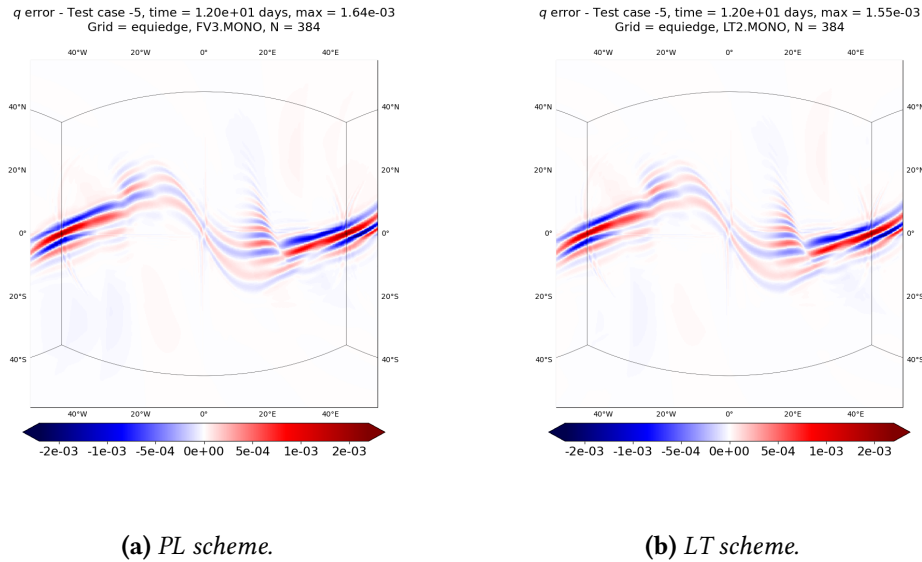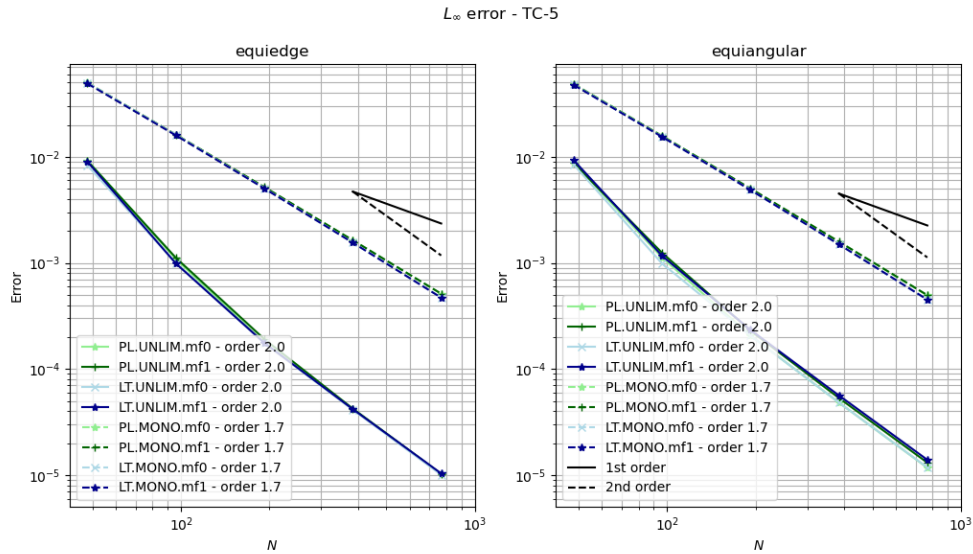**Figure 5.19:** L_∞ *error convergence for the advection on the sphere test using the two Gaussian hills (IC4, Table 5.2) and the variable in time divergent-free wind (VF2, Table 5.3) on the equi-edge grid (g0, left) and on the equiangular grid (g2, right) after 12 days. Blue lines indicate the use of the LT scheme, while green lines represent the PL scheme. Solid lines represent the results with the unlimited PPM (UNLIM) scheme, whereas dashed lines represent the results with the monotonic PPM (MONO). Light colors show the result without mass fixer (mf0), whereas dark colors show the results with flux averaging (mf1).*
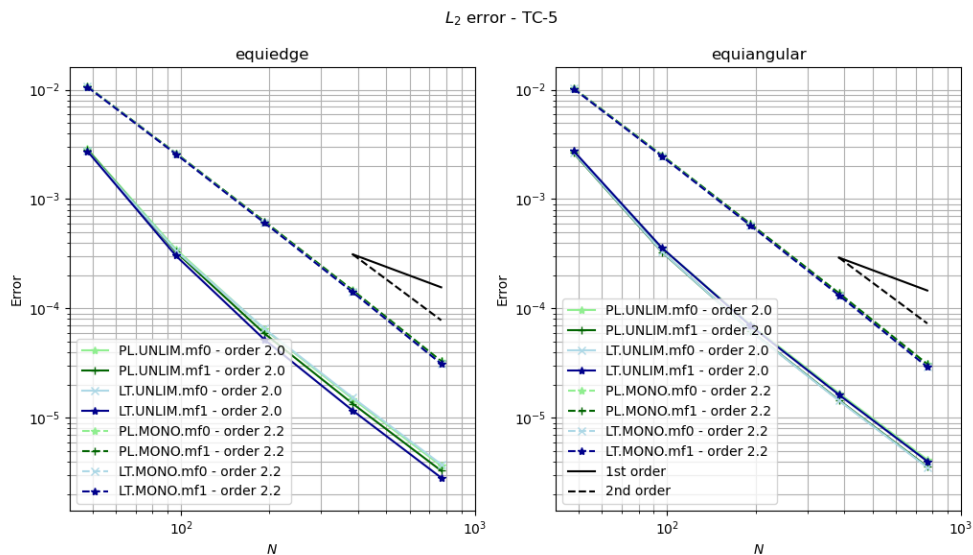
Figure 5.20: *As Figure 5.19 but considering the L_2 norm.*

5.4.5 Divergent deformational flow

The fifth and last test case considers the divergent wind VF3 from Table 5.3, along with the initial condition IC4 from Table 5.2, where the velocity is time-dependent. This test is also suggested by Nair and Lauritzen (2010), and Figure 5.21 shows how the solution evolves over time. Since the wind is divergent, we observe that it deforms the two Gaussian hills, creating new extrema. Eventually, the final solution is equal to the initial condition after 12 days. This test is the spherical analogous of the planar divergent deformational flow test presented in Section 3.4.3.

(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 5.21: *Advection experiment results using the two Gaussian hills (IC4, Table 5.2) and the divergent wind (VF3, Table 5.3). These figures show the advected profile after 2 (5.21a), 4 (5.21b), 6 (5.21c), 8 (5.21d), 10 (5.21e), and 12 (5.21f) days. We are using the LT-MONO-mf1 scheme on the equiangular grid (g2) with N = 384.*

Figures 5.22 and 5.23 show the final error at a cube face for the equi-edge grid (g0) and the equiangular grid (g2), respectively. The results without a mass fixer are very similar and

are not shown here. We can observe that the errors for PL are much larger, with significant errors present in many cells, whereas LT has smaller errors that are concentrated in some ripples.
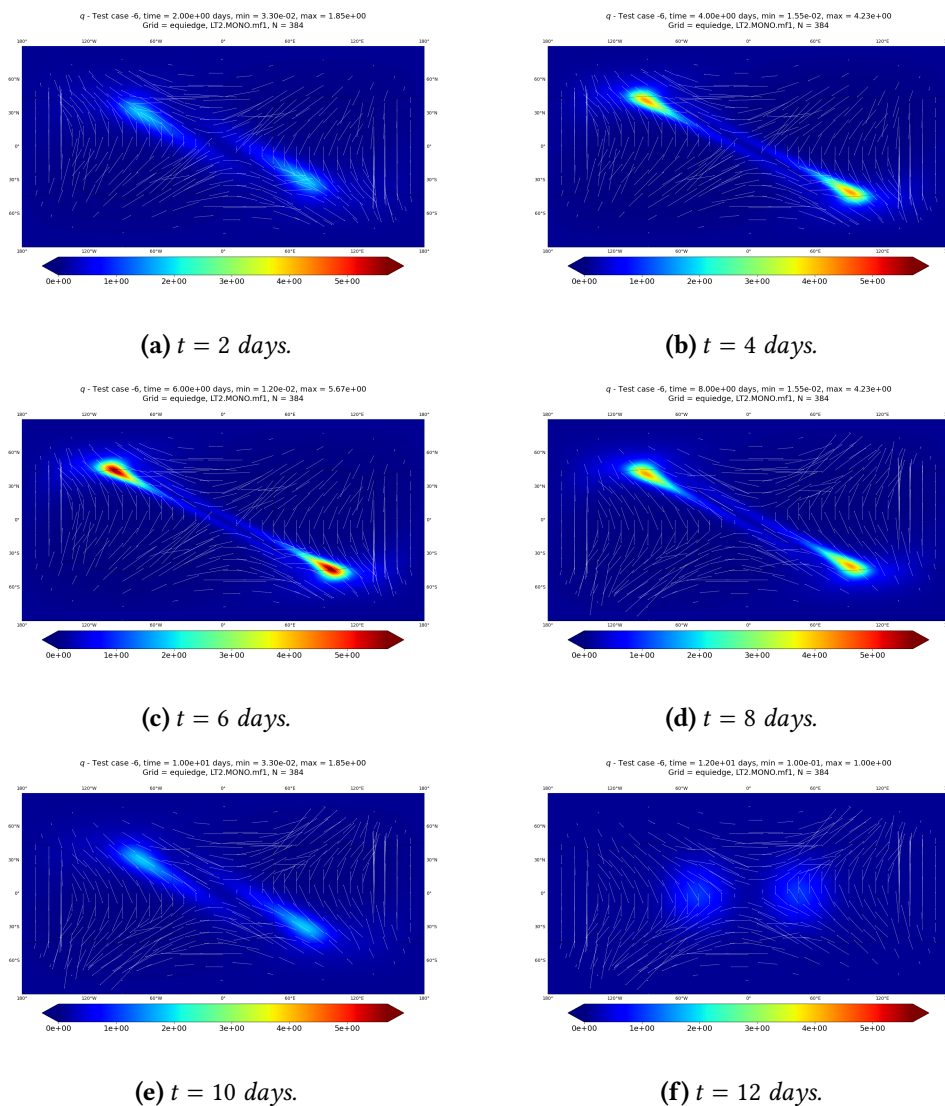

(a) *PL scheme.*
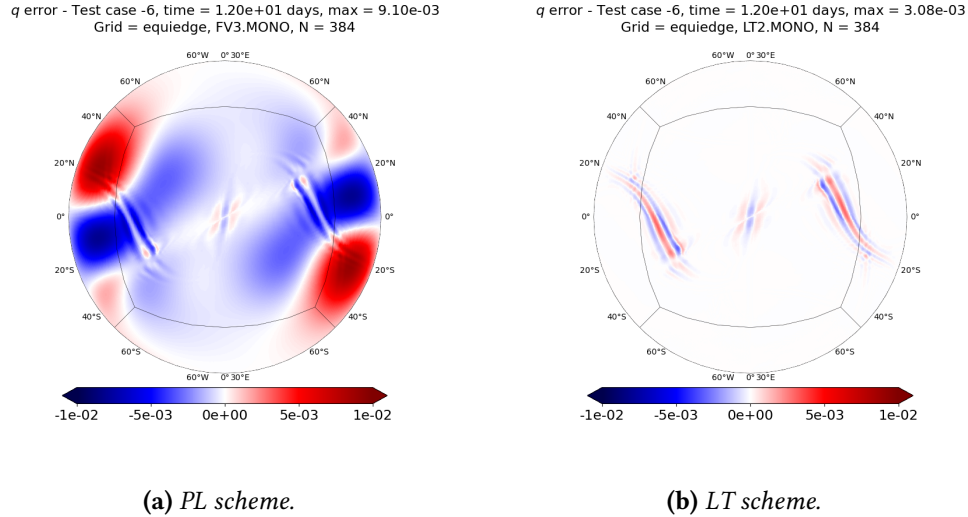
(b) *LT scheme.*

Figure 5.22: *Advection experiment errors using the two Gaussian hills (IC4, Table 5.2) and the divergent wind (VF3, Table 5.3) after 12 days, using the monotonic scheme (MONO) with PL (left) and LT schemes (right) on the equi-edge grid (g0) with N = 384.*

(a) *PL scheme.*

(b) *LT scheme.*
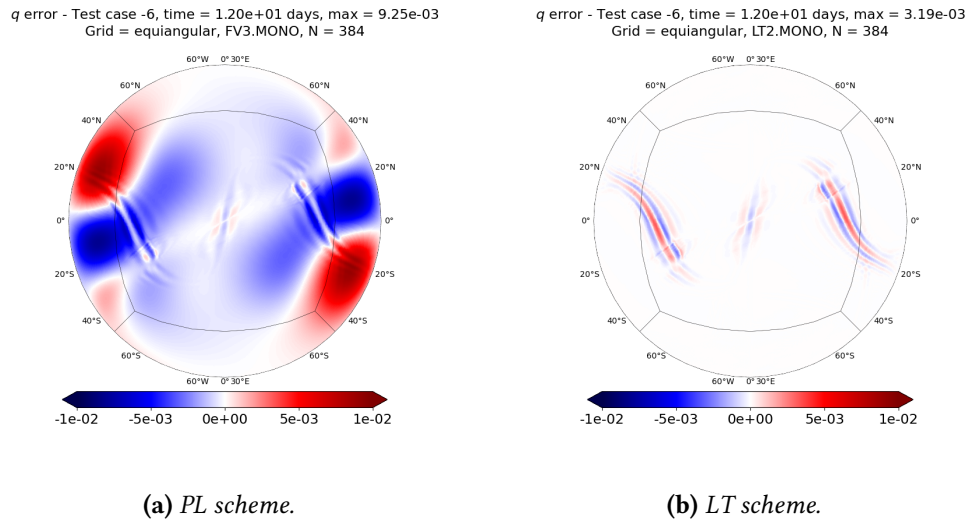
Figure 5.23: *As Figure 5.22 but using the equiangular grid (g2).*

Figures 5.24 and 5.25 we show the error convergence in L_∞ and L_2 norms. These figures highlight a major significant distinction between LT and PL schemes, unlike the previous tests. It is clear that PL with the unlimited PPM achieves only first-order accuracy, whereas LT with the unlimited PPM (UNLIM) achieves third-order accuracy, surpassing second-order the expectation, for both equi-edge grid (g0) and the equiangular grid (g2) and norms. For the monotonic scheme (MONO), LT demonstrates second-order accuracy in the L_2 norm, while PL is only first-order. LT with the monotonic scheme (MONO) exhibits smaller errors in the L_∞ norm compared to the PL scheme for all grids. This discrepancy arises because the PL splitting is designed for divergence-free flows, while LT is designed to be

second-order regardless of the flow characteristics. Finally, these results are similar to the planar divergent deformational flow test presented in Section 3.4.3.
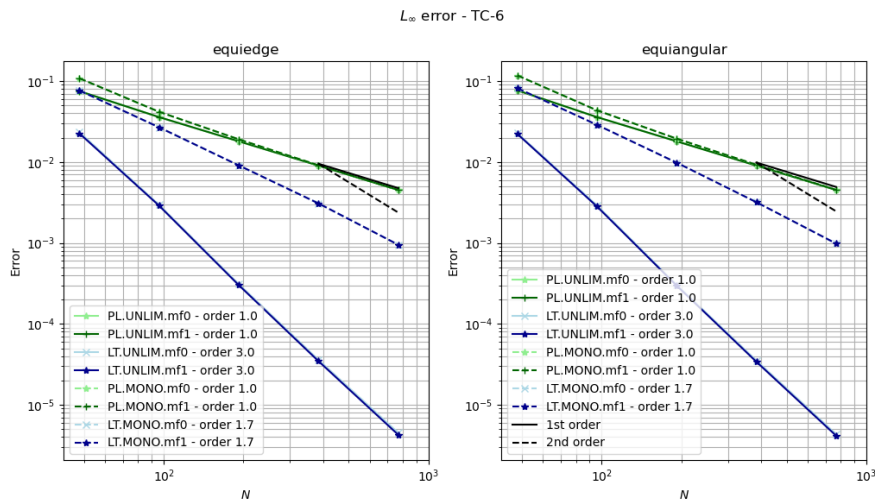
Figure 5.24: L_∞ *error convergence for the advection on the sphere test using the two Gaussian hills (IC4, Table 5.2) and the divergent wind (VF3, Table 5.3) on the equi-edge grid (g0, left) and on the equiangular grid (g2, right) after 12 days. Blue lines indicate the use of the LT scheme, while green lines represent the PL scheme. Solid lines represent the results with the unlimited PPM (UNLIM) scheme, whereas dashed lines represent the results with the monotonic PPM (MONO). Light colors show the result without mass fixer (mf0), whereas dark colors show the results with flux averaging (mf1).*
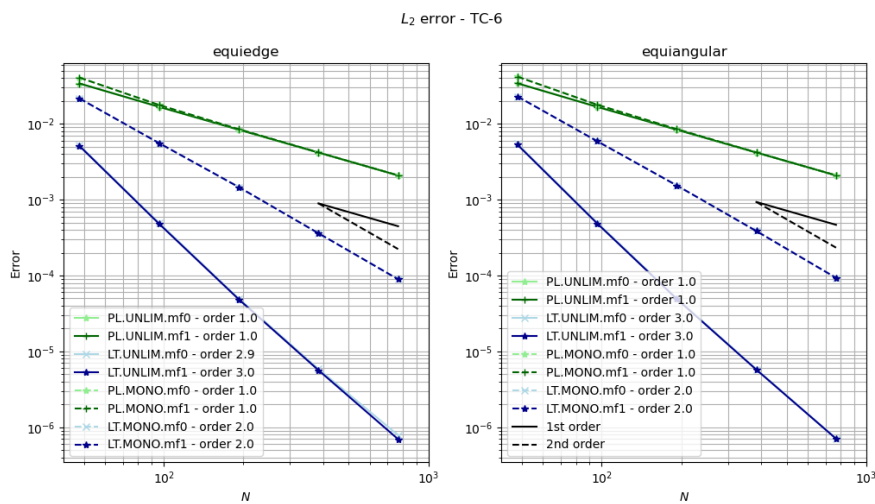
Figure 5.25: *As Figure 5.24 but considering the L_2 norm.*

5.5   Concluding remarks

In summary, in this Chapter, we demonstrate how the dimension-splitting methods from Chapter 3, namely the PL and LT methods, can be extended to the cubed-sphere to solve the advection equation on the sphere using the cubed-sphere grids equi-edge (g0) and equiangular (g2), along with the duo-grid interpolation presented in Chapter 4. We observed a major difference in the metric term that appears in this case, and it may be treated differently in the PPM flux computation. Also, on the cubed-sphere, we need to apply a mass fixer, namely averaging the fluxes at the cube edges, to ensure exact mass preservation.

We showed that LT may use a more accurate metric term formulation, since this scheme is more flexible and does not need to eliminate the splitting error for a constant scalar wind and divergence-free wind, which is demanded for the PL scheme. This difference in requirements allows the LT scheme to utilize a more accurate metric term formulation compared to PL.

The conclusions of this Chapter are essentially extensions of the results from Chapter 3 from the plane to the cubed-sphere. In fact, the LT scheme, which utilize a second-order departure point calculation, showed to have smaller errors than the PL scheme, which is designed to preserve a constant scalar field for divergence-free winds. Both schemes are second-order when no limiter is employed and the wind is divergence-free. The major difference between LT and PL is when the wind is not divergence-free. In this case, PL is only first order, while LT is second-order. Even with a limiter, LT is much more accurate than PL in this case. This was demonstrated consistently throughout the simulations. Therefore, our major conclusion here is that the LT scheme is much more accurate regardless of whether the wind is divergence-free or not, while PL is only accurate for divergence-free winds.

Additionally, the Gaussian hill and cosine bell advection through a rotated zonal wind showed that some errors of PL and LT presented small spikes whenever the Gaussian hill passed over a corner. The overall results for this test showed that the LT scheme have slightly smaller errors. We could also observe that the mass fixer did not significantly impact the results, and the equi-edge grid (g0) grid generally exhibited smaller errors compared to the equiangular grid (g2), with LT showing good performance in both grids.

