Enhancing Accuracy of FV3 finite-volume operators

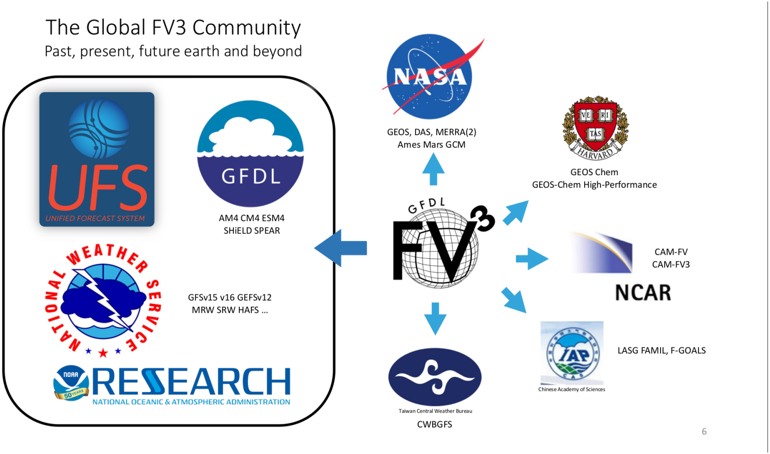
Luan Santos
Advisor: Prof. Dr. Pedro Peixoto
In colaboration with Joseph Mouallem and Lucas Harris (GFDL/NOAA)

Institute of Mathematics and Statistics (IME)
University of São Paulo (USP)
luan.santos@usp.br

Numerical methods for GFD seminar
October 6, 2023

Funded by FAPESP (grant number 2020/10280-4)

FV3



www.gfdl.noaa.gov/fv3

github.com/NOAA-GFDL/GFDL_atmos_cubed_sphere

Aims

- To discuss strategies to reduce larger errors at the edges (grid imprinting).
- Introduce the duogrid to compute stencils near to the edges.
- Improve the accuracy of the discrete operators from FV3 dynamical core.

The gnomonic equiangular cubed-sphere from Ronchi et al. (1996)

Six maps from a cube panel p to the sphere $\Phi_p : [-\alpha, \alpha]^2 \to \mathbb{S}^2_R$, $p = 1, \cdots, 6$, $\alpha = \frac{\pi}{4}$.

The gnomonic equiangular cubed-sphere from Ronchi et al. (1996)

Six maps from a cube panel p to the sphere $\Phi_p : [-\alpha, \alpha]^2 \to \mathbb{S}^2_R$, $p = 1, \cdots, 6$, $\alpha = \frac{\pi}{4}$.

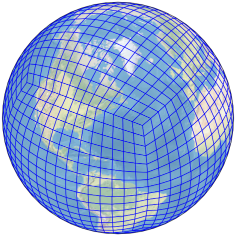


Figure: CS grid with $N = 15$ cells along a coordinate axis

The gnomonic equiangular cubed-sphere from Ronchi et al. (1996)

Six maps from a cube panel p to the sphere $\Phi_p : [-\alpha, \alpha]^2 \to \mathbb{S}^2_R$, $p = 1, \cdots, 6$, $\alpha = \frac{\pi}{4}$.

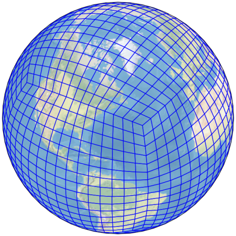


Figure: CS grid with $N = 15$ cells along a coordinate axis

- $\Phi_1(x, y) = \frac{R(1, \tan x, \tan y)}{\sqrt{1 + \tan^2 x + \tan^2 y}}$;
- $\sqrt{g}(x, y) = \sqrt{\det\left(D\Phi_p^T D\Phi_p\right)}$;

The gnomonic equiangular cubed-sphere from Ronchi et al. (1996)

Six maps from a cube panel p to the sphere $\Phi_p : [-\alpha, \alpha]^2 \to \mathbb{S}^2_R$, $p = 1, \cdots, 6$, $\alpha = \frac{\pi}{4}$.

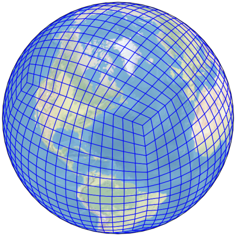


Figure: CS grid with $N = 15$ cells along a coordinate axis

- $\Phi_1(x, y) = \frac{R(1, \tan x, \tan y)}{\sqrt{1+\tan^2 x + \tan^2 y}}$;
- $\sqrt{g}(x, y) = \sqrt{\det\left(D\Phi_p^T D\Phi_p\right)}$;
- Point on the sphere is represented by (x, y, p);
- Gridpoints use indexes (i, j, p);
- Edge values are represented using half integers.

Linear advection equation on the cubed-sphere

$$\begin{cases} \partial_t(\sqrt{g}q) + \partial_x(u\sqrt{g}q) + \partial_y(v\sqrt{g}q) = 0, \\ q_0(x,y,p) = q(x,y,p,0), \end{cases}$$

Linear advection equation on the cubed-sphere

$$\begin{cases} \partial_t(\sqrt{g}q) + \partial_x(u\sqrt{g}q) + \partial_y(v\sqrt{g}q) = 0, \\ q_0(x,y,p) = q(x,y,p,0), \end{cases}$$

for each panel $p = 1, \ldots, 6$, $\forall (x,y,t) \in [-\alpha, \alpha]^2 \times [0, T]$.