Chapter 6

Cubed-sphere finite-volume shallow-water model

Now that we have described how to solve the advection equation on the cubed-sphere in Chapter 5, we are able to introduce the method of Lin and Rood (1997) for solving the shallow-water equations (SWE) on the cubed-sphere. In fact, this scheme considers the SWE in the vector invariant form, and therefore, the flux operators discussed in Chapter 5 are used to update the fluid depth, as well as the time-averaged relative vorticity and kinetic energy fluxes. This scheme first solves the SWE for a half time-step to obtain C-grid covariant winds and then utilizes this new information to advance the D-grid covariant winds for a full time step. The C-grid half-step employs upwind flux operators, which are computationally inexpensive, while the D-grid uses PPM-based fluxes, providing higher accuracy. We note that other flux operators could be employed here, but the choice presented is what is utilized in FV3.

Although the advection equation on the sphere plays a crucial role in the development of dynamical cores by modeling the advection of scalar fields on the sphere, it does not capture important features present in the SWE on the sphere, such as the Coriolis effect, inertia-gravity waves, geostrophic adjustment, Rossby waves, among others. Therefore, SWE serve as an excellent benchmark for assessing dynamical cores in general, as they are only two-dimensional but represent a complex geophysical model for atmosphere dynamics. Furthermore, the 3D non-hydrostatic solver of FV3 utilizes a vertical Lagrangian coordinate system, requiring the solution of the shallow-water equations on the Lagrangian surfaces (Harris et al., 2021; Lin, 2004).

The goal of this Chapter is to provide a detailed description of the SWE solver from Lin and Rood (1997). Since this scheme uses advection operators to update the variables, we are going to incorporate the new advection scheme LT introduced in Chapter 5 and compare it with the PL advection scheme from Putman and Lin (2007), which is currently employed in FV3. Thus, we will extend the comparisons made in Chapter 5 to the context of the SWE.

This Chapter is outlined as follows: In Section 6.1, we introduce the SWE and some of its properties, and then we discuss the C-grid and D-grid discretization proposed by

Lin and Rood (1997) in Section 6.2. Our modifications for their scheme are presented in Section 6.3. Following that, in Section 6.4, we present numerical results using classical tests from the literature. Finally, in Section 6.5, we provide concluding remarks.

6.1 The shallow-water equations on the sphere

In this Section, we introduce the shallow-water equations (SWE) on the sphere using a cubed-sphere mapping (equi-edge or equiangular) as discussed in Section 4.2. All the notation from Sections 4.1 and 4.2 is utilized here. For simplicity, we omit the dependence on p since it does not affect the description across cube faces. Additionally, we assume that the ghost cells are filled using the duo-grid interpolation scheme outlined in Sections 4.3.2 and 4.3.3. The shallow-water equations (SWE) are a set of hyperbolic partial differential equations describing how the fluid depth, denoted by h, and the wind $\boldsymbol{u}$ evolve with time. Since the cubed-sphere system is non-orthogonal, the SWEs will feature the covariant winds $\mathfrak{U}, \mathfrak{V}$ and contravariant winds $\mathfrak{u}, \mathfrak{v}$, as discussed in Section 4.2.6. The SWE on the a cubed-sphere panel are expressed as in its vector invariant form as (Nair et al., 2005a; Rančić et al., 1996):

$$\partial_t(\sqrt{\mathfrak{g}}h)(x, y, t) = -[\partial_x(\mathfrak{u}\sqrt{\mathfrak{g}}h + \partial_y(\mathfrak{v}\sqrt{\mathfrak{g}}h)](x, y, t), \tag{6.1}$$

$$\partial_t\mathfrak{U}(x, y, t) = -[\partial_x K - \mathfrak{v}\sqrt{\mathfrak{g}}\xi + \partial_x\Phi](x, y, t), \tag{6.2}$$

$$\partial_t\mathfrak{V}(x, y, t) = -[\partial_y K + \mathfrak{u}\sqrt{\mathfrak{g}}\xi + \partial_y\Phi](x, y, t), \tag{6.3}$$

$\Phi = g(h + b)$ is the geopotential, g is the gravity, b is the bottom topography,

$$K = \frac{\mathfrak{u}\mathfrak{U} + \mathfrak{v}\mathfrak{V}}{2}, \tag{6.4}$$

is the kinetic energy,

$$\xi = f + \zeta, \tag{6.5}$$

is the absolute vorticity, where

$$f = 2\Omega \sin\phi, \tag{6.6}$$

is the Coriolis parameter, ϕ is the latitude, $\Omega = 7.2921 \times 10^{-5}$ is the Earth rotation speed, and

$$\zeta = \frac{1}{\sqrt{\mathfrak{g}}}(\partial_x\mathfrak{V} - \partial_y\mathfrak{U}), \tag{6.7}$$

is the relative vorticity.

Now, let us describe some elementary properties of the SWE. By taking ∂_y in Equation (6.2) and ∂_x in Equation (6.3) and subtracting the obtained results, we get that the absolute vorticity satisfies:

$$\partial_t(\sqrt{\mathfrak{g}}\xi)(x, y, t) = -[\partial_x(\mathfrak{u}\sqrt{\mathfrak{g}}\xi) + \partial_y(\mathfrak{v}\sqrt{\mathfrak{g}}\xi)](x, y, t). \tag{6.8}$$

One can also easily show (as described in Section 5.1) that the total mass of h and ζ is preserved.

By replacing Equations (4.40) and (4.46) in Equation (6.4), it follows that the kinetic

energy may be rewritten in terms of the normalized contravariant and covariant wind components as:

$$K = \frac{uU + vV}{2}.$$ (6.9)

Recall that the normalized contravariant and covariant wind components are given by Equation (4.40) and Equation (4.45), respectively.

The total energy is defined as:

$$E = g\frac{h^2}{2} + gb + hK.$$ (6.10)

One can deduce an equation for the time evolution of the total energy (see, for example, Ringler et al. (2010)) and observe that its integral over the sphere is preserved; that is, the total energy is conserved.

6.1.1 Momentum equation discretization

The continuity equation (6.1) has the exact same form as the advection equation when written in its conservative form. Therefore, this equation can be solved on the A-grid using C-grid contravariant winds, as explored in Chapters 3 and 5. This is how the continuity equation is solved in FV3. As we shall see later, this equation is solved twice: once using the 2D upwind flux and another using the dimension-splitting method from Chapter 5 with PPM.

Therefore, we need to describe how we can solve the momentum equations (6.2) and (6.3). The method of Lin and Rood (1997) employs two types of approaches on their shallow-water solver: one using a C-grid wind and the other using a D-grid wind. The C-grid method serves as an intermediate step utilized by the D-grid method. Our goal now is to describe a general discretization of the momentum equations for the C-grid and D-grid winds. The full description of the C-grid and D-grid solvers proposed by Lin and Rood (1997) will be provided in Sections 6.2.1 and 6.2.2, respectively.

D-grid discretization of the momentum equation

We introduce the following average operators for the covariant wind components in x and y directions, respectively:

$$\overline{\mathfrak{U}^x_{i,j+\frac{1}{2}}}(t) = \frac{1}{\Delta x}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \mathfrak{U}(x, y_{j+\frac{1}{2}}, t)\, dx, \quad i = 1, \ldots, N, j = 0, \ldots, N,$$ (6.11)

$$\overline{\mathfrak{V}^y_{i+\frac{1}{2},j}}(t) = \frac{1}{\Delta y}\int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}} \mathfrak{V}(x_{i+\frac{1}{2}}, y, t)\, dy, \quad i = 0, \ldots, N, j = 1, \ldots, N.$$ (6.12)

We shall also use the notation $q_{ij}(t) = q(x_i, y_j, t)$ for any function q and integer or half-integer indices i and j. We also use the centered difference notations $\delta_i q_{ij}(t) = q_{i+\frac{1}{2},j}(t) - q_{i-\frac{1}{2},j}(t)$ and $\delta_j q_{ij}(t) = q_{i,j+\frac{1}{2}}(t) - q_{i,j-\frac{1}{2}}(t)$ for any integer or half-integer indices i and j.

By integrating Equation (6.2) with respect to x on $[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]$ and Equation (6.3) with

respect to y on $[y_{j-\frac{1}{2}}, y_{j+\frac{1}{2}}]$, we get the following equations:

$$\frac{d}{dt}\overline{\mathfrak{U}^x_{i,j+\frac{1}{2}}}(t) = -\frac{\delta_i K_{i,j+\frac{1}{2}}(t)}{\Delta x} - \frac{\delta_i \Phi_{i,j+\frac{1}{2}}(t)}{\Delta x} + \frac{1}{\Delta x}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} (\mathfrak{v}\sqrt{\mathfrak{g}}\xi)(x_i, y_{j+\frac{1}{2}}, t)\,dx, \qquad (6.13)$$

$$\frac{d}{dt}\overline{\mathfrak{V}^y_{i+\frac{1}{2},j}}(t) = -\frac{\delta_j K_{i+\frac{1}{2},j}(t)}{\Delta y} - \frac{\delta_j \Phi_{i+\frac{1}{2},j}(t)}{\Delta y} - \frac{1}{\Delta y}\int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}} (\mathfrak{u}\sqrt{\mathfrak{g}}\xi)(x_{i+\frac{1}{2}}, y, t)\,dy. \qquad (6.14)$$

Integrating Equations (6.13) and (6.14) on time over $[t^n, t^{n+1}]$, we obtain:

$$\overline{\mathfrak{U}^x_{i,j+\frac{1}{2}}}(t^{n+1}) = \overline{\mathfrak{U}^x_{i,j+\frac{1}{2}}}(t^n) - \int_{t_n}^{t_{n+1}} \left[\frac{\delta_i K_{i,j+\frac{1}{2}}(t)}{\Delta x} + \frac{\delta_i \Phi_{i,j+\frac{1}{2}}(t)}{\Delta x} - \left(\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \frac{(\mathfrak{v}\sqrt{\mathfrak{g}}\xi)(x_i, y_{j+\frac{1}{2}}, t)}{\Delta x}\,dx\right)\right]dt,$$

$$(6.15)$$

$$\overline{\mathfrak{V}^y_{i+\frac{1}{2},j}}(t^{n+1}) = \overline{\mathfrak{V}^y_{i+\frac{1}{2},j}}(t^n) - \int_{t_n}^{t_{n+1}} \left[\frac{\delta_j K_{i+\frac{1}{2},j}(t)}{\Delta y} + \frac{\delta_j \Phi_{i+\frac{1}{2},j}(t)}{\Delta y} + \left(\int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}} \frac{(\mathfrak{u}\sqrt{\mathfrak{g}}\xi)(x_{i+\frac{1}{2}}, y, t)}{\Delta y}\,dy\right)\right]dt.$$

$$(6.16)$$

Using the midpoint rule and using the normalized covariant winds in Equations (6.15) and (6.16), we derive a general scheme to update the normalized D-grid covariant winds:

$$U^{n+1}_{i,j+\frac{1}{2}} = U^n_{i,j+\frac{1}{2}} - \left(\frac{\delta_i K^n_{i,j+\frac{1}{2}}}{\hat{\delta} x_{i,j+\frac{1}{2}}} + \frac{\delta_i \Phi^n_{i,j+\frac{1}{2}}}{\hat{\delta} x_{i,j+\frac{1}{2}}} - \frac{G_{i,j+\frac{1}{2}}}{\hat{\delta} x_{i,j+\frac{1}{2}}}[\xi, v^n]\right), \qquad (6.17)$$

$$V^{n+1}_{i+\frac{1}{2},j} = V^n_{i+\frac{1}{2},j} - \left(\frac{\delta_j K^n_{i+\frac{1}{2},j}}{\hat{\delta} y_{i+\frac{1}{2},j}} + \frac{\delta_j \Phi^n_{i+\frac{1}{2},j}}{\hat{\delta} y_{i+\frac{1}{2},j}} + \frac{F_{i+\frac{1}{2},j}[\xi, u^n]}{\hat{\delta} y_{i+\frac{1}{2},j}}\right). \qquad (6.18)$$

Then, this schemes requires an approximation of the time-averaged kinetic energy at the B-grid:

$$K^n_{i+\frac{1}{2},j+\frac{1}{2}} \approx \frac{1}{2}\left[\int_{t_n}^{t_{n+1}} (uU)(x_{i+\frac{1}{2}}, y_{j+\frac{1}{2}}, t)\,dt + \int_{t_n}^{t_{n+1}} (vV)(x_{i+\frac{1}{2}}, y_{j+\frac{1}{2}}, t)\,dt\right], \qquad (6.19)$$

and an approximation of the time-averaged geopotential on B-grid:

$$\Phi^n_{i+\frac{1}{2},j+\frac{1}{2}} \approx \int_{t_n}^{t_{n+1}} \Phi(x_{i+\frac{1}{2}}, y_{j+\frac{1}{2}}, t)\,dt, \qquad (6.20)$$

The terms should $F_{i+\frac{1}{2},j}[\xi^n, u^n]$ and $G_{i,j+\frac{1}{2}}[\xi^n, v^n]$ should approximate the time-averaged absolute vorticity fluxes:

$$F_{i+\frac{1}{2},j}[\xi^n, u^n] \approx \int_{t^n}^{t^{n+1}} \int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}} (\mathfrak{u}\sqrt{\mathfrak{g}}\xi)(x_{i+\frac{1}{2}}, y, t)\,dy\,dt \qquad (6.21)$$

$$G_{i,j+\frac{1}{2}}[\xi^n, v^n] \approx \int_{t^n}^{t^{n+1}} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} (\mathfrak{v}\sqrt{\mathfrak{g}}\xi)(x, y_{j+\frac{1}{2}}, t)\,dx\,dt. \qquad (6.22)$$

Notice that since ξ satisfies the advection equation (6.8), these integrals may be approxi-

mated using finite-volume fluxes assuming that ξ may be advected on the A-grid. Indeed, this is possible because, as we shall see soon, the D-grid covariant wind facilitates the estimation of ξ on the A-grid by using centered finite differences. All these approximations needed for the D-grid scheme are described in Section 6.2.2.

C-grid discretization of the momentum equation

Similar to the derivation of the D-grid covariant wind scheme, we may deduce the following C-grid covariant wind scheme for a half-time step:

$$U^{n+1}_{i+\frac{1}{2},j} = U^n_{i+\frac{1}{2},j} - \left(\frac{\delta_i K^n_{i+\frac{1}{2},j}}{\hat{\delta} x_{i+\frac{1}{2},j}} + \frac{\delta_i \Phi^n_{i+\frac{1}{2},j}}{\hat{\delta} x_{i+\frac{1}{2},j}} - \frac{G_{i+\frac{1}{2},j}[\xi^n, v^n]}{\hat{\delta} x_{i+\frac{1}{2},j}} \right), \tag{6.23}$$

$$V^{n+1}_{i,j+\frac{1}{2}} = V^n_{i,j+\frac{1}{2}} - \left(\frac{\delta_j K_{i,j+\frac{1}{2}}}{\hat{\delta} y_{i,j+\frac{1}{2}}} + \frac{\delta_j \Phi^n_{i,j+\frac{1}{2}}}{\hat{\delta} y_{i,j+\frac{1}{2}}} + \frac{F_{i,j+\frac{1}{2}}[\xi, u^n]}{\hat{\delta} y_{i,j+\frac{1}{2}}} \right). \tag{6.24}$$

Then, this schemes requires an approximation of the time-averaged kinetic energy at the A-grid

$$K^n_{ij} \approx \frac{1}{2} \left[\int_{t_n}^{t_{n+\frac{1}{2}}} (uU)(x_i, y_j, t)\, dt + \int_{t_n}^{t_{n+\frac{1}{2}}} (vV)(x_i, y_j, t)\, dt \right], \tag{6.25}$$

and an approximation of the time-averaged geopotential on A-grid points:

$$\Phi^n_{ij} \approx \int_{t_n}^{t_{n+\frac{1}{2}}} \Phi(x_i, y_j, t)\, dt. \tag{6.26}$$

The terms should $F_{i,j+\frac{1}{2}}[\xi^n, u^n]$ and $G_{i+\frac{1}{2},j}[\xi^n, v^n]$ should approximate the time-averaged absolute vorticity fluxes:

$$F_{i,j+\frac{1}{2}}[\xi^n, u^n] \approx \int_{t^n}^{t^{n+\frac{1}{2}}} \int_{y_j}^{y_{j+1}} (\mathfrak{u} \sqrt{\mathfrak{g}} \xi)(x_i, y, t)\, dy\, dt \tag{6.27}$$

$$G_{i+\frac{1}{2},j}[\xi^n, v^n] \approx \int_{t^n}^{t^{n+\frac{1}{2}}} \int_{x_i}^{x_{i+1}} (\mathfrak{v} \sqrt{\mathfrak{g}} \xi)(x, y_j, t)\, dx\, dt. \tag{6.28}$$

Once again, since ξ satisfies the advection equation (6.8), these integrals may be approximated using finite-volume fluxes assuming that ξ may be advected on the B-grid. Indeed, this is possible because, as we shall see soon, the C-grid covariant wind facilitates the estimation of ξ on the B-grid by using centered finite differences. All these approximations are described in Section 6.2.1.

6.2   The FV3 shallow-water solver

This Section is dedicated to presenting all the details of the shallow-water solver proposed by Lin and Rood (1997) on the cubed-sphere. The C-grid intermediate step is described in Section 6.2.1, while the D-grid step is detailed in Section 6.2.2.

6.2.1 C-grid intermediate step

The C-grid intermediate step serves to provide the C-grid contravariant winds centered at time $u_{i+\frac{1}{2},j}^{n+\frac{1}{2}}$ and $v_{i,j+\frac{1}{2}}^{n+\frac{1}{2}}$ that are required by the advection fluxes when using PPM, as discussed in Chapters 3 and 5. One could utilize second-order extrapolation to obtain these centered at time C-grid winds, using two time levels, namely:

$$u_{i+\frac{1}{2},j}^{n+\frac{1}{2}} = \frac{3}{2}u_{i,j+\frac{1}{2}}^{n} - \frac{1}{2}u_{i,j+\frac{1}{2}}^{n-1}, \tag{6.29}$$

$$v_{i,j+\frac{1}{2}}^{n+\frac{1}{2}} = \frac{3}{2}v_{i,j+\frac{1}{2}}^{n} - \frac{1}{2}v_{i,j+\frac{1}{2}}^{n-1}. \tag{6.30}$$

This approach is very popular in Semi-Lagrangian methods. However, as pointed out by Lin and Rood (1997), this extrapolation introduces $2\Delta x$ numerical noise, which may degrade the solution in presence of sharp bottom topography. Therefore, Lin and Rood (1997) proposes solving the SWE on a C-grid for a half-time step to provide the winds centered at time $n + \frac{1}{2}$. To make this half-time step cheaper, upwind fluxes are going to be used. Our goal now is to describe the details of this C-grid wind solver. We are going to describe everything that is needed to advance the C-grid winds given by Equations (6.23) and (6.24).

Wind interpolation

We are given the D-grid covariant wind, that is, we have $U_{i,j+\frac{1}{2}}^{n}$ for $i = 0, \dots, N$, $j = 1, \dots, N$, and $V_{i+\frac{1}{2},j}^{n}$ $j = 0, \dots, N$, $i = 1, \dots, N$. We may then use the duo-grid interpolation (Section 4.3.3) to get the values on the duo-grid. After that, we have all the values $U_{i,j+\frac{1}{2}}^{n}$ for $i = 0, \dots, N + v$, $j = 1, \dots, N + v$, and $V_{i+\frac{1}{2},j}^{n}$ $j = 0, \dots, N + v$, $i = 1, \dots, N + v$.

We define the average operator in the x direction as:

$$\overline{q_{ij}}^{x} = \begin{cases} 0.5(q_{i+\frac{1}{2},j} + q_{i-\frac{1}{2},j}), & \text{if } i = -v + 1 \text{ or } i = N + v, \\ \frac{9}{16}(q_{i+\frac{1}{2},j} + q_{i-\frac{1}{2},j}) - \frac{1}{16}(q_{i+\frac{3}{2},j} + q_{i-\frac{3}{2},j}), & \text{otherwise}, \end{cases} \tag{6.31}$$

for any integer i and integer or half integer j, and

$$\overline{q_{i+\frac{1}{2},j}}^{x} = \begin{cases} 0.5(q_{i+1,j} + q_{ij}), & \text{if } i = -v + 1 \text{ or } i = N + v, \\ \frac{9}{16}(q_{i+1,j} + q_{ij}) - \frac{1}{16}(q_{i+2,j} + q_{i-1,j}), & \text{otherwise}, \end{cases} \tag{6.32}$$

for any integer i and integer or half integer j. The average operator $\overline{q_{ij}}^{y}$ in the y direction is defined analogously.

We may interpolate the normalized covariant component U from the D-grid to A-grid by using the average in the y direction:

$$U_{ij}^{n} = \overline{U_{ij}^{n}}^{y}, \tag{6.33}$$

for $i = -v + 1, \dots, N + v$, $j = -v + 1, \dots, N + v$, and similarly to the V component:

$$V_{ij}^n = \overline{V_{ij}^n}^x, \tag{6.34}$$

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

Moreover, using the A-grid covariant wind, we may convert the wind from covariant to contravariant on the A-grid representation using Equation (4.47):

$$u_{ij}^n = \frac{1}{\sin^2 \alpha_{ij}} \left(U_{ij}^n - \cos \alpha_{ij} V_{ij}^n \right), \tag{6.35}$$

for $i = -v + 1, \dots, N + v$, $j = -v + 1, \dots, N + v$, and similarly to the v component:

$$v_{ij}^n = \frac{1}{\sin^2 \alpha_{ij}} \left(V_{ij}^n - \cos \alpha_{ij} U_{ij}^n \right), \tag{6.36}$$

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

Using the A-grid covariant wind, we may interpolate it to the C-grid covariant wind as:

$$U_{i+\frac{1}{2},j}^n = \overline{U_{i+\frac{1}{2},j}^n}^x, \tag{6.37}$$

for $i = -v + 2, \dots, N + v$, $j = -v + 1, \dots, N + v$, and similarly to the V component:

$$V_{i,j+\frac{1}{2}}^n = \overline{V_{i,j+\frac{1}{2}}^n}^y, \tag{6.38}$$

for $i = -v + 1, \dots, N + v$, $j = -v + 2, \dots, N + v$.

Then, we may get the C-grid covariant wind using the original D-grid contravariant wind:

$$u_{i+\frac{1}{2},j}^n = \frac{1}{\sin^2 \alpha_{i+\frac{1}{2},j}} \left(U_{i+\frac{1}{2},j}^n - \cos \alpha_{i+\frac{1}{2},j} V_{i+\frac{1}{2},j}^n \right), \tag{6.39}$$

for $i = -v + 2, \dots, N + v$, $j = -v + 1, \dots, N + v$, and similarly to the v component:

$$v_{i,j+\frac{1}{2}}^n = \frac{1}{\sin^2 \alpha_{i,j+\frac{1}{2}}} \left(V_{i,j+\frac{1}{2}}^n - \cos \alpha_{i,j+\frac{1}{2}} U_{i,j+\frac{1}{2}}^n \right), \tag{6.40}$$

for $i = -v + 1, \dots, N + v$, $j = -v + 2, \dots, N + v$.

And similarly, we obtain the D-grid contravariant wind:

$$v_{i+\frac{1}{2},j}^n = \frac{1}{\sin^2 \alpha_{i+\frac{1}{2},j}} \left(V_{i+\frac{1}{2},j}^n - \cos \alpha_{i+\frac{1}{2},j} U_{i+\frac{1}{2},j}^n \right), \tag{6.41}$$

for $i = -v + 2, \dots, N + v$, $j = -v + 1, \dots, N + v$, and similarly to the u component:

$$u^n_{i,j+\frac{1}{2}} = \frac{1}{\sin^2 \alpha_{i,j+\frac{1}{2}}} \left(U^n_{i,j+\frac{1}{2}} - \cos \alpha_{i,j+\frac{1}{2}} V^n_{i,j+\frac{1}{2}} \right), \qquad (6.42)$$

Fluid depth

The fluid depth is update using the upwind scheme, expressed as:

$$h^{n+\frac{1}{2}}_{ij} = h^n_{ij} + \mathbf{F}^{UPW}_{ij}[h^n, u^n] + \mathbf{G}^{UPW}_{ij}[h^n, v^n], \qquad (6.43)$$

for $i, j = 0, \dots, N + 1$, where the upwind update operators are given by

$$\mathbf{F}^{UPW}_{ij}[h^n, u^n] = -\frac{1}{|\hat{\Omega}_{ij}|} \left(\mathcal{A}^x_{i+\frac{1}{2},j} \mathcal{F}^{UPW,x}_{i+\frac{1}{2},j}[h^n, u^n] - \mathcal{A}^x_{i-\frac{1}{2},j} \mathcal{F}^{UPW,x}_{i-\frac{1}{2},j}[h^n, u^n] \right), \qquad (6.44)$$

and

$$\mathbf{G}^{UPW}_{ij}[h^n, v^n] = -\frac{1}{|\hat{\Omega}_{ij}|} \left(\mathcal{A}^y_{i,j+\frac{1}{2}} \mathcal{F}^{UPW,y}_{i,j+\frac{1}{2}}[h^n, v^n] - \mathcal{A}^y_{i,j-\frac{1}{2}} \mathcal{F}^{UPW,y}_{i,j-\frac{1}{2}}[h^n, v^n] \right), \qquad (6.45)$$

where

$$\mathcal{A}^x_{i+\frac{1}{2},j} = \frac{\Delta t}{2} \times \begin{cases} \Delta y \sqrt{\mathfrak{g}}_{i+\frac{1}{2},j} u^n_{i+\frac{1}{2},j} = \hat{\delta} y_{i+\frac{1}{2},j} \sin \alpha_{i+\frac{1}{2},j} u^n_{i+\frac{1}{2},j} & \text{for mt0,} \\ \Delta y u^n_{i+\frac{1}{2},j} & \text{for mt1,} \end{cases} \qquad (6.46)$$

and

$$\mathcal{F}^{UPW,x}_{i+\frac{1}{2},j}[h^n, u^n] = \begin{cases} \mathfrak{F}^{UPW,x}_{i+\frac{1}{2},j}[\sqrt{\mathfrak{g}} h^n, u^n], & \text{for mt0,} \\ \mathfrak{F}^{UPW,x}_{i+\frac{1}{2},j}[h^n, u^n], & \text{for mt1,} \end{cases} \qquad (6.47)$$

where the 1D upwind flux in the x direction is defined by:

$$\mathfrak{F}^{UPW,x}_{i+\frac{1}{2},j}[\psi^n, u^n] = \begin{cases} \psi^n_{ij} & \text{if} \quad u^n_{i+\frac{1}{2},j} > 0, \\ \psi^n_{i+1,j} & \text{if} \quad u^n_{i+\frac{1}{2},j} \leq 0, \end{cases} \qquad (6.48)$$

for $i = 0, \dots, N$, $j = -v + 1, \dots, N + v$. The terms, $\mathcal{A}^y_{i,j+\frac{1}{2}}$ and $\mathcal{F}^{UPW,y}_{i,j+\frac{1}{2}}$ are defined similarly using v.

We recall the metric term discussion of PPM presented in Section 5.3.1 is also valid for the upwind flux, and therefore the same methods of metric term formulation, mt0 and mt1, presented there are valid in this context.

Geopotential gradient

Once we have computed $h_{ij}^{n+\frac{1}{2}}$, we are able to estimate the time-averaged geopotential (Equation (6.26)) on the A-grid as:

$$\Phi_{ij}^n = \Delta t\, g(h_{ij}^{n+\frac{1}{2}} + b_{ij}), \tag{6.49}$$

for $i, j = 0, \dots, N + 1$. We use $h_{ij}^{n+\frac{1}{2}}$ instead of h_{ij}^n so the C-grid scheme becomes backward-forward in time; otherwise, the C-grid scheme would be unconditionally unstable (Lin & Rood, 1997). Following that, we estimate the geopotential gradient on the edge midpoints by using centered differences:

$$\delta_i \Phi_{i+\frac{1}{2},j}^n = \Phi_{i+1,j}^n - \Phi_{ij}^n, \quad i = 0, \dots, N, j = 1, \dots, N, \tag{6.50}$$

$$\delta_j \Phi_{i,j+\frac{1}{2}}^n = \Phi_{i,j+1}^n - \Phi_{ij}^n, \quad i = 1, \dots, N, j = 0, \dots, N. \tag{6.51}$$

Absolute vorticity fluxes

Using the C-grid covariant winds $(U_{i+\frac{1}{2},j}^n, V_{i,j+\frac{1}{2}}^n)$, we may compute the relative vorticity (Equation (6.7)) at the B-grid using a centered finite difference:

$$
\begin{aligned}
\zeta_{i+\frac{1}{2},j+\frac{1}{2}}^n &= \frac{1}{\sqrt{\mathfrak{g}}_{i+\frac{1}{2},j+\frac{1}{2}}} \left[\frac{\mathfrak{V}_{i+1,j+\frac{1}{2}}^n - \mathfrak{V}_{i,j+\frac{1}{2}}^n}{\Delta x} - \frac{\mathfrak{U}_{i+\frac{1}{2},j+1}^n - \mathfrak{U}_{i+\frac{1}{2},j}^n}{\Delta y} \right] \\
&= \frac{1}{|\hat{\Omega}_{i+\frac{1}{2},j+\frac{1}{2}}|} \left[\left(\hat{\delta} y_{i+1,j+\frac{1}{2}} V_{i+1,j+\frac{1}{2}}^n - \hat{\delta} y_{i,j+\frac{1}{2}} V_{i,j+\frac{1}{2}}^n \right) - \left(\hat{\delta} x_{i+\frac{1}{2},j+1} U_{i+\frac{1}{2},j+1}^n - \hat{\delta} x_{i+\frac{1}{2},j} U_{i+\frac{1}{2},j}^n \right) \right],
\end{aligned}
\tag{6.52}
$$

for $i, j = -1, \dots, N + 1$. Then, we obtain the absolute vorticity on the B-grid as:

$$\xi_{i+\frac{1}{2},j+\frac{1}{2}}^n = f_{i+\frac{1}{2},j+\frac{1}{2}} + \zeta_{i+\frac{1}{2},j+\frac{1}{2}}^n, \tag{6.53}$$

for $i, j = -1, \dots, N + 1$. Thus, it follows from Equation (6.8) that absolute vorticity may be updated on the B-grid as follows using the upwind flux:

$$\xi_{i+\frac{1}{2},j+\frac{1}{2}}^{n+1} = \xi_{i+\frac{1}{2},j+\frac{1}{2}}^n + \mathbf{F}_{i+\frac{1}{2},j+\frac{1}{2}}^{UPW}[\xi^n, u^n] + \mathbf{G}_{i+\frac{1}{2},j+\frac{1}{2}}^{UPW}[\xi^n, v^n], \tag{6.54}$$

for $i, j = 0, \dots, N$. The upwind update operators on the B-grid are given by

$$\mathbf{F}_{i+\frac{1}{2},j+\frac{1}{2}}^{UPW}[\xi^n, u^n] = \frac{-1}{|\hat{\Omega}_{i+\frac{1}{2},j+\frac{1}{2}}|} \left(\mathcal{A}_{i+1,j+\frac{1}{2}}^x \mathcal{F}_{i+1,j+\frac{1}{2}}^{UPW,x}[\xi^n, u^n] - \mathcal{A}_{i,j+\frac{1}{2}}^x \mathcal{F}_{i,j+\frac{1}{2}}^{UPW,x}[\xi^n, u^n] \right), \tag{6.55}$$

and

$$\mathbf{G}_{i+\frac{1}{2},j+\frac{1}{2}}^{UPW}[\zeta^n, v^n] = \frac{-1}{|\hat{\Omega}_{i+\frac{1}{2},j+\frac{1}{2}}|} \left(\mathcal{A}_{i+\frac{1}{2},j+1}^y \mathcal{F}_{i+\frac{1}{2},j+1}^{UPW,y}[\xi^n, v^n] - \mathcal{A}_{i+\frac{1}{2},j}^y \mathcal{F}_{i+\frac{1}{2},j}^{UPW,y}[\xi^n, v^n] \right), \tag{6.56}$$

where

$$
\mathcal{A}^x_{i,j+\frac{1}{2}} = \frac{\Delta t}{2} \times
\begin{cases}
\Delta y \sqrt{\mathfrak{g}}_{i,j+\frac{1}{2}} u^n_{i,j+\frac{1}{2}} = \hat{\delta} y_{i,j+\frac{1}{2}} \sin \alpha_{i,j+\frac{1}{2}} u^n_{i,j+\frac{1}{2}} & \text{for mt0,} \\
\Delta y u^n_{i,j+\frac{1}{2}} & \text{for mt1,}
\end{cases}
\tag{6.57}
$$

and

$$
\mathcal{F}^{UPW,x}_{i,j+\frac{1}{2}}[\xi^n, u^n] =
\begin{cases}
\mathfrak{F}^{UPW,x}_{i,j+\frac{1}{2}}[\sqrt{\mathfrak{g}}\xi^n, u^n], & \text{for mt0,} \\
\mathfrak{F}^{UPW,x}_{i,j+\frac{1}{2}}[\xi^n, u^n], & \text{for mt1,}
\end{cases}
\tag{6.58}
$$

where the 1D upwind flux in the x direction is defined by:

$$
\mathfrak{F}^{UPW,x}_{i,j+\frac{1}{2}}[\psi^n, u^n] =
\begin{cases}
\psi^n_{i-\frac{1}{2},j+\frac{1}{2}} & \text{if} \quad u^n_{i,j+\frac{1}{2}} > 0, \\
\psi^n_{i+\frac{1}{2},j+\frac{1}{2}} & \text{if} \quad u^n_{i,j+\frac{1}{2}} \leq 0.
\end{cases}
\tag{6.59}
$$

The terms $\mathcal{A}^y_{i+\frac{1}{2},j}$ and $\mathcal{F}^{UPW,y}_{i+\frac{1}{2},j}$ are defined similarly using v. Notice that the D-grid contravariant winds $(u^n_{i,j+\frac{1}{2}}, v^n_{i+\frac{1}{2},j})$ are needed for the upwind flux on the B-grid.

Finally we point out that we do not need to update the absolute vorticity using Equation (6.54), instead, we only need to compute the terms and $\mathcal{A}^x_{i,j+\frac{1}{2}}$ and $\mathcal{F}^{UPW,x}_{i,j+\frac{1}{2}}$ for $i = 0, \dots, N$, $j = 1, \dots, N$, and $\mathcal{A}^y_{i+\frac{1}{2},j}$ and $\mathcal{F}^{UPW,y}_{i+\frac{1}{2},j}$ for $i = 1, \dots, N$, $j = 0, \dots, N$, to update the C-grid winds using the momentum equations (6.23) and (6.24). That is, we only need to compute the terms:

$$
F_{i,j+\frac{1}{2}}[\xi^n, u^n] = \mathcal{A}^x_{i,j+\frac{1}{2}} \mathcal{F}^{UPW,x}_{i,j+\frac{1}{2}}[\xi^n, u^n],
\tag{6.60}
$$

$$
G_{i+\frac{1}{2},j}[\xi^n, v^n] = \mathcal{A}^y_{i+\frac{1}{2},j} \mathcal{F}^{UPW,y}_{i+\frac{1}{2},j}[\xi^n, v^n].
\tag{6.61}
$$

Kinetic energy fluxes

To estimate the kinetic energy fluxes, we need to estimate the temporal integrals in Equation (6.25). In Lin and Rood (1997) and in the current FV3 implementation, it is assumed that the u and U obeys:

$$
\partial_t U + \partial_x (uU)(x, y_j, t) = 0,
\tag{6.62}
$$

then, using an 1D finite-volume numerical flux $F^x_{i+\frac{1}{2},j}$ (recall Problem 2.4), we may approximate

$$
F^x_{i+\frac{1}{2},j}[U^n, u^n] \approx \frac{1}{0.5\Delta t} \int_{t_n}^{t_{n+\frac{1}{2}}} uU(x_{i+\frac{1}{2}}, y_j, t)\, dt
\tag{6.63}
$$

Similarly for v and V, we use $F^y_{i,j+\frac{1}{2}}[V^n, v^n]$ an then we have an estimation for the time-averaged kinetic energy. Of course, Equation (6.62) is not true, but it is used to advected the wind on a upwind direction.

Therefore, the time-averaged kinetic on the A-grid is computed using the for-

mula:

$$K_{ij}^n = \frac{0.5\Delta t}{2}\left(u_{ij}^n \mathfrak{F}_{ij}^{UPW,x}[U^n, u^n] + v_{ij}^n \mathfrak{F}_{ij}^{UPW,y}[V^n, v^n]\right), \tag{6.64}$$

for $i, j = 0, \dots, N+1$, where we have the 1D upwind flux in the x direction

$$\mathfrak{F}_{ij}^{UPW,x}[U^n, u^n] = \begin{cases} U_{i-\frac{1}{2},j}^n & \text{if} \quad u_{ij}^n > 0, \\ U_{i+\frac{1}{2},j}^n & \text{if} \quad u_{ij}^n \le 0, \end{cases} \tag{6.65}$$

and the 1D upwind flux in the y direction

$$\mathfrak{F}_{ij}^{UPW,y}[V^n, v^n] = \begin{cases} V_{i,j-\frac{1}{2}}^n & \text{if} \quad v_{ij}^n > 0, \\ V_{i,j+\frac{1}{2}}^n & \text{if} \quad v_{ij}^n \le 0. \end{cases} \tag{6.66}$$

In this step, we use the A-grid contravariant winds u_{ij}^n, v_{ij}^n obtained in Equations (6.35) and (6.36). We also use the C-grid covariant winds obtained in Equations (6.37) and (6.38). Thus, we estimate the kinetic energy gradient using a centered difference:

$$\delta_i K_{i+\frac{1}{2},j}^n = K_{i+1,j}^n - K_{ij}^n, \quad i = 0, \dots, N, j = 1, \dots, N, \tag{6.67}$$

$$\delta_j K_{i,j+\frac{1}{2}}^n = K_{i,j+1}^n - K_{ij}^n, \quad i = 1, \dots, N, j = 0, \dots, N. \tag{6.68}$$

Hence, we have completed the description of the C-grid wind update on a half-step using Equations (6.23) and (6.24).

6.2.2 D-grid step

Now we are going to describe how we can advance the D-grid scheme, given by Equations (6.17) and (6.18), using the C-grid winds $U_{i+\frac{1}{2},j}^{n+\frac{1}{2}}$ and $V_{i,j+\frac{1}{2}}^{n+\frac{1}{2}}$ centered at time obtained by the C-grid solver.

Wind interpolation

We are given the normalized covariant wind components on a C-grid that is, we have $V_{i,j+\frac{1}{2}}^{n+\frac{1}{2}}$ for $i = 0, \dots, N$, $j = 1, \dots, N$, and $U_{i+\frac{1}{2},j}^{n+\frac{1}{2}}$ $j = 0, \dots, N$, $i = 1, \dots, N$, obtained in the C-grid intermediate step. We may then use the duo-grid interpolation (Section 4.3.3) to get the values on the duo-grid. After that, we have all the values $V_{i,j+\frac{1}{2}}^{n+\frac{1}{2}}$ for $i = 0, \dots, N+v$, $j = 1, \dots, N+v$, and $U_{i+\frac{1}{2},j}^{n+\frac{1}{2}}$ $j = 0, \dots, N+v$, $i = 1, \dots, N+v$.

We may interpolate the covariant wind V from D-grid points to C-grid points as

$$V_{i+\frac{1}{2},j}^{n+\frac{1}{2}} = \frac{1}{4}\left(V_{i,j+\frac{1}{2}}^{n+\frac{1}{2}} + V_{i+1,j+\frac{1}{2}}^{n+\frac{1}{2}} + V_{i,j-\frac{1}{2}}^{n+\frac{1}{2}} + V_{i+1,j-\frac{1}{2}}^{n+\frac{1}{2}}\right), \tag{6.69}$$

for $i = -1, \dots, N+2$, $j = -v+1, \dots, N+v$, and similarly to the covariant wind U from

C-grid points to D-grid points as

$$U_{i,j+\frac{1}{2}}^{n+\frac{1}{2}} = \frac{1}{4}\left(U_{i+\frac{1}{2},j}^{n+\frac{1}{2}} + U_{i+\frac{1}{2},j+1}^{n+\frac{1}{2}} + U_{i-\frac{1}{2},j}^{n+\frac{1}{2}} + U_{i-\frac{1}{2},j+1}^{n+\frac{1}{2}}\right),$$

(6.70)

for $j = -1, \dots, N + 2, i = -\nu + 1, \dots, N + \nu$.

Thus, we obtain and similarly to the contravariant wind u at C-grid points applying Equation (4.47):

$$u_{i+\frac{1}{2},j}^{n+\frac{1}{2}} = \frac{1}{\sin^2 \alpha_{i+\frac{1}{2},j}}\left(U_{i+\frac{1}{2},j}^{n+\frac{1}{2}} - \cos \alpha_{i+\frac{1}{2},j} V_{i+\frac{1}{2},j}^{n+\frac{1}{2}}\right),$$

(6.71)

for $i = -1, \dots, N + 2, j = -\nu + 1, \dots, N + \nu$, and similarly to the contravariant wind v at D-grid points:

$$v_{i,j+\frac{1}{2}}^{n+\frac{1}{2}} = \frac{1}{\sin^2 \alpha_{i,j+\frac{1}{2}}}\left(V_{i,j+\frac{1}{2}}^{n+\frac{1}{2}} - \cos \alpha_{i,j+\frac{1}{2}} U_{i,j+\frac{1}{2}}^{n+\frac{1}{2}}\right).$$

(6.72)

for $j = -1, \dots, N + 2, i = -\nu + 1, \dots, N + \nu$. Hence, we have the C-grid contravariant time-averaged winds that are needed for the flux operators.