Linear advection equation on the cubed-sphere

$$\begin{cases} \partial_t(\sqrt{g}q) + \partial_x(u\sqrt{g}q) + \partial_y(v\sqrt{g}q) = 0, \\ q_0(x,y,p) = q(x,y,p,0), \end{cases}$$

for each panel $p = 1, \ldots, 6$, $\forall (x,y,t) \in [-\alpha,\alpha]^2 \times [0,T]$.

- $q = q(x,y,p,t)$ is the advected quantity;
- u and v are the contravariant components of the wind.

Linear advection equation on the cubed-sphere

$$\begin{cases} \partial_t(\sqrt{g}q) + \partial_x(u\sqrt{g}q) + \partial_y(v\sqrt{g}q) = 0, \\ q_0(x, y, p) = q(x, y, p, 0), \end{cases}$$

for each panel $p = 1, \ldots, 6$, $\forall (x, y, t) \in [-\alpha, \alpha]^2 \times [0, T]$.

- $q = q(x, y, p, t)$ is the advected quantity;
- u and v are the contravariant components of the wind.

In each panel, we can solve the advection equation just like on the plane.

Linear advection equation on the cubed-sphere

$$\begin{cases} \partial_t(\sqrt{g}q) + \partial_x(u\sqrt{g}q) + \partial_y(v\sqrt{g}q) = 0, \\ q_0(x,y,p) = q(x,y,p,0), \end{cases}$$

for each panel $p = 1, \ldots, 6$, $\forall (x,y,t) \in [-\alpha,\alpha]^2 \times [0,T]$.

- $q = q(x,y,p,t)$ is the advected quantity;
- u and v are the contravariant components of the wind.

In each panel, we can solve the advection equation just like on the plane. However, we must to take into account:

1. Metric tensor term $\sqrt{g}$;

Linear advection equation on the cubed-sphere

$$\begin{cases} \partial_t(\sqrt{g}q) + \partial_x(u\sqrt{g}q) + \partial_y(v\sqrt{g}q) = 0, \\ q_0(x, y, p) = q(x, y, p, 0), \end{cases}$$

for each panel $p = 1, \ldots, 6$, $\forall (x, y, t) \in [-\alpha, \alpha]^2 \times [0, T]$.

- $q = q(x, y, p, t)$ is the advected quantity;
- u and v are the contravariant components of the wind.

In each panel, we can solve the advection equation just like on the plane. However, we must to take into account:

1. Metric tensor term $\sqrt{g}$;
2. Cubed-sphere topology;

Linear advection equation on the cubed-sphere

$$\begin{cases} \partial_t(\sqrt{g}q) + \partial_x(u\sqrt{g}q) + \partial_y(v\sqrt{g}q) = 0, \\ q_0(x,y,p) = q(x,y,p,0), \end{cases}$$

for each panel $p = 1, \ldots, 6$, $\forall (x,y,t) \in [-\alpha,\alpha]^2 \times [0,T]$.

- $q = q(x,y,p,t)$ is the advected quantity;
- u and v are the contravariant components of the wind.

In each panel, we can solve the advection equation just like on the plane. However, we must to take into account:

1. Metric tensor term $\sqrt{g}$;
2. Cubed-sphere topology;
3. Coordinate system discontinuity at the cube edges.

Dimension splitting scheme

Lie-Trotter splitting: firstly we solve

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

and then solve

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

Dimension splitting scheme

Lie-Trotter splitting: firstly we solve

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

and then solve

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

Using numerical updates **F** and **G**:

$$\begin{aligned} Q^x &= Q^n + \mathbf{F}(Q^n, \tilde{u}^n), \\ Q^{yx} &= Q^x + \mathbf{G}(Q^x, \tilde{v}^n). \end{aligned}$$

Swapping the directions x and y, we obtain Q^{xy}.

Dimension splitting scheme

Lie-Trotter splitting: firstly we solve

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

and then solve

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

Using numerical updates **F** and **G**:

$$\begin{aligned} Q^x &= Q^n + \mathbf{F}(Q^n, \tilde{u}^n), \\ Q^{yx} &= Q^x + \mathbf{G}(Q^x, \tilde{v}^n). \end{aligned}$$

Swapping the directions x and y, we obtain Q^{xy}.

Average Lie-Trotter (**AVLT**):

$$\begin{aligned} Q^{n+1} &= \frac{(Q^{xy} + Q^{yx})}{2}, \quad \text{or} \\ Q^{n+1} &= Q^n - \Delta t \mathbb{D}^n_{ijp}, \end{aligned}$$

where

$$\begin{aligned} \mathbb{D}^n_{ijp} = \frac{-1}{\Delta t \sqrt{g_{ijp}}} \Bigg[& \mathbf{F}\left(Q^n + \frac{1}{2}\mathbf{G}(Q^n, \tilde{v}^n), \tilde{u}^n\right) \\ & + \mathbf{G}\left(Q^n + \frac{1}{2}\mathbf{F}(Q^n, \tilde{u}^n), \tilde{v}^n\right)\Bigg]. \end{aligned}$$

is the discrete divergence.

Dimension splitting scheme

Lie-Trotter splitting: firstly we solve

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

and then solve

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

Using numerical updates **F** and **G**:

$$\begin{aligned} Q^x &= Q^n + \mathbf{F}(Q^n, \tilde{u}^n), \\ Q^{yx} &= Q^x + \mathbf{G}(Q^x, \tilde{v}^n). \end{aligned}$$

Swapping the directions x and y, we obtain Q^{xy}.

Average Lie-Trotter (**AVLT**):

$$\begin{aligned} Q^{n+1} &= \frac{(Q^{xy} + Q^{yx})}{2}, \quad \text{or} \\ Q^{n+1} &= Q^n - \Delta t \mathbb{D}^n_{ijp}, \end{aligned}$$

where

$$\mathbb{D}^n_{ijp} = \frac{-1}{\Delta t \sqrt{g_{ijp}}} \left[\mathbf{F}\left(Q^n + \frac{1}{2}\mathbf{G}(Q^n, \tilde{v}^n), \tilde{u}^n \right) + \mathbf{G}\left(Q^n + \frac{1}{2}\mathbf{F}(Q^n, \tilde{u}^n), \tilde{v}^n \right) \right].$$

is the discrete divergence.

Lin and Rood (1996) scheme

Lin and Rood (1996) use inner advective operators **f**, **g** to eliminate splitting error when the scalar field is constant.

$$\mathbb{D}^n_{ijp} = \frac{-1}{\Delta t \sqrt{g_{ijp}}} \left[\mathbf{F}\left(Q^n + \frac{1}{2}\mathbf{g}(Q^n, \tilde{v}^n), \tilde{u}^n \right) + \mathbf{G}\left(Q^n + \frac{1}{2}\mathbf{f}(Q^n, \tilde{u}^n), \tilde{v}^n \right) \right].$$

They eliminate the splitting error provided $\mathbf{F}(Q^n, \tilde{u}) = -\overline{q}\frac{\Delta t}{\Delta x}\delta_i(\sqrt{g}\tilde{u})^n_{ijp}$ when $Q^n = \overline{q}$ is constant.

Lin and Rood (1996) scheme

Lin and Rood (1996) use inner advective operators **f**, **g** to eliminate splitting error when the scalar field is constant.

$$\mathbb{D}^n_{ijp} = \frac{-1}{\Delta t \sqrt{g_{ijp}}} \left[\mathbf{F}\left(Q^n + \frac{1}{2}\mathbf{g}(Q^n, \tilde{v}^n), \tilde{u}^n \right) + \mathbf{G}\left(Q^n + \frac{1}{2}\mathbf{f}(Q^n, \tilde{u}^n), \tilde{v}^n \right) \right].$$

They eliminate the splitting error provided $\mathbf{F}(Q^n, \tilde{u}) = -\overline{q}\frac{\Delta t}{\Delta x}\delta_i(\sqrt{g}\tilde{u})^n_{ijp}$ when $Q^n = \overline{q}$ is constant. In particular, we shall consider the inner advective operators from Putman and Lin (2007) (**PL07**).

Lin and Rood (1996) scheme

Lin and Rood (1996) use inner advective operators **f**, **g** to eliminate splitting error when the scalar field is constant.

$$\mathbb{D}^n_{ijp} = \frac{-1}{\Delta t \sqrt{g_{ijp}}} \left[\mathbf{F}\left(Q^n + \frac{1}{2}\mathbf{g}(Q^n, \tilde{v}^n), \tilde{u}^n \right) + \mathbf{G}\left(Q^n + \frac{1}{2}\mathbf{f}(Q^n, \tilde{u}^n), \tilde{v}^n \right) \right].$$

They eliminate the splitting error provided $\mathbf{F}(Q^n, \tilde{u}) = -\overline{q}\frac{\Delta t}{\Delta x}\delta_i(\sqrt{g}\tilde{u})^n_{ijp}$ when $Q^n = \overline{q}$ is constant. In particular, we shall consider the inner advective operators from Putman and Lin (2007) (**PL07**).

$$\mathbf{f}_{ijp}(Q^n, \tilde{u}^n) = -Q^n_{ijp} + \frac{Q^n_{ijp} + \mathbf{F}_{ijp}(Q^n, \tilde{u}^n)}{1 - \frac{\Delta t}{\Delta x}\delta_i(\sqrt{g}\tilde{u})^n_{ijp}}.$$

Lin and Rood (1996) scheme

Lin and Rood (1996) use inner advective operators **f**, **g** to eliminate splitting error when the scalar field is constant.

$$\mathbb{D}^n_{ijp} = \frac{-1}{\Delta t\sqrt{g_{ijp}}}\left[\mathbf{F}\left(Q^n + \frac{1}{2}\mathbf{g}(Q^n, \tilde{v}^n), \tilde{u}^n\right) + \mathbf{G}\left(Q^n + \frac{1}{2}\mathbf{f}(Q^n, \tilde{u}^n), \tilde{v}^n\right)\right].$$

They eliminate the splitting error provided $\mathbf{F}(Q^n, \tilde{u}) = -\overline{q}\frac{\Delta t}{\Delta x}\delta_i(\sqrt{g}\tilde{u})^n_{ijp}$ when $Q^n = \overline{q}$ is constant. In particular, we shall consider the inner advective operators from Putman and Lin (2007) (**PL07**).

$$\mathbf{f}_{ijp}(Q^n, \tilde{u}^n) = -Q^n_{ijp} + \frac{Q^n_{ijp} + \mathbf{F}_{ijp}(Q^n, \tilde{u}^n)}{1 - \frac{\Delta t}{\Delta x}\delta_i(\sqrt{g}\tilde{u})^n_{ijp}}.$$

We still need to address how to:

1. solve the 1D advection equation;

Lin and Rood (1996) scheme

Lin and Rood (1996) use inner advective operators **f**, **g** to eliminate splitting error when the scalar field is constant.