For the kinetic energy fluxes, we require B-grid winds. For this reason, we compute the B-grid covariant wind using:

$$U_{i+\frac{1}{2},j+\frac{1}{2}}^{n+\frac{1}{2}} = \frac{1}{2}\left(U_{i+\frac{1}{2},j+1}^{n+\frac{1}{2}} + U_{i+\frac{1}{2},j}^{n+\frac{1}{2}}\right),$$

(6.73)

for $i = -1, \dots, N + 2, j = -\nu + 1, \dots, N + \nu$, and

$$V_{i+\frac{1}{2},j+\frac{1}{2}}^{n+\frac{1}{2}} = \frac{1}{2}\left(V_{i,j+\frac{1}{2}}^{n+\frac{1}{2}} + V_{i+1,j+\frac{1}{2}}^{n+\frac{1}{2}}\right),$$

(6.74)

and then we convert the winds from covariant to contravariant as

$$u_{i+\frac{1}{2},j+\frac{1}{2}}^{n+\frac{1}{2}} = \frac{1}{\sin^2 \alpha_{i+\frac{1}{2},j+\frac{1}{2}}}\left(U_{i+\frac{1}{2},j+\frac{1}{2}}^{n+\frac{1}{2}} - \cos \alpha_{i+\frac{1}{2},j+\frac{1}{2}} V_{i+\frac{1}{2},j+\frac{1}{2}}^{n+\frac{1}{2}}\right),$$

(6.75)

for $i = -1, \dots, N + 2, j = -\nu + 1, \dots, N + \nu$, and similarly to the contravariant wind v at D-grid points:

$$v_{i+\frac{1}{2},j+\frac{1}{2}}^{n+\frac{1}{2}} = \frac{1}{\sin^2 \alpha_{i+\frac{1}{2},j+\frac{1}{2}}}\left(V_{i+\frac{1}{2},j+\frac{1}{2}}^{n+\frac{1}{2}} - \cos \alpha_{i+\frac{1}{2},j+\frac{1}{2}} U_{i+\frac{1}{2},j+\frac{1}{2}}^{n+\frac{1}{2}}\right).$$

(6.76)

Fluid depth

The fluid depth is updated using the dimension-splitting method with PPM, as given by Equations (5.26). This yields the values of h_{ij}^{n+1} for $i, j = 1, \dots, N$. Then, we may generate its ghost cell values using the duo-grid interpolation.

Geopotential gradient

Once we have computed h_{ij}^{n+1}, we are able to estimate the time-averaged geopotential on the A-grid as:

$$\Phi_{ij}^n = \Delta t g(h_{ij}^{n+1} + b_{ij}), \tag{6.77}$$

for $i, j = -\nu + 1, \dots, N + \nu$. Then, the time-averaged geopotential on the B-grid (Equation (6.20)) may be using interpolation:

$$\Phi_{i+\frac{1}{2},j+\frac{1}{2}}^n = \frac{\overline{\overline{\Phi_{i+\frac{1}{2},j+\frac{1}{2}}^n}}^{xy} + \overline{\overline{\Phi_{i+\frac{1}{2},j+\frac{1}{2}}^n}}^{yx}}{2}. \tag{6.78}$$

Again, we use h_{ij}^{n+1} instead of h_{ij}^n so the D-grid scheme also becomes backward-forward in time, avoiding numerical instability. Following that, we estimate the geopotential gradient on the D-grid and C-grid points, respectively, by using centered differences:

$$\delta_i \Phi_{i,j+\frac{1}{2}}^n = \Phi_{i+\frac{1}{2},j+\frac{1}{2}}^n - \Phi_{i-\frac{1}{2},j+\frac{1}{2}}^{n+1}, \quad i = 1, \dots, N, j = 0, \dots, N, \tag{6.79}$$

$$\delta_j \Phi_{i+\frac{1}{2},j}^n = \Phi_{i+\frac{1}{2},j+\frac{1}{2}}^n - \Phi_{i+\frac{1}{2},j-\frac{1}{2}}^n, \quad i = 0, \dots, N, j = 1, \dots, N. \tag{6.80}$$

Absolute vorticity fluxes

Using the D-grid covariant winds $(U_{i,j+\frac{1}{2}}^n, V_{i+\frac{1}{2},j}^n)$, we may compute the relative vorticity (Equation (6.7)) at the A-grid points using a centered finite difference:

$$\begin{aligned}\zeta_{ij}^n &= \frac{1}{\sqrt{\mathfrak{g}}_{ij}}\left[\frac{\mathfrak{V}_{i+\frac{1}{2},j}^n - \mathfrak{V}_{i-\frac{1}{2},j}^n}{\Delta x} - \frac{\mathfrak{U}_{i,j+\frac{1}{2}}^n - \mathfrak{U}_{i,j-\frac{1}{2}}^n}{\Delta y}\right] \\ &= \frac{1}{|\hat{\Omega}_{ij}|}\left[\left(\hat{\delta}y_{i+\frac{1}{2},j}V_{i+\frac{1}{2},j}^n - \hat{\delta}y_{i-\frac{1}{2},j}V_{i-\frac{1}{2},j}^n\right) - \left(\hat{\delta}x_{i,j+\frac{1}{2}}U_{i,j+\frac{1}{2}}^n - \hat{\delta}x_{i,j-\frac{1}{2}}U_{i,j-\frac{1}{2}}^n\right)\right],\end{aligned} \tag{6.81}$$

for $i, j = -\nu + 1, \dots, N + \nu$. Then, we obtain the absolute vorticity on the A-grid as:

$$\xi_{ij}^n = f_{ij} + \zeta_{ij}^n, \tag{6.82}$$

for $i, j = -\nu + 1, \dots, N + \nu$. Using again that the relative vorticity is advected, we may use the PPM fluxes $\mathcal{F}_{i+\frac{1}{2},j}^{PPM,x}, \mathcal{F}_{i,j+\frac{1}{2}}^{PPM,y}$ (Equations (5.22) and (5.25)) to compute the relative vorticity fluxes at the edges. Then, its follows from Equation (5.26) that we need to compute the terms:

$$F_{i+\frac{1}{2},j}[\xi^n, u^n] = \frac{1}{2}\mathcal{A}_{i+\frac{1}{2},j}^x\left(\mathcal{F}_{i,j+\frac{1}{2}}^{PPM,x}[\xi^n, \tilde{c}^{x,n}] + \mathcal{F}_{i,j+\frac{1}{2}}^{PPM,x}[\xi^n + \mathbf{g}(\xi^n, \tilde{c}^{y,n}), \tilde{c}^{x,n}]\right), \tag{6.83}$$

$$G_{i,j+\frac{1}{2}}[\xi^n, v^n] = \frac{1}{2}\mathcal{A}_{i,j+\frac{1}{2}}^y\left(\mathcal{F}_{i,j+\frac{1}{2}}^{PPM,y}[\xi^n, \tilde{c}^{y,n}] + \mathcal{F}_{i,j+\frac{1}{2}}^{PPM,y}[\xi^n + \mathbf{f}(\xi^n, \tilde{c}^{x,n}), \tilde{c}^{y,n}]\right), \tag{6.84}$$

where $\mathcal{A}_{i+\frac{1}{2},j}^x$ and $\mathcal{A}_{i,j+\frac{1}{2}}^y$ are given by Equations (5.21) and (5.24), respectively. The inner operators $\mathbf{f}$ and $\mathbf{g}$ are given in (5.1) and the terms $\tilde{c}^{x,n}$ and $\tilde{c}^{y,n}$ are the time-averaged CFL numbers described in Section 5.3.3. When using the duo-grid, these fluxes are computed

twice. However, we do not employ flux averaging at the cube interfaces (Section 5.3.4) as we achieved better results without it.

Kinetic energy fluxes

To estimate the integrals in Equation (6.19), we assume again that:

$$\partial_t U + \partial_x (uU)(x, y_{j+\frac{1}{2}}, t) = 0, \tag{6.85}$$

and a similar equation is assumed to hold for v and V, and therefore the integrals of Equation (6.19) may estimated again using finite-volume fluxes. In this case, we are going to consider the PPM fluxes $\mathfrak{F}^{PPM,x}_{i+\frac{1}{2}, j+\frac{1}{2}}$ and $\mathfrak{F}^{PPM,y}_{i+\frac{1}{2}, j+\frac{1}{2}}$ (Equation (5.17)), and the kinetic energy on B-grid is given by:

$$K^n_{i+\frac{1}{2}, j+\frac{1}{2}} = \frac{\Delta t}{2} \left(\tilde{u}^{n+\frac{1}{2}}_{i+\frac{1}{2}, j+\frac{1}{2}} \mathfrak{F}^{PPM,x}_{i+\frac{1}{2}, j+\frac{1}{2}}[U^n, \tilde{u}^n] + \tilde{v}^{n+\frac{1}{2}}_{i+\frac{1}{2}, j+\frac{1}{2}} \mathfrak{F}^{PPM,y}_{i+\frac{1}{2}, j+\frac{1}{2}}[V^n, \tilde{v}^n] \right), \tag{6.86}$$

for $i, j = 0, \dots, N$.

The time-averaged B-grid winds $\tilde{u}^n$ and $\tilde{v}^n$ are computed using the DP1 scheme or the scheme DP2. In this step, we use the B-grid contravariant winds $u^{n+\frac{1}{2}}_{i+\frac{1}{2}, j+\frac{1}{2}}, v^{n+\frac{1}{2}}_{i+\frac{1}{2}, j+\frac{1}{2}}$ obtained in Equations (6.75) and (6.76). If we use the DP2 scheme, we need $u^n_{i+\frac{1}{2}, j+\frac{1}{2}}, v^n_{i+\frac{1}{2}, j+\frac{1}{2}}$, which may obtained just as Equations (6.75) and (6.76). We also utilize the D-grid covariant winds obtained at time level n.

Finally, on the cubed-sphere using the duo-grid, these PPM fluxes are computed twice. Therefore, we average them at the cube interfaces to obtain a unique value, as described in Section 5.3.4. Thus, we estimate the kinetic energy gradient using a centered difference:

$$\delta_i K^n_{i, j+\frac{1}{2}} = K^n_{i+\frac{1}{2}, j+\frac{1}{2}} - K^n_{i-\frac{1}{2}, j+\frac{1}{2}}, \quad i = 1, \dots, N, j = 0, \dots, N, \tag{6.87}$$

$$\delta_j K^n_{i+\frac{1}{2}, j} = K^n_{i+\frac{1}{2}, j+\frac{1}{2}} - K^n_{i+\frac{1}{2}, j-\frac{1}{2}}, \quad i = 0, \dots, N, j = 1, \dots, N. \tag{6.88}$$

Hence, we have completed the description of the D-grid wind update using Equations (6.17) and (6.18).

6.2.3 Divergence damping

The divergence on B-grid points may be computed using the D-grid contravariant winds, using centered finite-differences as:

$$
\begin{aligned}
D_{i+\frac{1}{2},j+\frac{1}{2}}[u^n,v^n] &= \frac{1}{\sqrt{\mathfrak{g}}_{i+\frac{1}{2},j+\frac{1}{2}}}\left[\frac{(\sqrt{\mathfrak{g}}\mathfrak{u})^n_{i+1,j+\frac{1}{2}} - (\sqrt{\mathfrak{g}}\mathfrak{u})^n_{i,j+\frac{1}{2}}}{\Delta x} + \frac{(\sqrt{\mathfrak{g}}\mathfrak{v})^n_{i+\frac{1}{2},j+1} - (\sqrt{\mathfrak{g}}\mathfrak{v})^n_{i+\frac{1}{2},j}}{\Delta y}\right] \\
&= \frac{1}{|\hat{\Omega}_{i+\frac{1}{2},j+\frac{1}{2}}|}\left[\hat{\delta}y_{i+1,j+\frac{1}{2}}u^n_{i+1,j+\frac{1}{2}}\sin\alpha_{i+1,j+\frac{1}{2}} - \hat{\delta}y_{i,j+\frac{1}{2}}u^n_{i,j+\frac{1}{2}}\sin\alpha_{i,j+\frac{1}{2}}\right. \\
&\quad\left. + \hat{\delta}x_{i+\frac{1}{2},j+1}v^n_{i+\frac{1}{2},j+1}\sin\alpha_{i+\frac{1}{2},j+1} - \hat{\delta}x_{i+\frac{1}{2},j}v^n_{i+\frac{1}{2},j}\sin\alpha_{i+\frac{1}{2},j}\right].
\end{aligned}
$$

This is straightforward from the definition of divergence in terms of a cubed-sphere mapping (Equation (5.2)). We may then compute the gradient of this divergence at the edge midpoints, namely:

$$
D^u_{i,j+\frac{1}{2}}[u^n,v^n] = \frac{D_{i+\frac{1}{2},j+\frac{1}{2}}[u^n,v^n] - D_{i-\frac{1}{2},j+\frac{1}{2}}[u^n,v^n]}{\hat{\delta}x_{i,j+\frac{1}{2}}}, \quad i=1,\dots,N, j=0,\dots,N, \tag{6.89}
$$

$$
D^v_{i+\frac{1}{2},j}[u^n,v^n] = \frac{D_{i+\frac{1}{2},j+\frac{1}{2}}[u^n,v^n] - D_{i+\frac{1}{2},j-\frac{1}{2}}[u^n,v^n]}{\hat{\delta}y_{i+\frac{1}{2},j}}, \quad i=0,\dots,N, j=1,\dots,N. \tag{6.90}
$$

Notice that we can apply the divergence operator $D_{i+\frac{1}{2},j+\frac{1}{2}}$ again to the inputs $D^u_{i,j+\frac{1}{2}}$ and $D^v_{i+\frac{1}{2},j}$, and we may repeat this procedure as many times as we want. This process results in divergence damping (dd) operator which is an explicit dissipation mechanism. Dissipation mechanisms are commonly used in dynamical cores to ensure numerical stability and avoid accumulation of energy at the smallest grid scale. There are many other ways of introducing dissipation besides divergence damping (see Jablonowski and Williamson (2011) for a review) For an comprehensive analysis of the divergence damping operator, refer to Whitehead et al. (2011).

In FV3, the number of times we apply the divergence operator is denoted by *nord* (Harris et al., 2021). We are considering a divergence damping coefficient:

$$
v_D = (d_4 \min_{i,j} |\Omega_{ij}|)^{nord+1}, \tag{6.91}
$$

where $d_4 \geq 0$ is a given constant. The obtained results of after applying the divergence operator *nord* times are multiplied by v_D and added in Equations (6.17) and (6.18).

We point out that there are other numerical dissipation mechanisms available in FV3, such as vorticity damping and frictional heating (Harris et al., 2021). We are considering only the divergence damping for simplicity. We also point out that monotonic scheme that we are using (MONO) has implicit diffusion.

6.3   Proposed modifications

As we have seen in Section 6.2.1, there are two options for using the upwind fluxes, which depend on the treatment of the metric term to be considered, namely, mt0 or mt1. FV3 utilizes mt0. We are going to use the same option, and the C-grid solver is not modified. This choice is made because no significant improvements were observed when using mt1. Therefore, we are going to propose modifications only to the D-grid scheme.

Notice that, for the D-grid solver, the kinetic energy flux computed in Equation (6.86) using PPM may be computed using DP1 or DP2. Also, observe that this term does not have the metric term $\sqrt{\mathfrak{g}}$ to be considered. However, we find that using DP2 does not improve the results; actually, they get worse. We believe that the DP2 scheme does not improve the kinetic energy flux because the advection hypotheses assumed by FV3 for the wind components (Equation (6.85)) are not true. Therefore, improving the advection flux in this step does not necessarily improve the kinetic energy flux computation. Thus, we are going to use the DP1 scheme for the kinetic energy flux (Equation (6.86)).

There are two parts of the D-grid scheme where we may use the LT advection scheme from Chapter 5: the fluid depth update and the vorticity fluxes (Equations (6.83) and (6.84)). Currently, these parts are solved using the PL advection scheme, and we propose assessing the impact of using the LT advection scheme for both parts. The shallow-water solver that uses PL for these fluxes is referred to simply as the PL scheme, and when LT is used, we refer to it as the LT scheme.

6.4   Numerical experiments

In this Section, we are going to compare the PL and LT shallow-water schemes described (6.3) using classical shallow-water tests presented in the literature. These schemes shall be tested with the monotonic PPM scheme (MONO).

The D-grid output $(U_{i,j+\frac{1}{2}}^n, V_{i+\frac{1}{2},j}^n)$ is interpolated to the A-grid as

$$U_{ij}^n = \frac{(U_{i,j+\frac{1}{2}}^n + U_{i,j-\frac{1}{2}}^n)}{2}, \tag{6.92}$$

$$V_{ij}^n = \frac{(V_{i+\frac{1}{2},j}^n + V_{i-\frac{1}{2},j}^n)}{2}, \tag{6.93}$$

and then converted to latitude-longitude winds using Equations (4.47) and (4.41), where we obtain $(u_\lambda)_{ij}^n$, $(v_\phi)_{ij}^n$ on the A-grid. This facilitates plotting and reference solution calculation. We point out that this does not impact the error check, provided our schemes are at best-case second-order, as this averaging is second-order accurate and the wind conversion is an exact transformation.

Then, the errors are computed using a reference solution on the A-grid, denoted by $(h^{REF})^n$, $(u_\lambda^{REF})^n$ and $(v_\phi^{REF})^n$. The reference solution shall be assumed to be computed exactly or we follow the approach of Peixoto (2016) and use the ENDGame (Even Newer Dynamics for General atmospheric modelling of the environment) shallow-water solver developed

by Thuburn et al. (2010), which is the current UK MetOffice operational dynamical core (https://www.metoffice.gov.uk/research/foundation/dynamics/endgame, last accessed on April 3rd, 2024). This model is semi-Lagrangian and semi-implicit on a latitude-longitude grid. As suggested by Peixoto (2016), we use a grid size of 2048×1024 with a time step of 50 seconds whenever we employ it. This grid has a 20km resolution at the equator. The outputs of ENDGame are interpolated to the cubed-sphere A-grid from the latitude-longitude grid using cubic interpolation, giving us the reference solutions $(h^{REF})^n$, $(u_\lambda^{REF})^n$, and $(v_\phi^{REF})^n$. All the test cases (TCs) that we are going to consider are presented in Table 6.1.

To compute the error convergence, we consider cubed-sphere grids with values of $N_k = 48 \times 2^k$, and $dt\_atmos^{(k)} = \frac{dt\_atmos^{(0)}}{2^k}$ for $k = 0, \ldots, 4$, where the value of $dt\_atmos^{(0)}$ is taken from Table 5.3. The value of n_split is the same regardless of the value of N_k. We are going to consider both the equi-edge grid (g0) and the equiangular grid (g2) with the spherical midpoint formulation, and it will always be made clear whether the divergence damping is being used or not (Section 6.2.3). Whenever we use divergence damping, we consider $d_4 = 0.12$ and $nord = 2$, as used in Mouallem et al. (2023). In this case, the dissipation is a fourth-order operator and therefore is more scale selective. Additionally, we shall use the **dg1** duo-grid interpolation method (Section 4.3.2) as in Mouallem et al. (2023) to keep our analysis consistent with their results.