$$\mathbb{D}_{ijp}^{n} = \frac{-1}{\Delta t \sqrt{g_{ijp}}} \left[\mathbf{F}\left(Q^n + \frac{1}{2}\mathbf{g}(Q^n, \tilde{v}^n), \tilde{u}^n \right) + \mathbf{G}\left(Q^n + \frac{1}{2}\mathbf{f}(Q^n, \tilde{u}^n), \tilde{v}^n \right) \right].$$

They eliminate the splitting error provided $\mathbf{F}(Q^n, \tilde{u}) = -\overline{q}\frac{\Delta t}{\Delta x}\delta_i(\sqrt{g}\tilde{u})_{ijp}^{n}$ when $Q^n = \overline{q}$ is constant. In particular, we shall consider the inner advective operators from Putman and Lin (2007) (**PL07**).

$$\mathbf{f}_{ijp}(Q^n, \tilde{u}^n) = -Q_{ijp}^{n} + \frac{Q_{ijp}^{n} + \mathbf{F}_{ijp}(Q^n, \tilde{u}^n)}{1 - \frac{\Delta t}{\Delta x}\delta_i(\sqrt{g}\tilde{u})_{ijp}^{n}}.$$

We still need to address how to:

1. solve the 1D advection equation;
2. compute stencils near to the cube edges.

Piecewise Parabolic Method (PPM)

- $\partial_t(\sqrt{g}q) + \partial_x(u\sqrt{g}q) = 0$

Piecewise Parabolic Method (PPM)

- $\partial_t(\sqrt{g}q) + \partial_x(u\sqrt{g}q) = 0$
- $\sqrt{g}q$ average values update:

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

Piecewise Parabolic Method (PPM)

- $\partial_t(\sqrt{g}q) + \partial_x(u\sqrt{g}q) = 0$
- $\sqrt{g}q$ average values update:

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

- Time-averaged flux:

$$\begin{aligned} F^n_{i+\frac{1}{2}} &\approx \frac{1}{\Delta t}\int_{t^n}^{t^{n+1}} (u\sqrt{g}q)(x_{i+\frac{1}{2}}, t)\, dt \\ &= \frac{1}{\Delta t}\int_{x^d_{i+\frac{1}{2}}(t^n)}^{x_{i+\frac{1}{2}}} (\sqrt{g}q)(x, t^n)\, dx \end{aligned}$$

$x^d_{i+\frac{1}{2}}$ is the departure point (DP).

Piecewise Parabolic Method (PPM)

- $\partial_t(\sqrt{g}q) + \partial_x(u\sqrt{g}q) = 0$
- $\sqrt{g}q$ average values update:

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

- Time-averaged flux:

$$F_{i+\frac{1}{2}}^n \approx \frac{1}{\Delta t}\int_{t^n}^{t^{n+1}} (u\sqrt{g}q)(x_{i+\frac{1}{2}}, t)\, dt$$
$$= \frac{1}{\Delta t}\int_{x_{i+\frac{1}{2}}^d(t^n)}^{x_{i+\frac{1}{2}}} (\sqrt{g}q)(x, t^n)\, dx$$

$x_{i+\frac{1}{2}}^d$ is the departure point (DP).

1. DP computation:

$$\begin{cases} \frac{d}{dt}x_{i+\frac{1}{2}}^d(t) = u(x_{i+\frac{1}{2}}^d(t), t), \\ x_{i+\frac{1}{2}}^d(t^{n+1}) = x_{i+\frac{1}{2}}. \end{cases}$$

Piecewise Parabolic Method (PPM)

- $\partial_t(\sqrt{g}q) + \partial_x(u\sqrt{g}q) = 0$
- $\sqrt{g}q$ average values update:

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

- Time-averaged flux:

$$F^n_{i+\frac{1}{2}} \approx \frac{1}{\Delta t}\int_{t^n}^{t^{n+1}} (u\sqrt{g}q)(x_{i+\frac{1}{2}}, t)\, dt$$
$$= \frac{1}{\Delta t}\int_{x^d_{i+\frac{1}{2}}(t^n)}^{x_{i+\frac{1}{2}}} (\sqrt{g}q)(x, t^n)\, dx$$

$x^d_{i+\frac{1}{2}}$ is the departure point (DP).

1. DP computation:

$$\begin{cases} \frac{d}{dt}x^d_{i+\frac{1}{2}}(t) = u(x^d_{i+\frac{1}{2}}(t), t), \\ x^d_{i+\frac{1}{2}}(t^{n+1}) = x_{i+\frac{1}{2}}. \end{cases}$$

2. Piecewise-Parabolic reconstruction of $\sqrt{g}q$ or q .

Piecewise Parabolic Method (PPM)

- $\partial_t(\sqrt{g}q) + \partial_x(u\sqrt{g}q) = 0$
- $\sqrt{g}q$ average values update:

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

- Time-averaged flux:

$$F^n_{i+\frac{1}{2}} \approx \frac{1}{\Delta t}\int_{t^n}^{t^{n+1}} (u\sqrt{g}q)(x_{i+\frac{1}{2}}, t)\, dt = \frac{1}{\Delta t}\int_{x^d_{i+\frac{1}{2}}(t^n)}^{x_{i+\frac{1}{2}}} (\sqrt{g}q)(x, t^n)\, dx$$

$x^d_{i+\frac{1}{2}}$ is the departure point (DP).

1. DP computation:

$$\begin{cases} \frac{d}{dt}x^d_{i+\frac{1}{2}}(t) = u(x^d_{i+\frac{1}{2}}(t), t), \\ x^d_{i+\frac{1}{2}}(t^{n+1}) = x_{i+\frac{1}{2}}. \end{cases}$$

2. Piecewise-Parabolic reconstruction of $\sqrt{g}q$ or q .
3. Integrate the parabolas to obtain the fluxes.

Main references: Carpenter et al. (1990) and Colella and Woodward (1984)

Piecewise Parabolic Method (PPM)

- In particular, we consider Euler method (RK1) and a second-order Runge-Kutta (RK2) to estimate the departure point.

Piecewise Parabolic Method (PPM)

- In particular, we consider Euler method (RK1) and a second-order Runge-Kutta (RK2) to estimate the departure point. RK2 requires linear interpolation in space of $u_{i+\frac{1}{2}}^{n+\frac{1}{2}}$.

Piecewise Parabolic Method (PPM)

- In particular, we consider Euler method (RK1) and a second-order Runge-Kutta (RK2) to estimate the departure point. RK2 requires linear interpolation in space of $u_{i+\frac{1}{2}}^{n+\frac{1}{2}}$.
- Lin (2004) and Putman and Lin (2007) assume a constant metric tensor on the integration domain, resulting in a first-order error:

$$\int_{x_{i+\frac{1}{2}}^{d}(t^n)}^{x_{i+\frac{1}{2}}} (\sqrt{g}q)(x,t^n)\,dx = \sqrt{g_{i+\frac{1}{2}}} \int_{x_{i+\frac{1}{2}}^{d}(t^n)}^{x_{i+\frac{1}{2}}} q(x,t^n)\,dx + O(\Delta x).$$

Piecewise Parabolic Method (PPM)

- In particular, we consider Euler method (RK1) and a second-order Runge-Kutta (RK2) to estimate the departure point. RK2 requires linear interpolation in space of $u_{i+\frac{1}{2}}^{n+\frac{1}{2}}$.
- Lin (2004) and Putman and Lin (2007) assume a constant metric tensor on the integration domain, resulting in a first-order error:

$$\int_{x_{i+\frac{1}{2}}^{d}(t^n)}^{x_{i+\frac{1}{2}}} (\sqrt{g}q)(x, t^n)\, dx = \sqrt{g_{i+\frac{1}{2}}} \int_{x_{i+\frac{1}{2}}^{d}(t^n)}^{x_{i+\frac{1}{2}}} q(x, t^n)\, dx + O(\Delta x).$$

This approach s referred to as **MT-PL07**, which ensures that $\mathbf{F}(Q^n, \tilde{u}) = -\overline{q}\frac{\Delta t}{\Delta x}\delta_i(\sqrt{g}\tilde{u})_{ijp}^{n}$ when $Q^n = \overline{q}$ is constant.

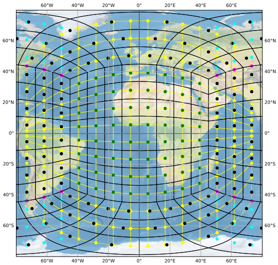
Piecewise Parabolic Method (PPM)

- In particular, we consider Euler method (RK1) and a second-order Runge-Kutta (RK2) to estimate the departure point. RK2 requires linear interpolation in space of $u_{i+\frac{1}{2}}^{n+\frac{1}{2}}$.
- Lin (2004) and Putman and Lin (2007) assume a constant metric tensor on the integration domain, resulting in a first-order error:

$$\int_{x_{i+\frac{1}{2}}^{d}(t^n)}^{x_{i+\frac{1}{2}}} (\sqrt{g}q)(x, t^n)\, dx = \sqrt{g_{i+\frac{1}{2}}} \int_{x_{i+\frac{1}{2}}^{d}(t^n)}^{x_{i+\frac{1}{2}}} q(x, t^n)\, dx + O(\Delta x).$$

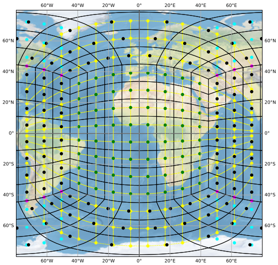
  This approach s referred to as **MT-PL07**, which ensures that $\mathbf{F}(Q^n, \tilde{u}) = -\overline{q}\frac{\Delta t}{\Delta x}\delta_i(\sqrt{g}\tilde{u})_{ijp}^{n}$ when $Q^n = \overline{q}$ is constant.
- This first-order error is eliminated by reconstructing $\sqrt{g}q$ (**MT-0**), but allows splitting error.

Duogrid and mass conservation

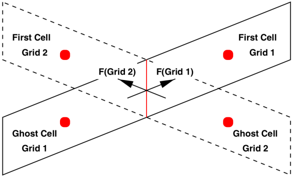


Duogrid (named by Chen (2021))

Duogrid and mass conservation

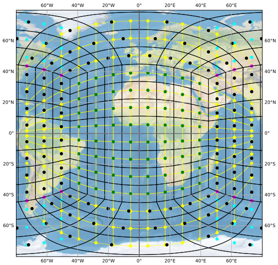


Duogrid (named by Chen (2021))

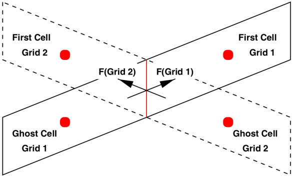


The flux being computed twice on the cube edge breaks the total mass conservation. Figure taken from Rossmanith (2006).

Duogrid and mass conservation



Duogrid (named by Chen (2021))

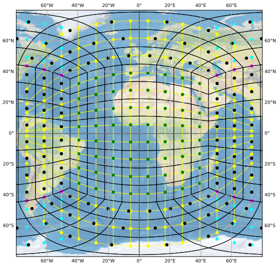


The flux being computed twice on the cube edge breaks the total mass conservation. Figure taken from Rossmanith (2006).