Test case	Description	Reference	$dt\_atmos^{(0)}$	n_split	Total time
TC2	Geostrophic balance	Williamson et al. (1992)	3600 s	7	5 $days$
TC5	Flow over a mountain	Williamson et al. (1992)	1800 s	7	15 $days$
TC6	Rossby-Haurwitz wave	Williamson et al. (1992)	1200 s	1	100 $days$

Table 6.1: *The test case considered in the numerical experiments, including descriptions, references, initial atmospheric time step dt_atmos$^{(0)}$, the number of times that the horizontal dynamics are solved on each atmospheric time step (n_split) and the total time of integration.*

6.4.1 Global steady geostrophic flow

We will consider the global steady geostrophic flow test case from Williamson et al. (1992). This test initializes the depth using Equation (4.51) (Figure 4.9) and the winds using Equation (4.52). We set $\alpha = \frac{\pi}{4}$ so that the flow is oriented with the corners of a cube. In this setup, the Coriolis parameter (Equation (6.6)) is modified as

$$f = 2\Omega(-\cos\phi\cos\lambda\sin\alpha + \sin\phi\cos\alpha), \tag{6.94}$$

and therefore, the initial condition does not change over time, allowing us to compute the exact solution at any time instant. Additional information about the simulation is given in Table 6.1. This test is performed with and without divergence damping.
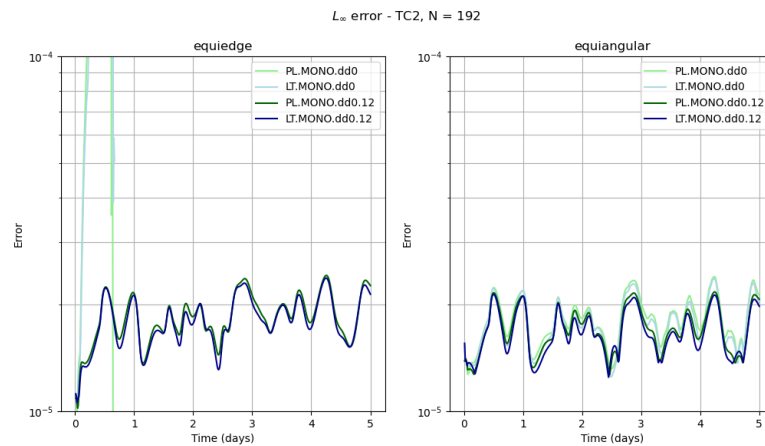

Figure 6.1: *Geostrophic balanced flow test: L_∞ relative error evolution for the fluid depth on the equi-edge grid (left) and on the equiangular grid (right) for 5 days and $N = 192$. Blue lines indicate the use of the LT scheme, while green lines represent the PL scheme. All schemes use the monotonic PPM (MONO). Light colors do not use divergence damping (dd), whereas dark color use divergence damping coefficient of 0.12.*
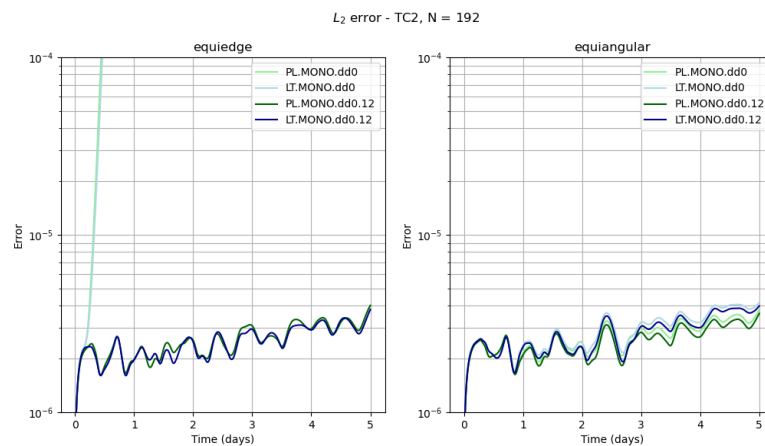

Figure 6.2: *As Figure 6.1 but using the L_2 norm.*

In Figure 6.1, we present the evolution of the L_∞ error in fluid depth over 5 days. We can observe that for the equi-edge grid, before day 1, both schemes LT and PL become numerically unstable when no divergence damping is employed. This behavior does not occur on the equiangular grid though. Both schemes are numerically stable on both grids when divergence damping is included. When the schemes are numerically stable, the errors of LT are only slightly smaller. On the other hand, Figure 6.2 shows that the L_2 error evolution of both schemes is similar for the equiangular grid, with PL being slightly smaller. On the equi-edge grid, the L_2 error of LT is slightly smaller.

(a) *PL scheme, max= 6.76×10^{-2}.*

(b) *LT scheme max= 6.40×10^{-2}.*

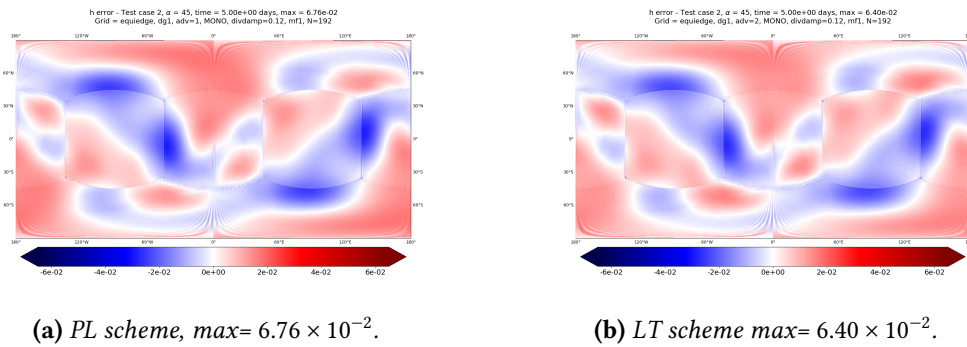**Figure 6.3:** *Geostrophic balanced flow test: depth error distribution after 5 days using the monotonic PPM (MONO) with PL (left) and LT schemes (right) on the equi-edge grid (g0) with N = 192. These results uses a divergence damping coefficient equal to 0.12.*
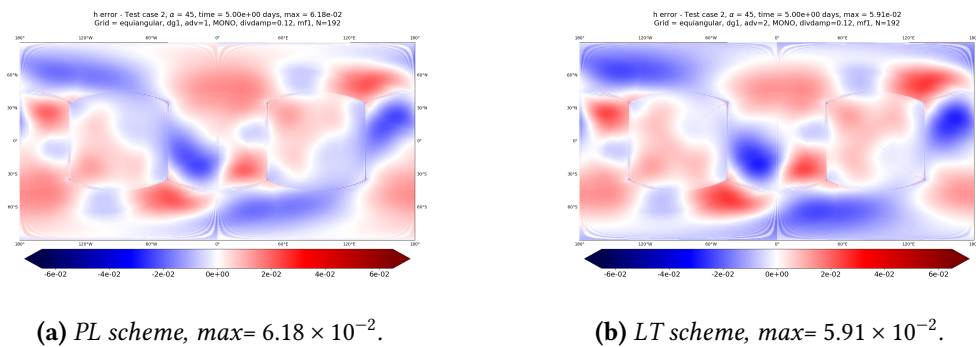

(a) *PL scheme, max= 6.18×10^{-2}.*

(b) *LT scheme, max= 5.91×10^{-2}.*

Figure 6.4: *As Figure 6.3 but using the equiangular grid (g2).*

(a) *PL scheme, max*$= 6.37 \times 10^{-2}$. **(b)** *LT scheme, max*$= 6.01 \times 10^{-2}$.
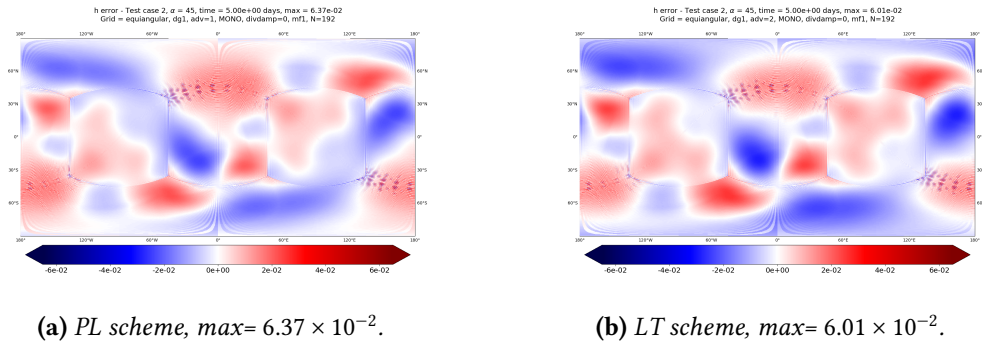
Figure 6.5: *As Figure 6.4 but using no divergence damping.*

In Figures 6.3 and 6.4 we show the final errors for both grids with divergence damping. The errors distribution is very similar for both schemes, and grid imprinting is still present, although the maximum errors, which occur at corners, are smaller for LT, indicating again that LT is slightly less sensitive to the corners. In Figure 6.5, we show the errors of PL and LT on the equiangular grid without divergence damping. Again, the errors of LT are slightly smaller. It is interesting to notice that in this case (no divergence damping), we can see some spurious waves being generated by the corners and affecting the solution at a cube edge, which does not occur when divergence damping is used. At last, the errors without divergence damping on the equi-edge grid are not shown because the schemes are numerically unstable in that case.

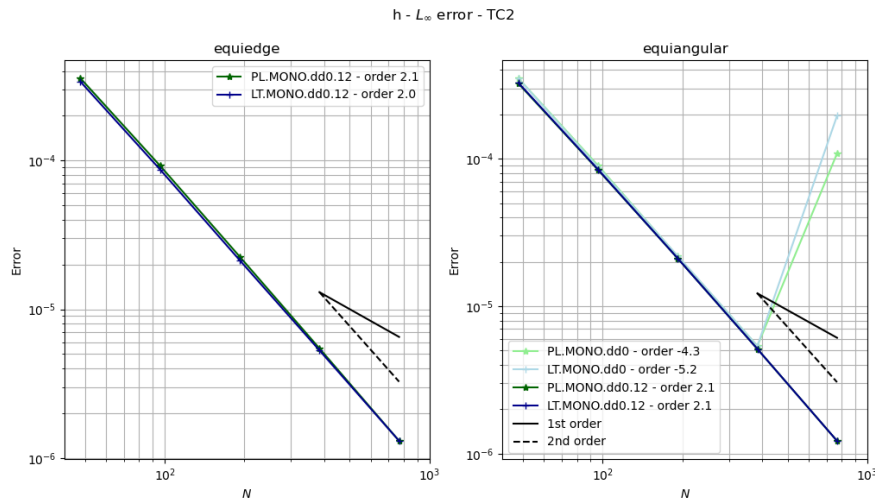**Figure 6.6:** L_∞ *error of the fluid depth component for the geostrophic flow test case after 5 days for the equi-edge grid (left) and the equiangular grid (right) considering the schemes LT (dark and light blue lines) and the PL (dark and light green lines). Light colors uses no divergence damping (dd) whereas dark colors use divergence damping.*

In Figure 6.6, we show the convergence of the error in the L_∞ for the fluid depth for all schemes and grids. The results on the equi-edge grid without divergence damping are not shown because the scheme becomes numerically unstable in this case. We can see that when using divergence damping, both the LT and PL schemes achieve second-order accuracy on both the equi-edge and equiangular grids. However, we can see that, when no divergence damping is employed, both schemes develop a very large error at a high resolution ($N = 768$). This result indicates that the equiangular grid may also develop numerical stability when there is divergence damping. However, this result shows that the equiangular grid is much more resilient to numerical instabilities than the equi-edge grid.

Finally, in Table 6.2, we show the times needed by each scheme for different grids and values of N. The results reflect the total execution time of the code on the ybytu machine from IME-USP (see Appendix B), with the total number of MPI processes fixed at 6 due to processor limitations. It is clear that the LT scheme adds a very small cost, showing that this scheme does not degrade the computational performance of the shallow-water solver.

N	Total runtime (seconds)			
	PL-g0	LT-g0	PL-g2	LT-g2
48	2.7762	2.9247	2.8379	2.8775
96	12.2006	13.0276	12.5094	13.1665
192	89.2555	95.9484	90.1434	96.4775
384	719.8602	759.0486	705.6963	743.4566
768	7308.6683	7529.9261	7190.8789	7418.2488

Table 6.2: *Total runtime for the LT and PL schemes using the equi-edge grid (g0) and the equiangular grid (g2) for different values of N. The additional cost of the LT scheme is due to the computation of a second-order estimate of 1D departure points. These results are obtained on the ybytu machine (refer to Appendix B), employing just one MPI process per cube face, resulting in a total of 6 MPI processes.*

6.4.2 Flow over a mountain

In this section, we present the flow over a mountain test case of Williamson et al. (1992). This test uses the same initial condition as the geostrophic balance test case, where h follows Equation (4.51) with $h_0 = 5960$ meters, and the winds follow Equation (4.52). The rotation parameter α is set to 0, so the initial wind is purely zonal. This test consider a bottom topography given by:

$$b(\lambda, \phi) = 2000 \left(1 - \frac{r}{r_0} \right), \tag{6.95}$$

where

$$r = \begin{cases} \sqrt{(\lambda - \lambda_0)^2 + (\phi - \phi_0)^2} & \text{if } \sqrt{(\lambda - \lambda_0)^2 + (\phi - \phi_0)^2} \leq r_0, \\ r_0 & \text{otherwise,} \end{cases} \tag{6.96}$$

$\lambda_0 = -\frac{\pi}{4}$, $\phi_0 = \frac{\pi}{6}$ and $r_0 = \frac{\pi}{9}$. These parameters define the mountain over a cube corner (Figure 6.7a).
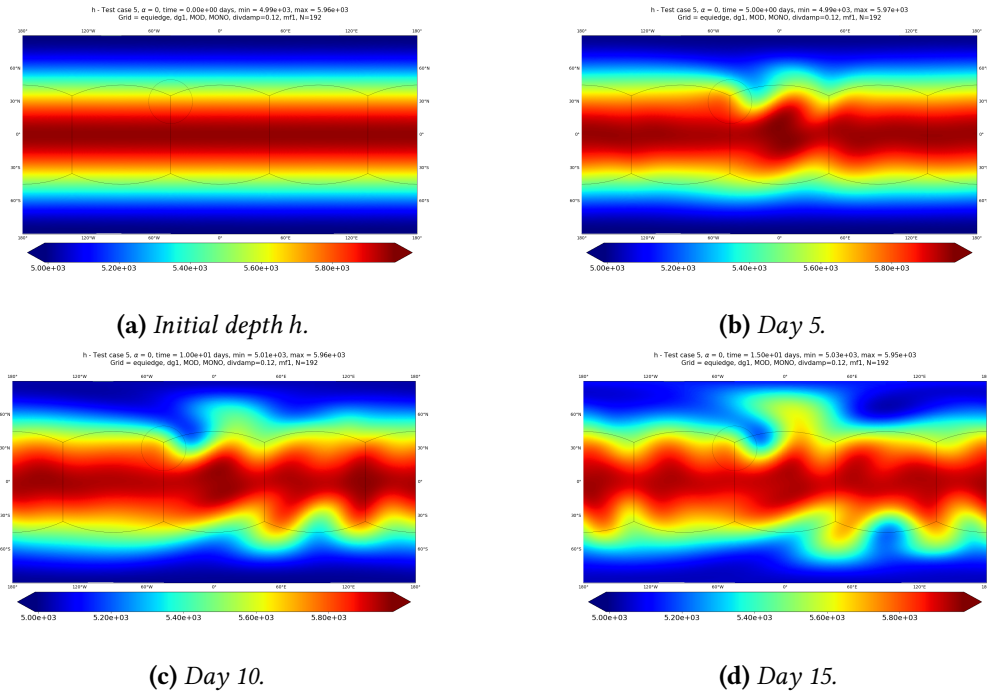

(a) *Initial depth h.*

(b) *Day 5.*

(c) *Day 10.*

(d) *Day 15.*

Figure 6.7: *Flow over a mountain: fluid depth initial condition (a) and after 5 (b), 10 (c), and 15 days (d). We are using the LT scheme on the equi-edge grid with N = 192 and a divergence damping coefficient equal to 0.12. The black circle shows the mountain's location.*

We ran this test for 15 days. In Figures 6.7, we show the evolution of fluid depth over time, specifically after 5, 10, and 15 days, using the LT scheme on an equi-edge grid with $N = 192$ and divergence damping. Figures 6.8 is similar but shows the meridional wind, illustrating the formation of a Rossby wave. In Figure 6.9 we show the wind divergence at days 10 and 15. It is clear from this last figure that we have a small amount of wind divergence, with a maximum absolute value equal to 6.7×10^{-6}, over to the mountain. Hence, we expect that LT and PL should yield similar results.

(a) *Initial meridional wind v_ϕ.*

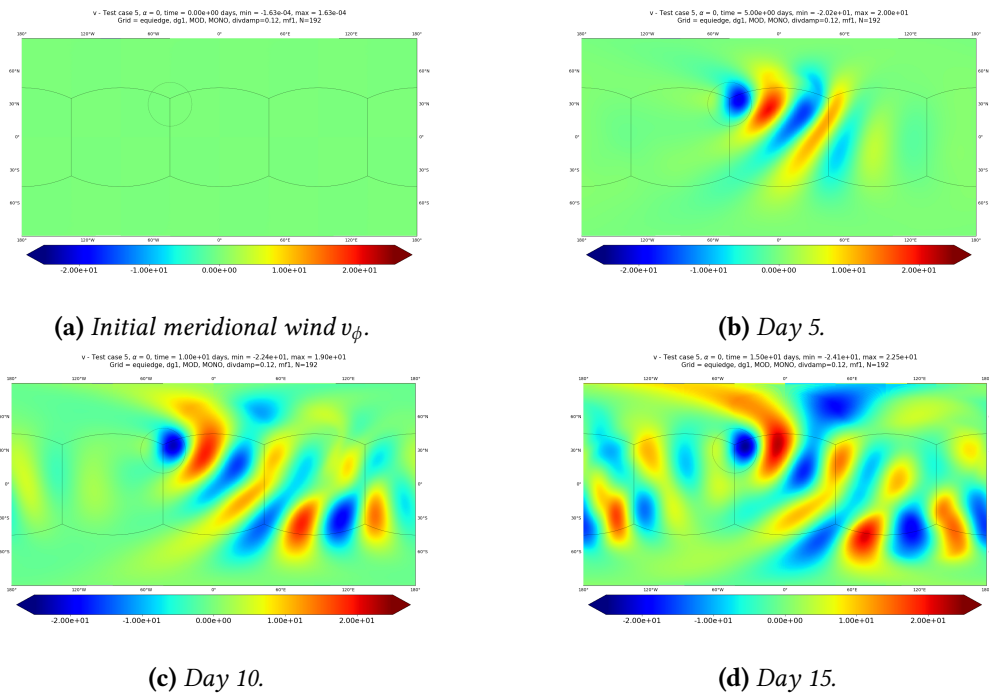**(b)** *Day 5.*

(c) *Day 10.*

(d) *Day 15.*

Figure 6.8: *Flow over a mountain: meridional wind initial condition (a) and after 5 (b), 10 (c), and 15 days (d). We are using the LT scheme on the equi-edge grid with $N = 192$ and a divergence damping coefficient equal to 0.12. The black circle shows the mountain's location.*

(a) *Day 10.*

(b) *Day 15.*
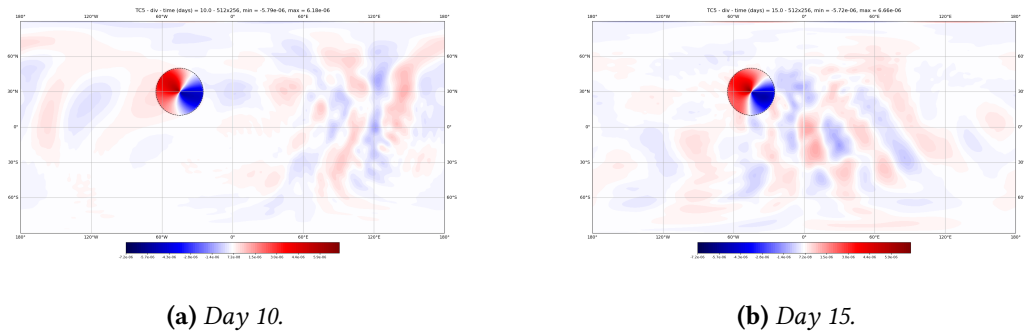
Figure 6.9: *Flow over a mountain: illustration of the wind divergence at days 10 and 5. The maximum absolute value of the divergence is 6.7×10^{-6}.*

To assess the accuracy of LT and PL schemes, we compute the errors using the ENDGame solution as a reference. We are going to consider simulations with and without divergence damping on the equi-edge and equiangular grids with $N = 192$. We shall investigate the error evolution over time. In fact, in Figures 6.10 and 6.11 we show the relative error evolution for the fluid depth on the L_∞ and L_2 norms. It is clear that both PL and LT yield very similar results. When divergence damping is employed, the errors do not grow very much. Otherwise, the errors may become much larger after day 12 at the equi-edge grid but not on the equiangular grid. This simulation agrees with the previous simulation (geostrophic balance flow), showing that the equiangular grid seems to be less susceptible to numerical instability.
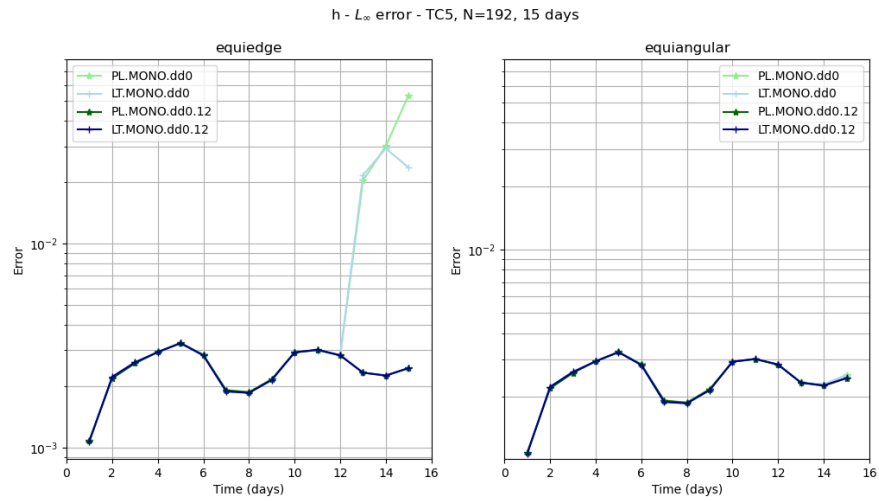
Figure 6.10: *Flow over a mountain test case: L_∞ relative error evolution for the fluid depth on the equi-edge grid (left) and on the equiangular grid (right) for 5 days and $N = 192$. Blue lines indicate the use of the LT scheme, while green lines represent the PL scheme. All schemes use the monotonic PPM (MONO). Light colors do not use divergence damping (dd), whereas dark color use divergence damping coefficient of 0.12. The error is computed with a spacing of 1 day using ENDGame as the reference solution.*
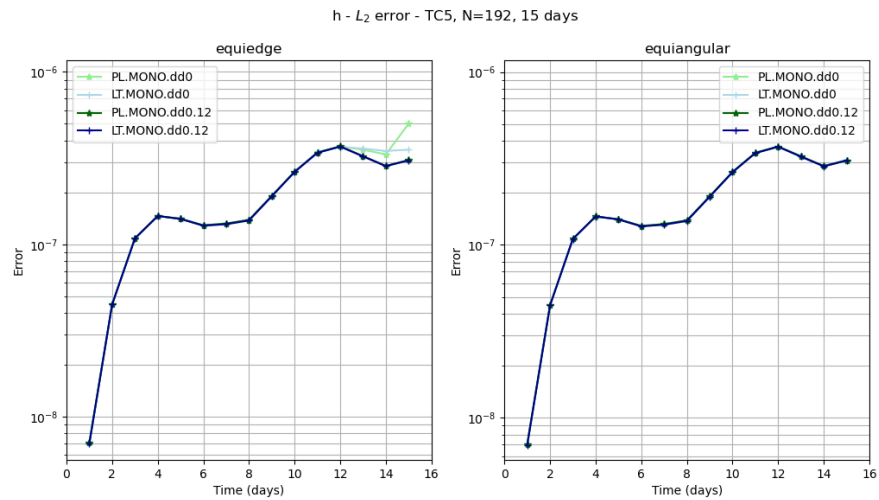
Figure 6.11: *As Figure 6.10 but using the L_2 norm.*

6.4.3 Rossby-Haurwitz wave

The test considered in this section is the Rossby-Haurwitz wave case, as suggested by Williamson et al. (1992). The initial conditions are exact solutions to the barotropic vorticity equation. Thus, the wind is divergence-free in this case. The expressions of the initial fields may be found in Williamson et al. (1992); we will omit them here. The solutions should propagate the wave to east and maintain its shape. Therefore, our goal in this test is to investigate the ability of the PL and LT schemes to preserve the wave shape.
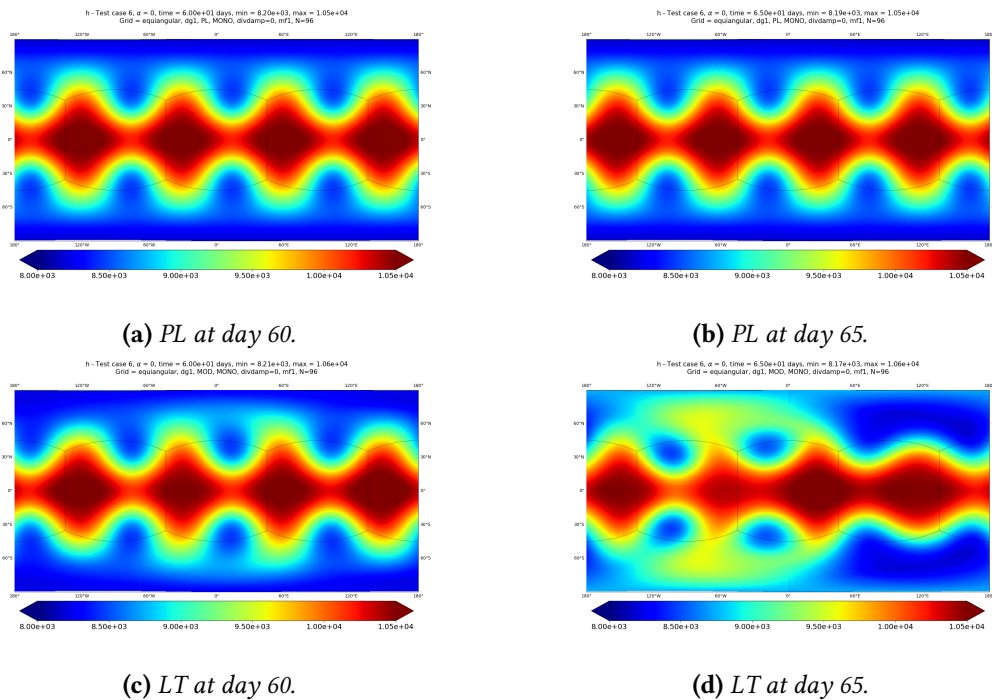

(a) *PL at day 60.*

(b) *PL at day 65.*

(c) *LT at day 60.*

(d) *LT at day 65.*

Figure 6.12: *Rossby-Haurwitz wave test case: Fluid depth on the equiangular grid, with N = 96 and without divergence damping. PL scheme results are shown in (a) and (b) on days 60 and 65, respectively. LT scheme results are shown in (c) and (d) on days 60 and 65, respectively.*

We ran this test for 100 days on the equi-edge and equiangular grids with $N = 96$. When we did not employ divergence damping, the results of the equi-edge grid became unstable for both schemes before day 60. However, the equiangular grid could perform this test without divergence damping. We show the wave shapes at days 60 and 65 in both schemes in Figure 6.12, for the PL scheme (Figures 6.12a and 6.12b) and for the LT scheme (Figures 6.12c and 6.12d). It is clear that the LT scheme can maintain the wave for 60 days, but the wave breaks at day 65. However, PL can preserve the shape of the wave for more than 65 days. In fact, it maintains the wave for approximately 100 days, as reported by Mouallem et al. (2023).

When we use divergence damping, the equi-edge grid achieves numerical stability, and we depict the wave in Figure 6.13. We can see that the addition of divergence damping anticipates the wave break for the LT scheme, which occurs at 55 days, as shown in Figures 6.13c and 6.13d. The PL scheme, again, maintains the wave (Figures 6.13a and 6.13b). However, in this case, the wave breaks earlier as well, at approximately day 95 (not shown). Similar results are observed for the equiangular grid with divergence damping and we

will omit them here.

The PL ability to keep the wave for 100 days is remarkable. The Rossby-Haurwitz wave is known for being a dynamically unstable test case, as investigated by Thuburn and Li (2000). Although our scheme has reduced the wave shape, it is similar, for instance, to the spectral model breaking time, as reported by Thuburn and Li (2000), which is between 50 and 55 days.

(a) *PL at day 55.*

(b) *PL at day 60.*

(c) *LT at day 55.*

(d) *LT at day 60.*
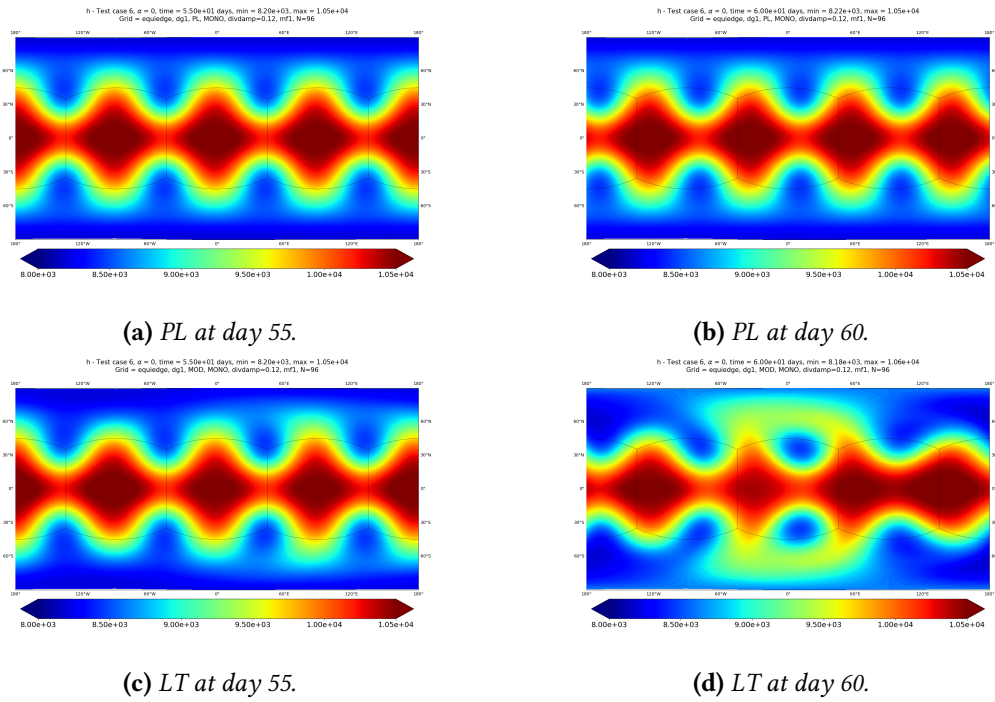
Figure 6.13: *Rossby-Haurwitz wave test case: Fluid depth on the equi-edge grid, with N = 96 and divergence damping. PL scheme results are shown in (a) and (b) on days 55 and 60, respectively. LT scheme results are shown in (c) and (d) on days 55 and 60, respectively.*

6.5   Conclusions

In this Chapter, we provided a detailed description of the FV3 shallow-water solver on the cubed-sphere. This solver begins with the shallow water equations written in the vector invariant form and considers their C and D-grid discretization. We presented a general discretization of the shallow water equations on the C and D-grids and showed that they can be solved using finite-volume fluxes. The C-grid solver is computed for a half time step, and the finite-volume fluxes use the classical upwind scheme, providing the winds centered at time. This part of the scheme is designed to be computationally cheap.

Following that, the D-grid solver can use the time-centered winds from the C-grid solver and apply the PPM fluxes to update the fluid depth, as well as absolute vorticity and kinetic energy fluxes. Then we discussed how our advection scheme from Chapter 5 can be used in the shallow-water solver. Since the C-grid is supposed to be computationally

inexpensive and our scheme uses PPM, we decided to leave it unchanged. We consider our new finite-volume advection scheme to compute the fluid depth and the absolute vorticity fluxes on the D-grid.

We then ran the geostrophic balance flow test case for 5 days with and without divergence damping. We observed that both advection schemes become numerically unstable for the equi-edge grid when no divergence damping is used, which is not observed on the equiangular grid. With divergence damping, both schemes are numerically stable on both grids. Additionally, we noticed that our scheme helps to slightly reduce the L_∞ errors when the scheme is numerically stable. We analyzed the runtime and found that our advection scheme adds a very small additional computational burden.

We also analyzed the flow over a mountain test case and showed that our scheme yields very similar results to the current FV3 advection scheme. Since this test case does not have an exact solution known, we have used the semi-Lagrangian semi-implicit shallow-water model ENDGame, which is used in the current operational UK Met Office dynamical core, as the reference solution. Additionally, the equiangular grid managed to run this test without divergence damping, while the equi-edge grid required divergence damping to run this test without creating large errors.

The Rossby-Haurwitz wave test case was also investigated. The goal of this test is to assess the ability of a scheme to propagate the wave to the east and maintain the shape of the wave. However, this test is dynamically unstable and the wave eventually breaks. We showed that our scheme keeps the wave for 55-60 days, while the FV3 scheme keeps the wave for 95-100 days. Although we have reduced the wave breaking time, we pointed out that it is a little better than, for instance, the spectral model.

The goal of this chapter was to describe the shallow-water solver, demonstrate how our scheme may be used and compare the computational performance. However, a more comprehensive analysis of the shallow-water model using our scheme is certainly needed. We point out that most of the classical shallow-water tests available in the literature (Galewsky et al., 2004; Williamson et al., 1992) have no or small divergence. Therefore, we expect that our scheme would introduce similar results in these tests because, as we have seen in Chapter 5, our advection is slightly better for divergence-free winds and significantly better only for divergent winds.

Chapter 7

Conclusions

The FV3 dynamical core has become very popular in the atmospheric modeling community. It has received more attention, especially after being adopted as the new Global Forecast System (GFS) of the USA. The objective of this thesis is to investigate all the details of the FV3 advection scheme, as the horizontal dynamics of FV3 are solved using only advection finite-volume fluxes, thus playing a key role in FV3. The major outcome of this thesis is a more accurate 2D advection scheme than the current FV3 advection scheme, as demonstrated in numerous numerical simulations.

The motivation for the new advection scheme method started in Chapter 2, where we provide a proof that the time-averaged fluxes of 1D finite-volume methods for the 1D advection equation require two tasks: a departure point calculation and the reconstruction of the solution using the average values. The average values are reconstructed using PPM, just as in FV3. We note that the FV3 scheme uses a first-order departure point scheme. We demonstrate how we can compute the departure point using a second-order Runge-Kutta scheme, which provides us with second-order accuracy. Then, we observed in numerical tests that this scheme improves accuracy significantly, even with monotonicity constraints, with only minor extra computational efforts. Namely, we only need to perform one linear interpolation for each edge per time step. We could only observe this improvement because we considered a variable wind test for the 1D linear advection equation. Most 1D tests in the literature use a constant wind, and therefore a first-order departure point is exact. This oversimplifies matters, as the departure point issue does not arise.

Next, we moved to the 2D advection equation on the plane in Chapter 3. The 2D advection scheme of FV3 consists of combining 1D flux PPM operators. This combination is made in such a way that when the scalar field is constant and the wind is divergence-free, the scheme is exact. We observed that this scheme is second-order for divergence-free winds; however, in a numerical simulation for a divergent wind, we showed that the FV3 scheme is only first-order. We then demonstrated how we may modify the FV3 scheme to achieve second-order accuracy for both divergence-free and divergent winds. This modification involves a slight change of the inner 1D flux operators, as well as the incorporation of the second-order departure point scheme outlined in Chapter 2. Although we lost the preservation of a constant scalar field, the error is only second-order accurate for this case. We show that the new scheme has slightly improved performance for divergence-

free winds, but for divergent winds, the results are significantly better. Then, we proposed a scheme that is second-order in general, whereas the FV3 scheme is second-order only for divergence-free winds.

Following that, our next objective was to study the advection equation on the sphere. We provided all the tools needed in Chapter 4, where we presented the duo-grid, which consists of extending the gridlines of the cubed sphere mapping. We may then use 1D Lagrangian interpolation to fill the ghost cell values. In this Chapter, we also introduce the equiangular cubed sphere, which is the most uniform cubed-sphere available in the literature. We also introduce the equi-edge grid, which is less uniform than the equiangular grid in general but offers more uniformity near to the cube edges, aiming to avoid grid imprinting in these regions. Additionally, we show that Lagrange interpolation based on geodesic distances is much less accurate than using Lagrange interpolation based on local cubed sphere mapping coordinates.

Using all these tools, we could solve the advection equation on the sphere in Chapter 5. We presented the advection scheme of FV3 and introduced our new scheme. We observed that our new scheme requires a different treatment of the metric terms in the flux 1D computation. Essentially, we extended the results from the plane to the cubed-sphere. Our scheme is second-order accurate on the cubed-sphere, while the FV3 scheme is second-order only for divergence-free winds and first-order for divergent winds. This was observed on both equi-edge and equiangular grids. Additionally, the equi-edge grids yielded smaller errors. Furthermore, the new scheme was slightly less sensitive to the corners, as it did not show slightly smaller errors at these locations. This was observed in tests where we evaluated a Gaussian hill, cosine bell, and a cylinder over the corners.

As an application of our new scheme, in Chapter 6, we provide a comprehensive description of the FV3 shallow-water solver. This solver utilizes the shallow-water equations in vector form and employs only advective finite volume fluxes to compute the fluid depth, kinetic energy, and absolute vorticity fluxes. We consider our scheme only to solve the continuity equation and the absolute vorticity fluxes. We observe that for the geostrophic balanced flow test case, our scheme helps to reduce the maximum error slightly. Additionally, we analyze the total runtime for each method at different grid resolutions, and we conclude that our scheme adds a very small extra cost. Another major conclusion of this chapter is that the equiangular grid is less susceptible to numerical instability, being able to perform better without the requirement of divergence damping. This was observed for both advection schemes. Finally, for the Rossby-Haurwitz wave test, the new scheme maintains the wave shape for 55-60 days, which is similar to the spectral model. The FV3 scheme, on the other hand, preserves the wave for approximately 95-100 days.

7.1   Future work ideas

A possible extension of this work would be to consider the non-hydrostatic solver of FV3 to assess the new advection scheme's ability in three dimensions and conduct more realistic simulations, as we have only analyzed idealized test cases in this work. Furthermore, we could also investigate the new scheme's performance on the stretched grids available in FV3.

Our modified advection scheme is slightly better for divergence-free flows in some situations, but for divergent winds, our new scheme is significantly better. We expect that our scheme would perform similarly to the current FV3 method in processes where there is small divergence/convergence of the wind, and it should yield better results for processes where divergence plays a key role.

Horizontal wind divergence plays a pivotal role in many phenomena on the atmosphere such as in tropical cyclones, hurricanes and in the Intertropical Convergence Zone (Holton & Hakim, 2012). For example, hurricanes are fueled by strong horizontal wind convergence at the Earth's surface, with strong horizontal wind divergence occurring at high altitudes. We could perform a study based on Gao et al. (2021), where the authors investigated the impact of using different PPM schemes of FV3 on hurricane intensity prediction. Their study highlights how modifying the advection scheme may improve hurricane intensity prediction and affect the eyewall convection location. Therefore, we could use our advection scheme in these simulations and observe its effect on hurricane intensity prediction.

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*

$$\frac{d}{ds}\int_{s_0}^s f(s, \theta)\,d\theta = f(s, s) + \int_{s_0}^s \partial_s f(s, \theta)\,d\theta.$$

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 $h_n \xrightarrow{n \to \infty} 0$. Then

$$\frac{F(s + h_n) - F(s)}{h_n} = \frac{1}{h_n}\int_{s_0}^{s+h_n} f(s + h_n, \theta)\,d\theta - \frac{1}{h_n}\int_{s_0}^s f(s, \theta)\,d\theta \tag{A.2}$$

$$= \frac{1}{h_n}\left(\int_s^{s+h_n} f(s + h_n, \theta)\,d\theta + \int_{s_0}^s f(s + h_n, \theta)\,d\theta - \int_{s_0}^s f(s, \theta)\,d\theta\right). \tag{A.3}$$

It follows from Theorem A.2 (with $g = 1$) that there exists θ_n between s and $s + h_n$ such that:

$$\frac{1}{h_n}\int_s^{s+h_n} f(s + h_n, \theta)\,d\theta = f(s + h_n, \theta_n) \xrightarrow{n \to \infty} f(s, s), \tag{A.4}$$

since $\theta_n \xrightarrow{n \to \infty} s$. From the mean value theorem, there exists s_n between s and $s + h_n$ such that:

$$\int_{s_0}^s \left(\frac{f(s + h_n, \theta) - f(s, \theta)}{h}\right) d\theta = \int_{s_0}^s \partial_s f(s_n, \theta)\,d\theta \xrightarrow{n \to \infty} \int_{s_0}^s \partial_s f(s, \theta)\,d\theta, \tag{A.5}$$

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:

$$\frac{1}{\Delta x} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} f(x)\,dx - f(x_i) = \frac{1}{\Delta x} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \left(f'(x_i)(x - x_i) + f''(\xi)\frac{(x - x_i)^2}{2} \right) dx$$

$$= \frac{1}{\Delta x} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} f''(\xi)\frac{(x - x_i)^2}{2}\,dx.$$

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

$$C_1 = \frac{1}{24}f''(\eta_i). \tag{A.8}$$

$\square$

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 constant 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:

$$\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}} f(x,y)\,dx\,dy = \Delta x \Delta y f(x_i,y_j) + \frac{\Delta x^2}{24}\Delta x \Delta y \partial_x^2 f(v_i,y_j) + \frac{\Delta y^2}{24}\Delta x \Delta y \partial_y^2 f(\theta_i,\eta_j),$$

where $v_i, \theta_i \in [x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]$, recalling that $\Delta x = \Delta y$. $\qquad\square$

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 $[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}] \times [y_{j-\frac{1}{2}}, y_{j+\frac{1}{2}}]$, $i, j = 1, \ldots, N$, with lengths $\Delta x = \Delta y$, we have*

$$\int_a^b \int_c^d f(x,y)\,dx\,dy - \sum_{i,j=1}^{N} f(x_i,y_j)\Delta x \Delta y = C_1 \Delta x^2, \qquad (A.10)$$

where C_1 depends only on f.