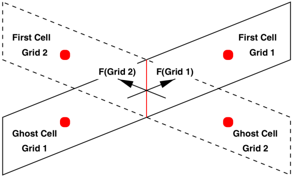
Mass conservation may be achieved by:

1. Average the flux (**AF**) at the cube interfaces (it can be shown that this approach reduces the $\mathbb{D}^n_{ijp}$ accuracy by one only at the cube edges);

Duogrid and mass conservation



Duogrid (named by Chen (2021))



The flux being computed twice on the cube edge breaks the total mass conservation. Figure taken from Rossmanith (2006).

Mass conservation may be achieved by:

1. Average the flux (**AF**) at the cube interfaces (it can be shown that this approach reduces the $\mathbb{D}^n_{ijp}$ accuracy by one only at the cube edges);
2. Compute the L_2 projection (**PR**) of $\mathbb{D}^n_{ijp}$ onto

$$V_0 = \{\delta \in \mathcal{CS}_N : \quad \sum_{p=1}^{6} \sum_{i,j=1}^{N} \delta_{ijp} \sqrt{g}_{ijp} \Delta x \Delta y = 0\}.$$

Numerical results

Scheme	Splitting	DP	ET	Mass fixer	Metric tensor
PL07	PL07	RK1	PL07	none	MT-PL07
PL07-PR	PL07	RK1	Duogrid	PR	MT-PL07
AVLT-AF	AVLT	RK2	Duogrid	AF	MT-0
AVLT-PR	AVLT	RK2	Duogrid	PR	MT-0

Numerical results

Scheme	Splitting	DP	ET	Mass fixer	Metric tensor
PL07	PL07	RK1	PL07	none	MT-PL07
PL07-PR	PL07	RK1	Duogrid	PR	MT-PL07
AVLT-AF	AVLT	RK2	Duogrid	AF	MT-0
AVLT-PR	AVLT	RK2	Duogrid	PR	MT-0

Steady test case from Williamson et al. (1992).

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

where $u_0 = \frac{2\pi}{5}$, $\alpha = -\frac{45\pi}{180}$.

Numerical results

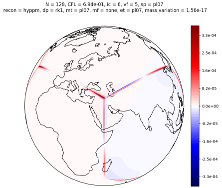
Scheme	Splitting	DP	ET	Mass fixer	Metric tensor
PL07	PL07	RK1	PL07	none	MT-PL07
PL07-PR	PL07	RK1	Duogrid	PR	MT-PL07
AVLT-AF	AVLT	RK2	Duogrid	AF	MT-0
AVLT-PR	AVLT	RK2	Duogrid	PR	MT-0

Steady test case from Williamson et al. (1992).

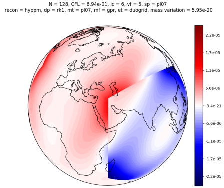
$$\begin{cases} u_\lambda(\lambda,\phi) & = u_0(\cos(\phi)\cos(\alpha)+\sin(\phi)\cos(\lambda)\sin(\alpha)), \\ v_\phi(\lambda,\phi) & = -u_0\sin(\lambda)\sin(\alpha), \end{cases}$$

where $u_0 = \frac{2\pi}{5}$, $\alpha = -\frac{45\pi}{180}$. The advected scalar field is defined in such way that $\nabla \cdot (\boldsymbol{u}q) = 0$.

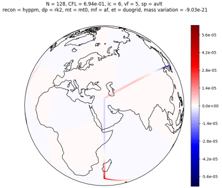
$\nabla \cdot (uq)$ computation



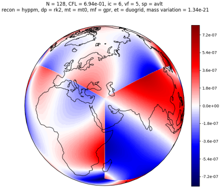
PL07



PL07-PR

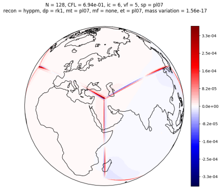


AVLT-AF

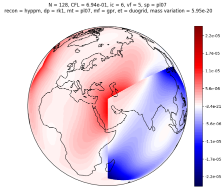


AVLT-PR

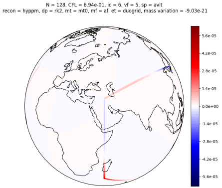
$\nabla \cdot (\boldsymbol{uq})$ computation