Proof. Using Theorem A.5, we have:

$$\frac{1}{\Delta x \Delta y}\int_a^b \int_c^d f(x,y)\,dx\,dy = \frac{1}{\Delta x \Delta y}\sum_{i,j=1}^{N}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}} f(x,y)\,dx\,dy$$

$$= \sum_{i,j=1}^{N} f(x_i,y_j) + \frac{\Delta x^2}{24}\sum_{i,j=1}^{N}\left(\partial_x^2 f(v_i,y_j) + \partial_y^2 f(\theta_i,\eta_j)\right).$$

We notice that

$$\Delta x \Delta y \sum_{i,j=1}^{N}\left(\partial_x^2 f(v_i,y_j) + \partial_y^2 f(\theta_i,\eta_j)\right) = \frac{(b-a)(d-c)}{N^2}\sum_{i,j=1}^{N}\left(\partial_x^2 f(v_i,y_j) + \partial_y^2 f(\theta_i,\eta_j)\right),$$

and we also point that from the inequality

$$\min_{x,u\in[a,b],y,v\in[c,d]} \left(\partial_x^2 f(x,y) + \partial_y^2 f(u,v) \right) \leq \frac{1}{N^2} \sum_{i,j=1}^{N} \left(\partial_x^2 f(v_i,y_j) + \partial_y^2 f(\theta_i,\eta_j) \right)$$

$$\leq \max_{x,u\in[a,b],y,v\in[c,d]} \left(\partial_x^2 f(x,y) + \partial_y^2 f(u,v) \right),$$

and with the aid of the intermediate value theorem, we have

$$\frac{1}{N^2} \sum_{i,j=1}^{N} \left(\partial_x^2 f(v_i,y_j) + \partial_y^2 f(\theta_i,\eta_j) \right) = \partial_x^2 f(\overline{x},\overline{y}) + \partial_y^2 f(\overline{u},\overline{v}),$$

for some $(\overline{x},\overline{y}) \in [a,b] \times [c,d]$, $(\overline{u},\overline{v}) \in [a,b] \times [c,d]$, from which the claim follows. $\qquad\square$

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):

$$Q_i(t^{n+1}) = Q_i(t^n) - \lambda \left(F_{i+\frac{1}{2}}^n(Q(t^n), \tilde{u}_{i+\frac{1}{2}}^n) - F_{i-\frac{1}{2}}^n(Q(t^n), \tilde{u}_{i-\frac{1}{2}}^n) \right) + \Delta t \tau_i^n. \qquad \text{(A.11)}$$

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 (2.22). Since $Q_i(t^n)$ is the exact solution of Equation (2.10), the LTE may be rewritten as

$$\tau_i^n = \frac{1}{\Delta x} \left[\left(\frac{1}{\Delta t} \int_{t^n}^{t^{n+1}} (uq)(x_{i+\frac{1}{2}}, t)\, dt - F_{i+\frac{1}{2}}^n(Q(t^n), \tilde{u}_{i+\frac{1}{2}}^n) \right) + \left(\frac{1}{\Delta t} \int_{t^n}^{t^{n+1}} (uq)(x_{i-\frac{1}{2}}, t)\, dt - F_{i-\frac{1}{2}}^n(Q(t^n), \tilde{u}_{i-\frac{1}{2}}^n) \right) \right]. \qquad \text{(A.12)}$$

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:*

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

and it is said to be consistent with order P in the p−norm if

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

From Equation (A.12), it follows that we basically need to ensure that the numerical flux function $\mathcal{F}_{i+\frac{1}{2}}^n$ 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:

$$E_i^n = Q_i(t^n) - Q_i^n, \quad i = 1, \dots, N,$$

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:*

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

and it is said to converge with order P in the p−norm if

$$\max_{1 \le n \le N_T^{(k)}} \|E^n\|_{p, \Delta x^{(k)}} = \mathcal{O}(\Delta x^P).$$

Subtracting Equation (2.22) from Equation (A.11) we get the following equation for the error:

$$
\begin{aligned}
E_i^{n+1} = E_i^n - \lambda & \left[\left(F_{i+\frac{1}{2}}^n(Q(t^n), \tilde{u}_{i+\frac{1}{2}}^n) - F_{i+\frac{1}{2}}^n(Q^n, \tilde{u}_{i+\frac{1}{2}}^n) \right) \right. \\
& \left. - \left(F_{i-\frac{1}{2}}^n(Q(t^n), \tilde{u}_{i-\frac{1}{2}}^n) - F_{i-\frac{1}{2}}^n(Q^n, \tilde{u}_{i-\frac{1}{2}}^n) \right) \right] + \tau_i^n \Delta t.
\end{aligned}
\tag{A.13}
$$

Notice that if $q, u \in C^3$, we can rewrite Equation (A.12) as:

$$
\tau_i^n = \left[\frac{1}{\Delta x \Delta t} \int_{t^n}^{t^{n+1}} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \frac{\partial(uq)}{\partial x}(x,t)\, dx\, dt - \left(\frac{F_{i+\frac{1}{2}}^n(Q(t^n), \tilde{u}_{i-\frac{1}{2}}^n) - F_{i-\frac{1}{2}}^n(Q(t^n), \tilde{u}_{i-\frac{1}{2}}^n)}{\Delta x} \right) \right].
$$

Using the midpoint rule for integration (Theorem A.4) and the mean value theorem for integrals (Theorem A.2), we have:

$$
\begin{aligned}
\tau_i^n &= \left[\frac{1}{\Delta t} \int_{t^n}^{t^{n+1}} \left(\frac{\partial(uq)}{\partial x}(x_i, t) + \frac{\Delta x^2}{24} \frac{\partial(uq)}{\partial x}(\xi, t) \right) dt - \left(\frac{F_{i+\frac{1}{2}}^n(Q(t^n), \tilde{u}_{i+\frac{1}{2}}^n) - F_{i-\frac{1}{2}}^n(Q(t^n), \tilde{u}_{i-\frac{1}{2}}^n)}{\Delta x} \right) \right] \\
&= \left[\frac{1}{\Delta t} \int_{t^n}^{t^{n+1}} \frac{\partial(uq)}{\partial x}(x_i, t)\, dt - \left(\frac{F_{i+\frac{1}{2}}^n(Q(t^n), \tilde{u}_{i+\frac{1}{2}}^n) - F_{i-\frac{1}{2}}^n(Q(t^n), \tilde{u}_{i-\frac{1}{2}}^n)}{\Delta x} \right) \right] + \frac{\Delta x^2}{24} \frac{\partial^3(uq)}{\partial x^3}(\xi, \bar{t}),
\end{aligned}
\tag{A.14}
$$

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

$$\frac{F_{i+\frac{1}{2}}^n(Q(t^n), \tilde{u}_{i+\frac{1}{2}}^n)) - F_{i-\frac{1}{2}}^n(Q(t^n), \tilde{u}_{i-\frac{1}{2}}^n)}{\Delta x}.$$

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 $\mathcal{H}_{\Delta x,n} : \mathbb{P}_\nu^N \to \mathbb{P}_\nu^N$ whose i-th entry is given by:

$$[\mathcal{H}_{\Delta x,n}(Q)]_i = Q_i - \lambda \left(F_{i+\frac{1}{2}}^n(Q, \tilde{u}_{i+\frac{1}{2}}^n) - F_{i-\frac{1}{2}}^n(Q, \tilde{u}_{i-\frac{1}{2}}^n) \right), \tag{A.15}$$

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 (2.22) 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:*

$$\|\mathcal{H}_{\Delta x,n}(Q) - \mathcal{H}_{\Delta x,n}(P)\|_{p,\Delta x} \le (1 + \alpha \Delta t)\|Q - P\|_{p,\Delta x}, \tag{A.17}$$

for all $Q, P \in \mathbb{R}_\nu^N$ 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 $\mathbb{C}_v^N$ similarly as $\mathbb{P}_v^N$. The Fourier modes $e^{\imath k\theta} \in \mathbb{C}_v^N$ for $k = 1, \dots, N$, have entries 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 $\mathbb{C}_v^N$ with respect to the inner product

$$
\langle Q, P \rangle = \frac{1}{N} \sum_{i=1}^{N} Q_i \overline{P_i},
$$

for $P, Q \in \mathbb{C}_v^N$ and $\bar{z}$ denotes the complex conjugate of z. Given $Q \in \mathbb{P}_v^N$, 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:

$$\mathcal{H}_{\Delta x,n}(e^{\iota k\theta}) = \rho(k)e^{\iota k\theta},$$

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 any PPM scheme 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 the unlimited PPM scheme (UNLIM). We can observe that UNLIM damp most of the Fourier modes for larger k, regardless of the CFL number. We point out that UNLIM is exact when the CFL number is equal to 1. From this analysis, we can conclude that UNLIM 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,
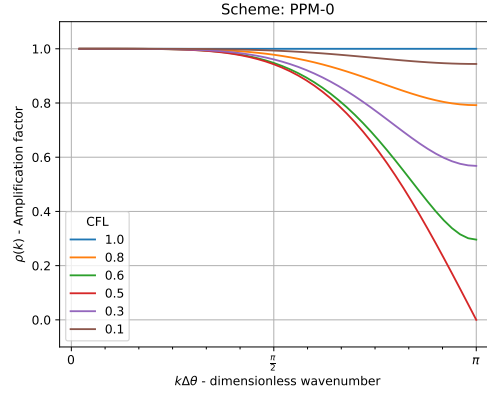
Figure A.1: *Amplification factor for the UNLIM scheme for different CFL numbers.*

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:

$$[\mathcal{H}_{\Delta x,\Delta y,n}(Q)]_{ij} = Q_{ij} - \Delta t \mathbb{D}_{ij}^{n}$$

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*

$$\|\mathcal{H}_{\Delta x,\Delta y,n}(Q) - \mathcal{H}_{\Delta x,\Delta y,n}(P)\|_{p,\Delta x\times\Delta y} \le (1 + \alpha\Delta t)\|Q - P\|_{p,\Delta x\times\Delta y}, \tag{A.20}$$

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:

$$\|E^{n+1}\|_{p,\Delta x\times\Delta y} \le e^{\alpha T}(\|E^0\|_{p,\Delta x\times\Delta y} + T \max_{n=1,\dots,N_T} \|\tau^n\|_{p,\Delta x\times\Delta y}).$$

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 con-

sidering 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:

$$\frac{F(x_0 + h) - F(x_0 - h)}{2h} = F'(x_0) + F^{(3)}(x_0)\frac{h^2}{3!} + \left(F^{(5)}(\theta_h) + F^{5)}(\theta_{-h})\right)\frac{h^4}{2 \cdot 5!}, \quad \theta_h \in [x_0, x_0+h], \quad \theta_{-h} \in [x_0-h, x]$$

(A.24)

and

$$\frac{F(x_0 + 2h) - F(x_0 - 2h)}{4h} = F'(x_0) + F^{(3)}(x_0)\frac{4h^2}{3!} + \left(F^{(5)}(\theta_{2h}) + F^{(5)}(\theta_{-2h})\right)\frac{16h^4}{2 \cdot 5!}, \quad \text{(A.25)}$$

$$\theta_{2h} \in [x_0, x_0 + 2h], \quad \theta_{-2h} \in [x_0 - 2h, x_0].$$

Using Equations (A.24) and (A.25), we obtain:

$$\frac{4}{3}\left(\frac{F(x_0 + h) - F(x_0 - h)}{2h}\right) = \frac{4}{3}F'(x_0) + F^{(3)}(x_0)\frac{4h^2}{3 \cdot 3!} + \left(F^{(5)}(\theta_h) + F^{(5)}(\theta_{-h})\right)\frac{h^4}{2 \cdot 5!},$$

(A.26)

$$\frac{1}{3}\left(\frac{F(x_0 + 2h) - F(x_0 - 2h)}{4h}\right) = \frac{1}{3}F'(x_0) + F^{(3)}(x_0)\frac{4h^2}{3 \cdot 3!} + \left(F^{(5)}(\theta_{2h}) + F^{(5)}(\theta_{-2h})\right)\frac{16h^4}{3 \cdot 2 \cdot 5!}$$

(A.27)

Subtracting Equation (A.27) from Equation (A.26) we get the desired Equation (A.22) with

$$C_1 = \frac{1}{720}\left(3F^{(5)}(\theta_h) + 3F^{(5)}(\theta_{-h}) - 16F^{(5)}(\theta_{2h}) - 16F^{(5)}(\theta_{-2h})\right), \quad \text{(A.28)}$$

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

$$C_1 = \frac{1}{720}\left(6F^{(5)}(\eta_1) - 32F^{(5)}(\eta_2)\right), \quad \text{(A.29)}$$

where $\eta_1, \eta_2 \in [x_0 - 2h, x_0 + 2h]$, which concludes the proof. $\qquad\square$

Lemma A.2. *Let $F \in C^4(\mathbb{R})$, $x_0 \in \mathbb{R}$ and $h > 0$. Then, the following identity holds:*

$$F''(x_0) = \frac{-2F(x_0 - 2h) + 15F(x_0 - h) - 28F(x_0) + 20F(x_0 + h) - 6F(x_0 + 2h) + F(x_0 + 3h)}{6h^2} + C_2 h^2,$$

(A.30)

where C_2 is a constant that depends only on F and h.

Proof. From the Taylor's expansion, we have:

$$F(x_0 - 2h) = F(x_0) - 2F'(x_0)h + 2F^{(2)}(x_0)h^2 - \frac{8}{6}F^{(3)}(x_0)h^3 + \frac{16}{24}F^{(4)}(\theta_{-2h})h^4,$$

$$F(x_0 - h) = F(x_0) - F'(x_0)h + \frac{1}{2}F^{(2)}(x_0)h^2 - \frac{1}{6}F^{(3)}(x_0)h^3 + \frac{1}{24}F^{(4)}(\theta_{-h})h^4,$$

$$F(x_0 + h) = F(x_0) + F'(x_0)h + \frac{1}{2}F^{(2)}(x_0)h^2 + \frac{1}{6}F^{(3)}(x_0)h^3 + \frac{1}{24}F^{(4)}(\theta_h)h^4,$$

$$F(x_0 + 2h) = F(x_0) + 2F'(x_0)h + 2F^{(2)}(x_0)h^2 + \frac{8}{6}F^{(3)}(x_0)h^3 + \frac{16}{24}F^{(4)}(\theta_{2h})h^4,$$

$$F(x_0 + 3h) = F(x_0) + 3F'(x_0)h + \frac{9}{2}F^{(2)}(x_0)h^2 + \frac{27}{6}F^{(3)}(x_0)h^3 + \frac{81}{24}F^{(4)}(\theta_{3h})h^4,$$

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:

$$-2F(x_0 - 2h) = -2F(x_0) + 4F'(x_0)h - 4F^{(2)}(x_0)h^2 + \frac{16}{6}F^{(3)}(x_0)h^3 - \frac{32}{24}F^{(4)}(\theta_{-2h})h^4,$$

$$15F(x_0 - h) = 15F(x_0) - 15F'(x_0)h + \frac{15}{2}F^{(2)}(x_0)h^2 - \frac{15}{6}F^{(3)}(x_0)h^3 + \frac{15}{24}F^{(4)}(\theta_{-h})h^4,$$

$$-28F(x_0) = -28F(x_0),$$

$$20F(x_0 + h) = 20F(x_0) + 20F'(x_0)h + 10F^{(2)}(x_0)h^2 + \frac{20}{6}F^{(3)}(x_0)h^3 + \frac{20}{24}F^{(4)}(\theta_h)h^4,$$

$$-6F(x_0 + 2h) = -6F(x_0) - 12F'(x_0)h - 12F^{(2)}(x_0)h^2 - 8F^{(3)}(x_0)h^3 - \frac{96}{24}F^{(4)}(\theta_{2h})h^4,$$

$$F(x_0 + 3h) = F(x_0) + 3F'(x_0)h + \frac{9}{2}F^{(2)}(x_0)h^2 + \frac{27}{6}F^{(3)}(x_0)h^3 + \frac{81}{24}F^{(4)}(\theta_{3h})h^4.$$

Summing all these equations, we get the desired Formula (A.30) with C_2 given by:

$$C_2 = \frac{1}{24}\left(32F^{(4)}(\theta_{-2h}) - 15F^{(4)}(\theta_{-h}) - 20F^{(4)}(\theta_h) + 96F^{(4)}(\theta_{2h}) - 81F^{(4)}(\theta_{3h}) \right). \quad \text{(A.31)}$$

Using the intermediate value theorem, we can express C_2 in a more compact way as

$$C_2 = \frac{1}{24}\left(128F^{(5)}(\eta_1) - 116F^{(5)}(\eta_2) \right), \quad \text{(A.32)}$$

where $\eta_1, \eta_2 \in [x_0 - 2h, x_0 + 3h]$, which concludes the proof. $\qquad \square$

Lemma A.3. *Let* $F \in C^4(\mathbb{R})$, $x_0 \in \mathbb{R}$ *and* $h > 0$. *Then, the following identity holds:*

$$F^{(3)}(x_0) = \frac{F(x_0 - 2h) - 7F(x_0 - h) + 16F(x_0) - 16F(x_0 + h) + 7F(x_0 + 2h) - F(x_0 + 3h)}{2h^3} + C_3 h,$$

$$(A.33)$$

where C_3 is a constant that depends only on F and h.

Proof. From the Taylor's expansion, we have:

$$F(x_0 - 2h) = F(x_0) - 2F'(x_0)h + 2F^{(2)}(x_0)h^2 - \frac{8}{6}F^{(3)}(x_0)h^3 + \frac{16}{24}F^{(4)}(\theta_{-2h})h^4,$$

$$F(x_0 - h) = F(x_0) - F'(x_0)h + \frac{1}{2}F^{(2)}(x_0)h^2 - \frac{1}{6}F^{(3)}(x_0)h^3 + \frac{1}{24}F^{(4)}(\theta_{-h})h^4,$$

$$F(x_0 + h) = F(x_0) + F'(x_0)h + \frac{1}{2}F^{(2)}(x_0)h^2 + \frac{1}{6}F^{(3)}(x_0)h^3 + \frac{1}{24}F^{(4)}(\theta_h)h^4,$$

$$F(x_0 + 2h) = F(x_0) + 2F'(x_0)h + 2F^{(2)}(x_0)h^2 + \frac{8}{6}F^{(3)}(x_0)h^3 + \frac{16}{24}F^{(4)}(\theta_{2h})h^4,$$

$$F(x_0 + 3h) = F(x_0) + 3F'(x_0)h + \frac{9}{2}F^{(2)}(x_0)h^2 + \frac{27}{6}F^{(3)}(x_0)h^3 + \frac{81}{24}F^{(4)}(\theta_{3h})h^4,$$

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:

$$F(x_0 - 2h) = F(x_0) - 2F'(x_0)h + \frac{4}{2}F^{(2)}(x_0)h^2 - \frac{8}{6}F^{(3)}(x_0)h^3 + \frac{16}{24}F^{(4)}(\theta_{-2h})h^4,$$

$$-7F(x_0 - h) = -7F(x_0) + 7F'(x_0)h - \frac{7}{2}F^{(2)}(x_0)h^2 + \frac{7}{6}F^{(3)}(x_0)h^3 - \frac{7}{24}F^{(4)}(\theta_{-h})h^4,$$

$$16F(x_0) = 16F(x_0),$$

$$-16F(x_0 + h) = -16F(x_0) - 16F'(x_0)h - \frac{16}{2}F^{(2)}(x_0)h^2 - \frac{16}{6}F^{(3)}(x_0)h^3 - \frac{16}{24}F^{(4)}(\theta_h)h^4,$$

$$7F(x_0 + 2h) = 7F(x_0) + 14F'(x_0)h + \frac{28}{2}F^{(2)}(x_0)h^2 + \frac{56}{6}F^{(3)}(x_0)h^3 + \frac{112}{24}F^{(4)}(\theta_{2h})h^4,$$

$$-F(x_0 + 3h) = -F(x_0) - 3F'(x_0)h - \frac{9}{2}F^{(2)}(x_0)h^2 - \frac{27}{6}F^{(3)}(x_0)h^3 - \frac{81}{24}F^{(4)}(\theta_{3h})h^4.$$

Summing all these equations, we have:

$$F(x_0 - 2h) - 7F(x_0 - h) + 16F(x_0) - 16F(x_0 + h) + 7F(x_0 + 2h) - F(x_0 + 3h) = 2F^{(3)}(x_0)h^3 - 2C_3h^4,$$

we get the desired Formula (A.33) with C_3 given by:

$$C_3 = \frac{1}{48}\left(-16F^{(4)}(\theta_{-2h}) + 7F^{(4)}(\theta_{-h}) + 16F^{(4)}(\theta_h) - 112F^{(4)}(\theta_{2h}) + 81F^{(4)}(\theta_{3h}) \right). \quad (A.34)$$

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. $\qquad\square$

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 (2.48) 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 (2.39). It follows that:

$$\int_{\overline{x}}^{\overline{x}+h} q(\xi)\,d\xi + \int_{\overline{x}-h}^{\overline{x}} q(\xi)\,d\xi = Q(\overline{x}+h) - Q(\overline{x}-h),$$
$$\int_{\overline{x}+h}^{\overline{x}+2h} q(\xi)\,d\xi + \int_{\overline{x}-2h}^{\overline{x}-h} q(\xi)\,d\xi = Q(\overline{x}+2h) - Q(\overline{x}-2h) - (Q(\overline{x}+h) - Q(\overline{x}-h)).$$

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. $\qquad\square$

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 (2.48) 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 (2.42) 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 (2.42) as:

$$q_i(x; Q) = q_{L,i} + \frac{(\Delta q_i + q_{6,i})}{\Delta x}(x - x_{i-\frac{1}{2}}) - \frac{q_{6,i}}{\Delta x^2}(x - x_{i-\frac{1}{2}})^2 \qquad (A.40)$$

and we write q using its Taylor expansion assuming $q \in C^4(\mathbb{R})$:

$$q(x) = q(x_{i-\frac{1}{2}}) + q'(x_{i-\frac{1}{2}})(x - x_{i-\frac{1}{2}}) + \frac{q''(x_{i-\frac{1}{2}})}{2}(x - x_{i-\frac{1}{2}})^2 + \frac{q^{(3)}(\theta_i)}{6}(x - x_{i-\frac{1}{2}})^3, \quad (A.41)$$

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:

$$q'(x_{i-\frac{1}{2}}) - \frac{(\Delta q_i + q_{6,i})}{\Delta x}, \qquad (A.42)$$

and:

$$\frac{q''(x_{i-\frac{1}{2}})}{2} - \left(-\frac{q_{6,i}}{\Delta x^2}\right). \qquad (A.43)$$

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 (2.45) with Equations (2.49) and (2.50), we may write $q_{6,i}$ as:

$$q_{6,i} = \frac{1}{4}\left(Q_{i-2} - 6Q_{i-1} + 10Q_i - 6Q_{i+1} + Q_{i+2}\right). \qquad (A.44)$$

Recalling the definition of Δq_i from Equation (2.43), and applying Equations (2.49) and (2.50), we may express Δq_i as:

$$\Delta q_i = \frac{1}{12}\left(Q_{i-2} - 8Q_{i-1} + 8Q_{i+1} - Q_{i+2}\right). \qquad (A.45)$$

Finally, we combine Equations (A.44) and (A.45) and write their sum as:

$$\frac{(\Delta q_i + q_{6,i})}{\Delta x} = \frac{2Q_{i-2} - 13Q_{i-1} + 15Q_i - 5Q_{i+1} + Q_{i+2}}{6\Delta x}. \qquad (A.46)$$

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:*

$$
q'(\overline{x}) = \frac{1}{6h}\left(\frac{2}{h}\int_{\overline{x}-2h}^{\overline{x}-h} q(x)\,dx - \frac{13}{h}\int_{\overline{x}-h}^{\overline{x}} q(x)\,dx + \frac{15}{h}\int_{\overline{x}}^{\overline{x}+h} q(x)\,dx \right.
$$
$$
\left. - \frac{5}{h}\int_{\overline{x}+h}^{\overline{x}+2h} q(x)\,dx + \frac{1}{h}\int_{\overline{x}+2h}^{\overline{x}+3h} q(x)\,dx \right) + C_2 h^2, \tag{A.47}
$$

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 (2.39). Like in Proposition A.2, we have:

$$
\frac{1}{6h}\left(\frac{2}{h}\int_{\overline{x}-2h}^{\overline{x}-h} q(x)\,dx - \frac{13}{h}\int_{\overline{x}-h}^{\overline{x}} q(x)\,dx + \frac{15}{h}\int_{\overline{x}}^{\overline{x}+h} q(x)\,dx - \frac{5}{h}\int_{\overline{x}+h}^{\overline{x}+2h} q(x)\,dx + \frac{1}{h}\int_{\overline{x}+2h}^{\overline{x}+3h} q(x)\,dx \right)
$$
$$
= \frac{1}{6h}\left(\frac{2}{h}\big(Q(\overline{x}-h) - Q(\overline{x}-2h)\big) - \frac{13}{h}\big(Q(\overline{x}) - Q(\overline{x}-h)\big) + \frac{15}{h}\big(Q(\overline{x}+h) - Q(\overline{x})\big) \right.
$$
$$
\left. - \frac{5}{h}\big(Q(\overline{x}+2h) - Q(\overline{x}+h)\big) + \frac{1}{h}\big(Q(\overline{x}+3h) - Q(\overline{x}+2h)\big) \right)
$$
$$
= \frac{1}{6h^2}\left(-2Q(\overline{x}-2h) + 15Q(\overline{x}-h) - 28Q(\overline{x}) + 20Q(\overline{x}+h) - 6Q(\overline{x}+2h) + Q(\overline{x}+3h) \right),
$$

which consists of the finite-difference scheme from Lemma A.2. Therefore, Equation (A.47) follows from Lemma A.2 with:

$$
C_2 = C_2(\mu_1, \mu_2) = \frac{1}{24}\left(128 q^{(3)}(\mu_1) - 116 q^{(3)}(\mu_2) \right), \tag{A.48}
$$

where $\mu_1, \mu_2 \in [x_0 - 2h, x_0 + 3h]$, which concludes the proof. $\qquad\square$

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:

$$
-\frac{2q_{6,i}}{\Delta x^2} = -\frac{1}{2\Delta x^2}\left(Q_{i-2} - 6Q_{i-1} + 10Q_i - 6Q_{i+1} + Q_{i+2} \right). \tag{A.50}
$$

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:*

$$q''(\overline{x}) = \frac{1}{2h^2} \left(-\frac{1}{h} \int_{\overline{x}-2h}^{\overline{x}-h} q(x)\,dx + \frac{6}{h} \int_{\overline{x}-h}^{\overline{x}} q(x)\,dx - \frac{10}{h} \int_{\overline{x}}^{\overline{x}+h} q(x)\,dx \right.$$
$$\left. + \frac{6}{h} \int_{\overline{x}+h}^{\overline{x}+2h} q(x)\,dx - \frac{1}{h} \int_{\overline{x}+2h}^{\overline{x}+3h} q(x)\,dx \right) + C_3 h, \tag{A.51}$$

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:

$$\frac{1}{2h^2} \left(-\frac{1}{h} \int_{\overline{x}-2h}^{\overline{x}-h} q(x)\,dx + \frac{6}{h} \int_{\overline{x}-h}^{\overline{x}} q(x)\,dx - \frac{10}{h} \int_{\overline{x}}^{\overline{x}+h} q(x)\,dx + \frac{6}{h} \int_{\overline{x}+h}^{\overline{x}+2h} q(x)\,dx - \frac{1}{h} \int_{\overline{x}+2h}^{\overline{x}+3h} q(x)\,dx \right)$$

$$= \frac{1}{2h^2} \left(-\frac{1}{h} \left(Q(\overline{x}-h) - Q(\overline{x}-2h) \right) + \frac{6}{h} \left(Q(\overline{x}) - Q(\overline{x}-h) \right) - \frac{10}{h} \left(Q(\overline{x}+h) - Q(\overline{x}) \right) \right.$$

$$\left. + \frac{6}{h} \left(Q(\overline{x}+2h) - Q(\overline{x}+h) \right) - \frac{1}{h} \left(Q(\overline{x}+3h) - Q(\overline{x}+2h) \right) \right)$$

$$= \frac{1}{2h^3} \left(Q(\overline{x}-2h) - 7Q(\overline{x}-h) + 16Q(\overline{x}) - 16Q(\overline{x}+h) + 7Q(\overline{x}+2h) - Q(\overline{x}+3h) \right),$$

which consists of the finite-difference scheme from Lemma A.3. Therefore, Equation (A.51) follows from Lemma A.3 with:

$$C_3 = C_3(\mu_1, \mu_2) = \frac{1}{48} \left(104 q^{(3)}(\mu_1) - 128 q^{(3)}(\mu_2) \right), \tag{A.52}$$

where $\mu_1, \mu_2 \in [x_0 - 2h, x_0 + 3h]$, which concludes the proof. $\qquad\square$

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 (2.48) satisfies:*

$$q''(x_{i-\frac{1}{2}}) - \left(-\frac{2q_{6,i}}{\Delta x^2} \right) = C_3 \Delta x, \tag{A.53}$$

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 (2.42) with the parameters $q_{R,i}$ and $q_{L,i}$ obeying Equations (2.49) and (2.50) 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. $\qquad \square$

Appendix B

Code availability and technical considerations

The codes needed for this work have been openly built on GitHub. The PPM implementation for the one-dimensional advection equation used in Chapter 2 is available at https://github.com/luanfs/FV3_adv_1D.

The dimension splitting method implementation used in Chapter 3 is available at https://github.com/luanfs/FV3_adv_2D.

The cubed-sphere code used in Chapter 4, 5 and 6 is available at https://github.com/luanfs/FV3_container. This code was executed using Docker, utilizing the containerized version of the SHiELD model developed by Cheng et al. (2022) (https://www.gfdl.noaa.gov/shield-in-a-box/, last accessed on March 27th, 2024).

The ENDGame shallow-water solver used in Chapter 6 is available at https://github.com/luanfs/endgame.

All these codes are written in Fortran, and the output graphs were generated using Python3.

The numerical experiments conducted in this study were primarily performed on the ybytu machine from Labmap (https://labmap.ime.usp.br, last accessed on March 27th, 2024) with the following specifications:

- Processor: 2x Intel Xeon Gold 6144 (8c/16t) 3.50 GHz;

- Memory: 540 GB;

- Disk: 4TB;

- Operating System: Debian GNU/Linux.

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/10.1016/B978-0-12-460817-7.50009-4 (cit. on pp. 4, 11, 73, 75, 76, 78).

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/10.1016/0167-8191(96)80002-0 (cit. on p. 3).

Brachet, M. (2018). *Schémas compacts hermitiens sur la Sphère : applications en climatologie et océanographie numérique* (Publication No. 2018LORR0111) [Theses]. Université de Lorraine. https://theses.hal.science/tel-01886875 (cit. on p. 95).

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, 10, 22).

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/10.1016/j.jcp.2008.01.033 (cit. on p. 93).

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/10.1029/2020MS002280 (cit. on pp. 6, 58, 62–64, 68, 71, 73, 74, 93).

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/10.1002/qj.3125 (cit. on pp. 28, 35, 51).

Cheng, K.-Y., Harris, L. M., & Sun, Y. Q. (2022). Enhancing the accessibility of unified modeling systems: GFDL System for High-resolution prediction on Earth-to-local domains (SHiELD) v2021b in a container. *Geoscientific Model Development*, *15*(3), 1097–1105. https://doi.org/10.5194/gmd-15-1097-2022 (cit. on p. 163).

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/10.1016/0021-9991(84)90143-8 (cit. on pp. 4, 9, 22, 24–26, 28, 32).

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/10.1007/978-3-642-58604-0 (cit. on p. 144).

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. 58, 68).

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/10.1016/j.camwa.2005.08.004 (cit. on p. 44).

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. 19).

Durran, D. (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. 21).

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. 10, 12, 24).

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. 144).

Galewsky, J., SCOTT, R., & Polvani, L. (2004). An initial-value problem to test numerical models of the shallow-water equations. *Tellus Series A-dynamic Meteorology and Oceanography - TELLUS A-DYN METEOROL OCEANOG*, *56*, 429–440. https://doi.org/10.1111/j.1600-0870.2004.00071.x (cit. on p. 137).

Gao, K., Harris, L., Zhou, L., Bender, M., & Morin, M. (2021). On the sensitivity of hurricane intensity and structure to horizontal tracer advection schemes in FV3. *Journal of the Atmospheric Sciences*, *78*(9), 3007–3021. https://doi.org/10.1175/JAS-D-20-0331.1 (cit. on p. 141).

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. 9, 10, 24).

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. 19, 67).

Harper, K., Uccellini, L. W., Kalnay, E., Carey, K., & Morone, L. (2007). 50th anniversary of operational numerical weather prediction. *Bulletin of the American Meteorological Society*, *88*(5), 639–650. https://doi.org/10.1175/BAMS-88-5-639 (cit. on p. 1).

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. 9, 10, 25, 26, 29, 64, 111, 125).

Harris, L., & 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, 9).

Harris, L., 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. 63, 71).

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/10.1029/2020MS002223 (cit. on p. 5).

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/10.1016/j.ocemod.2022.102140 (cit. on p. 57).

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. 44, 49).

Holton, J. R., & Hakim, G. J. (2012). *An introduction to dynamic meteorology.* Elsevier. (Cit. on p. 141).

Jablonowski, C., & Williamson, D. (2011). The Pros and Cons of Diffusion, Filters and Fixers in Atmospheric General Circulation Models. in: Lauritzen P., Jablonowski C., Taylor M., Nair R. (eds) Numerical Techniques for Global Atmospheric Models. Lecture Notes in Computational Science and Engineering. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-11640-7_13 (cit. on p. 125).

Jia, H., & Li, K. (2011). A third accurate operator splitting method. *Mathematical and Computer Modelling*, *53*(1), 387–396. https://doi.org/10.1016/j.mcm.2010.09.005 (cit. on p. 44).

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/10.1029/2018MS001517 (cit. on p. 60).

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/10.1016/j.amc.2015.04.053 (cit. on pp. 58, 67, 68).

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/10.1175/MWR-D-13-00176.1 (cit. on pp. 58, 67, 68).

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. 3).

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/10.1016/j.jcp.2009.10.036 (cit. on p. 35).

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. 10).

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/10.1175/MWR3425.1 (cit. on pp. 151, 153).

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/10.1175/1520-0493(1996)124<2588:CEUTSM>2.0.CO;2 (cit. on p. 9).

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. 9).

LeVeque, R. J. (1990). *Numerical Methods for Conservation Laws*. Birkhäuser Basel. https://doi.org/10.1007/978-3-0348-5116-9 (cit. on pp. 13, 50).

LeVeque, R. J. (2002). *Finite Volume Methods for Hyperbolic Problems*. Cambridge University Press. https://doi.org/10.1017/CBO9780511791253 (cit. on pp. 13, 23, 147, 149, 150).

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, 6, 10, 20, 26, 32, 57, 89, 91, 111).

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, 91).

REFERENCES

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. 26).

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, 9, 19, 23, 28, 35, 43, 44, 49, 56–58, 153).

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/10.1002/qj.49712354416 (cit. on pp. 4–6, 9, 20, 111–113, 115, 116, 119, 120).

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. 19).

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. 5).

Mouallem, J., Harris, L., & Benson, R. (2022). Multiple same-level and telescoping nesting in GFDL's dynamical core. *Geoscientific Model Development*, *15*(11), 4355–4371. https://doi.org/10.5194/gmd-15-4355-2022 (cit. on p. 5).

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/10.1029/2023MS003712 (cit. on pp. 5, 74, 93, 95, 127, 135).

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/10.1016/j.jcp.2010.08.014 (cit. on pp. 31, 51, 54, 94, 104, 107).

Nair, R. D., Thomas, S. J., & Loft, R. D. (2005a). A Discontinuous Galerkin Global Shallow Water Model. *Monthly Weather Review*, *133*(4), 876–888. https://doi.org/10.1175/MWR2903.1 (cit. on p. 112).

Nair, R. D., Thomas, S. J., & Loft, R. D. (2005b). 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. 67).

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. S. (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, 57, 126, 127).

Peixoto, P. S., & 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, 57).

Peixoto, P., & Barros, S. R. (2014). On vector field reconstructions for semi-Lagrangian transport methods on geodesic staggered grids. *Journal of Computational Physics*, *273*, 185–211. https://doi.org/10.1016/j.jcp.2014.04.043 (cit. on p. 57).

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, 9).

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/10.1016/j.jcp.2007.07.022 (cit. on pp. 4–7, 10, 19, 49, 56–58, 71, 72, 79, 81, 89, 91, 111).

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/10.1002/qj.49712253209 (cit. on pp. 57, 58, 112).

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. 35).

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, 58).

Randall, D. (2022). *An Introduction to Numerical Modeling of the Atmosphere*. (Cit. on p. 1).

Randall, D., 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. 2, 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. 43).

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. 113).

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/10.1006/jcph.1996.0047 (cit. on pp. 4, 6, 57, 58, 62, 68).

REFERENCES

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/10.1016/j.jcp.2005.08.027 (cit. on pp. 68, 93).

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, 57, 62, 79).

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. 57).

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. 4).

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/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/10.1007/978-0-387-21738-3 (cit. on pp. 23, 143).

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. 44).

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. 151).

Thuburn, J., Zerroukat, M., Wood, N., & Staniforth, A. (2010). Coupling a mass-conserving semi-Lagrangian scheme (SLICE) to a semi-implicit discretization of the shallow-water equations: Minimizing the dependence on a reference atmosphere. *Quarterly Journal of the Royal Meteorological Society*, *136*(646), 146–154. https://doi.org/10.1002/qj.517 (cit. on p. 127).

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., & Li, Y. (2000). Numerical simulations of Rossby–Haurwitz waves. *Tellus A*, *52*(2), 181–189. https://doi.org/10.1034/j.1600-0870.2000.00107.x (cit. on p. 136).

Trefethen, L. N. (2000). *Spectral Methods in MATLAB*. Society for Industrial; Applied Mathematics. https://doi.org/10.1137/1.9780898719598 (cit. on pp. 26, 150).

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. 19).

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. 4).

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/10.1016/j.jcp.2010.04.044 (cit. on p. 67).

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/10.1016/0021-9991(77)90095-X (cit. on pp. 4, 10, 23, 24).

Weller, H., Thuburn, J., & Cotter, C. J. (2012). Computational modes and grid imprinting on five quasi-uniform spherical C grids. *Monthly Weather Review*, *140*(8), 2734–2755. https://doi.org/10.1175/MWR-D-11-00193.1 (cit. on pp. 5, 57).

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. 10).

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/10.1016/j.jcp.2008.04.026 (cit. on p. 10).

Whitehead, J. P., Jablonowski, C., Rood, R. B., & Lauritzen, P. H. (2011). A Stability Analysis of Divergence Damping on a Latitude–Longitude Grid. *Monthly Weather Review*, *139*(9), 2976–2993. https://doi.org/10.1175/2011MWR3607.1 (cit. on p. 125).

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).

Williamson, D. L., Drake, J. B., Hack, J. J., Jakob, R., & Swarztrauber, P. N. (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. 75, 78, 94, 127, 128, 132, 135, 137).

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. 5).

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/10.1002/qj.2235 (cit. on p. 2).

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. 10).

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/10.1002/qj.4230 (cit. on p. 73).

Zerroukat, M., Wood, N., & Staniforth, A. (2006). The parabolic spline method (psm) for conservative transport problems. *International Journal for Numerical Methods in Fluids*, *51*(11), 1297–1318. https://doi.org/doi.org/10.1002/fld.1154 (cit. on p. 13).

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., M.Harris, L., 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. 64).