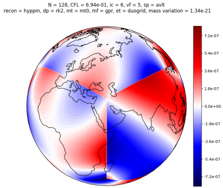
PL07



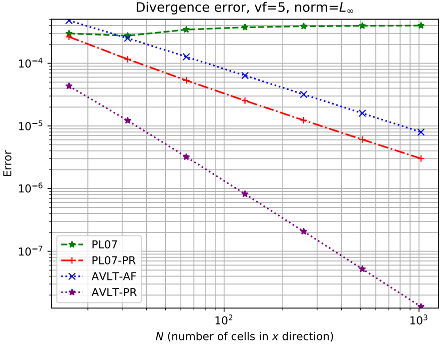
PL07-PR



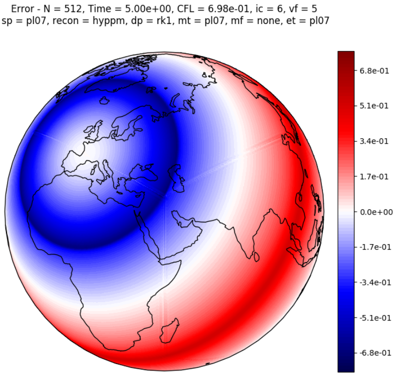
AVLT-AF



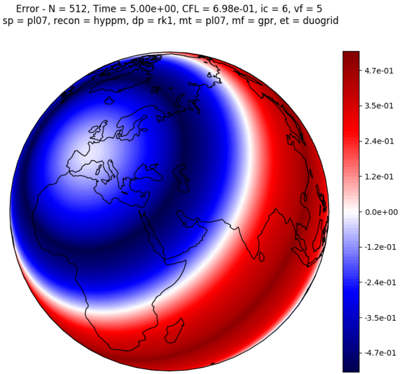
AVLT-PR



Steady test case error

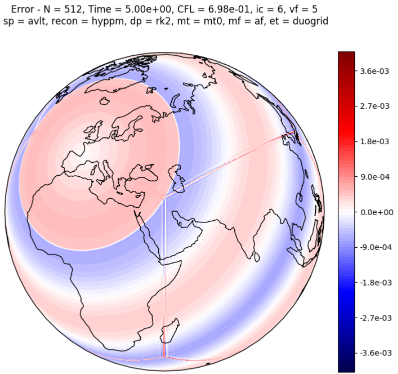


PL07

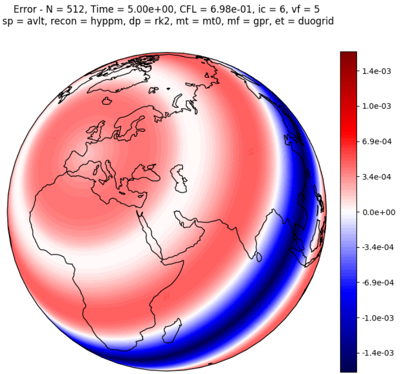


PL07-PR

Steady test case error

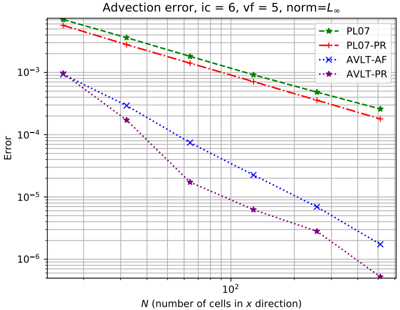


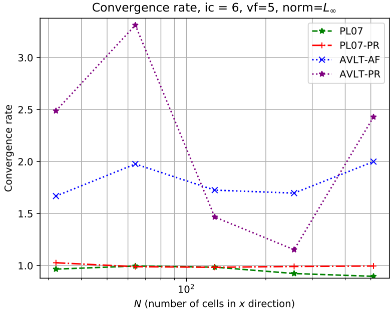
AVLT-AF



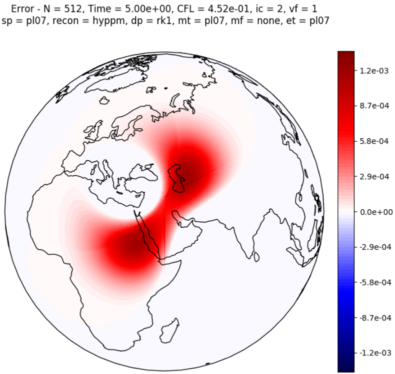
AVLT-PR

Steady test case error

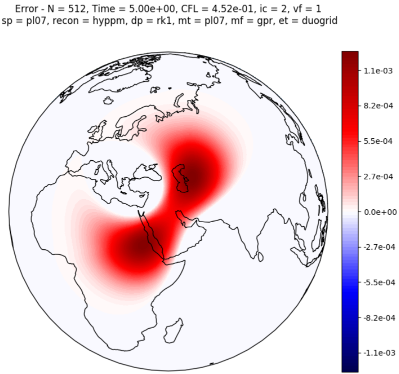




One Gaussian hill advection error

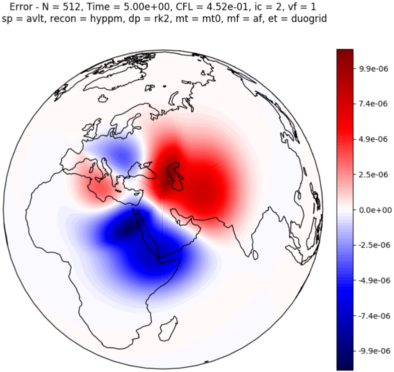


PL07

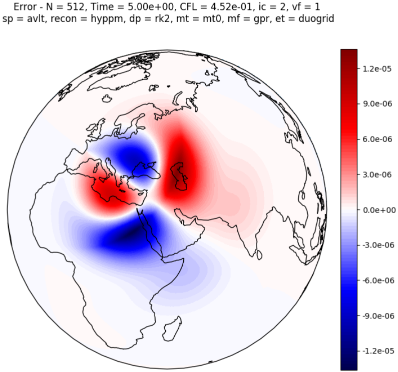


PL07-PR

One Gaussian hill advection error

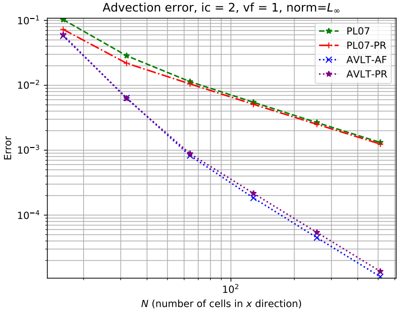


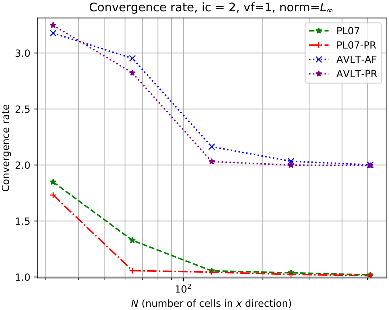
AVLT-AF



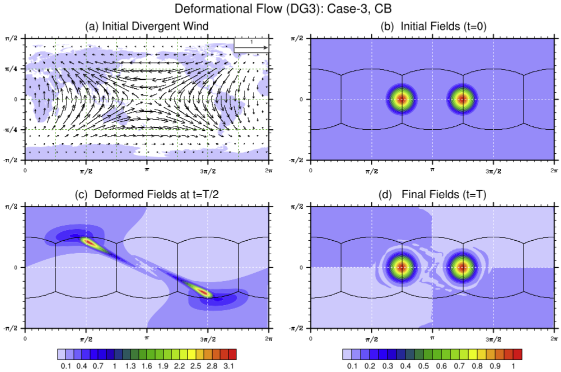
AVLT-PR

One Gaussian hill advection error

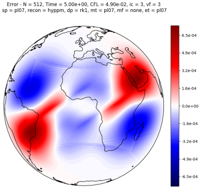




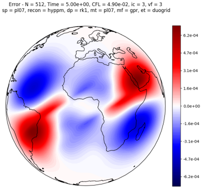
Non-divergent flow from Nair and Lauritzen (2010)



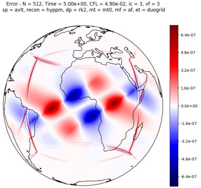
Non-divergent flow from Nair and Lauritzen (2010)



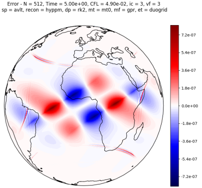
PL07



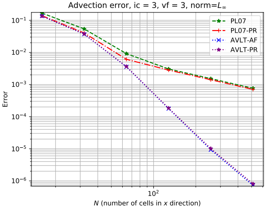
PL07-PR

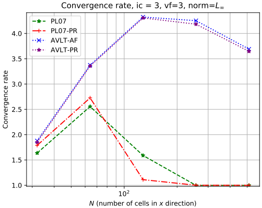


AVLT-AF



AVLT-PR





Conclusions

Take away message

- Duogrid reduces the grid imprinting but we need to use a mass fixer.
- Flux averaging leads to numerical inconsistency and generates grid imprinting.
- Divergence projection preserves the consistency order and is more efficient to reduce grid imprinting.

Conclusions

Take away message

- Duogrid reduces the grid imprinting but we need to use a mass fixer.
- Flux averaging leads to numerical inconsistency and generates grid imprinting.
- Divergence projection preserves the consistency order and is more efficient to reduce grid imprinting.

On going work

- Modify the PL07 splitting to ensure the elimination of splitting errors when employing the most accurate metric tensor treatment (MT-0).
- Propose a local mass fixer that preserves the scheme order.
- Implementation of some of these ideas on the FV3 shallow water core.

References I

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. 24–29).

Chen, X. (2021). The lmars based shallow-water dynamical core on generic gnomonic cubed-sphere geometry [e2020MS002280 2020MS002280]. *Journal of Advances in Modeling Earth Systems*, *13*(1), e2020MS002280. https://doi.org/https://doi.org/10.1029/2020MS002280 (cit. on pp. 35–38).

Colella, P., & Woodward, P. R. (1984). The piecewise parabolic method (ppm) for gas-dynamical simulations. *Journal of Computational Physics*, *54*(1), 174–201. https://doi.org/https://doi.org/10.1016/0021-9991(84)90143-8 (cit. on pp. 24–29).

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. 30–34).

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. 19–23).

References III

- Nair, R. D., & Lauritzen, P. H. (2010). A class of deformational flow test cases for linear transport problems on the sphere. *Journal of Computational Physics*, *229*(23), 8868–8887. https://doi.org/https://doi.org/10.1016/j.jcp.2010.08.014 (cit. on pp. 50, 51).
- Putman, W. M., & Lin, S.-J. (2007). Finite-volume transport on various cubed-sphere grids. *Journal of Computational Physics*, *227*(1), 55–78. https://doi.org/https://doi.org/10.1016/j.jcp.2007.07.022 (cit. on pp. 19–23, 30–34).
- Ronchi, C., Iacono, R., & Paolucci, P. (1996). The “cubed sphere”: A new method for the solution of partial differential equations in spherical geometry. *Journal of Computational Physics*, *124*(1), 93–114. https://doi.org/https://doi.org/10.1006/jcph.1996.0047 (cit. on pp. 4–7).

References IV

Rossmanith, J. A. (2006). A wave propagation method for hyperbolic systems on the sphere. *Journal of Computational Physics*, *213*(2), 629–658. https://doi.org/https://doi.org/10.1016/j.jcp.2005.08.027 (cit. on pp. 35–38).

Williamson, D., Drake, J., Hack, J., Jakob, R., & Swarztrauber, P. (1992). A standard test set for numerical approximations to the shallow water equations in spherical geometry. *J. Comput. Phys.*, *102*, 211–224. https://doi.org/10.1016/S0021-9991(05)80016-6 (cit. on pp. 39–41).

Code availability

- Code: https://github.com/luanfs/cubed-sphere
- Report: https://github.com/luanfs/doc-thesis

Work under progress!

Code availability

- Code: https://github.com/luanfs/cubed-sphere
- Report: https://github.com/luanfs/doc-thesis

Work under progress!

Thank you!

Contact: luan.santos@usp.br

https://luanfs.github.io/