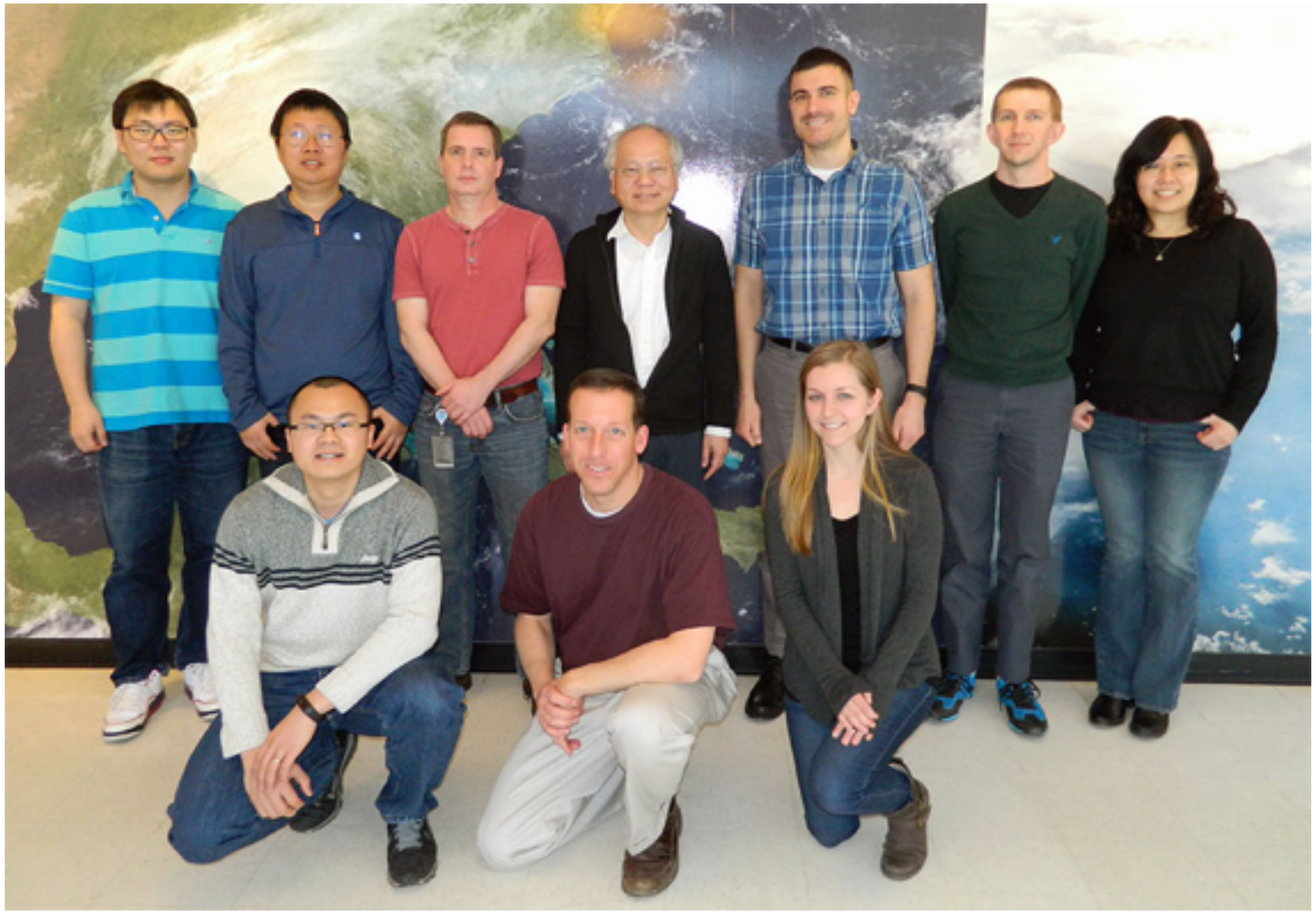
The GFDL Finite-Volume Cubed-sphere Dynamical Core

Lucas Harris, Xi Chen, Shian-Jiann Lin
and the GFDL FV^3 Team
NOAA/Geophysical Fluid Dynamics Laboratory

Dynamical Core Model Intercomparison Project
National Center for Atmospheric Research
Boulder, CO
7 June 2016

The GFDL FV^3 Team

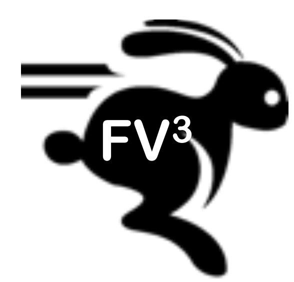
S-J Lin, Team Leader
Rusty Benson, Lead Engineer
Morris Bender NOAA/GFDL
Jan-Huey Chen UCAR
Xi Chen Princeton Univ.
Lucas Harris NOAA/GFDL
Zhi Liang NOAA/GFDL
Tim Marchok NOAA/GFDL
Matt Morin Engility
Bill Putman NASA/GSFC
Shannon Rees Engility
Bill Stern UCAR
Linjiong Zhou Princeton Univ.



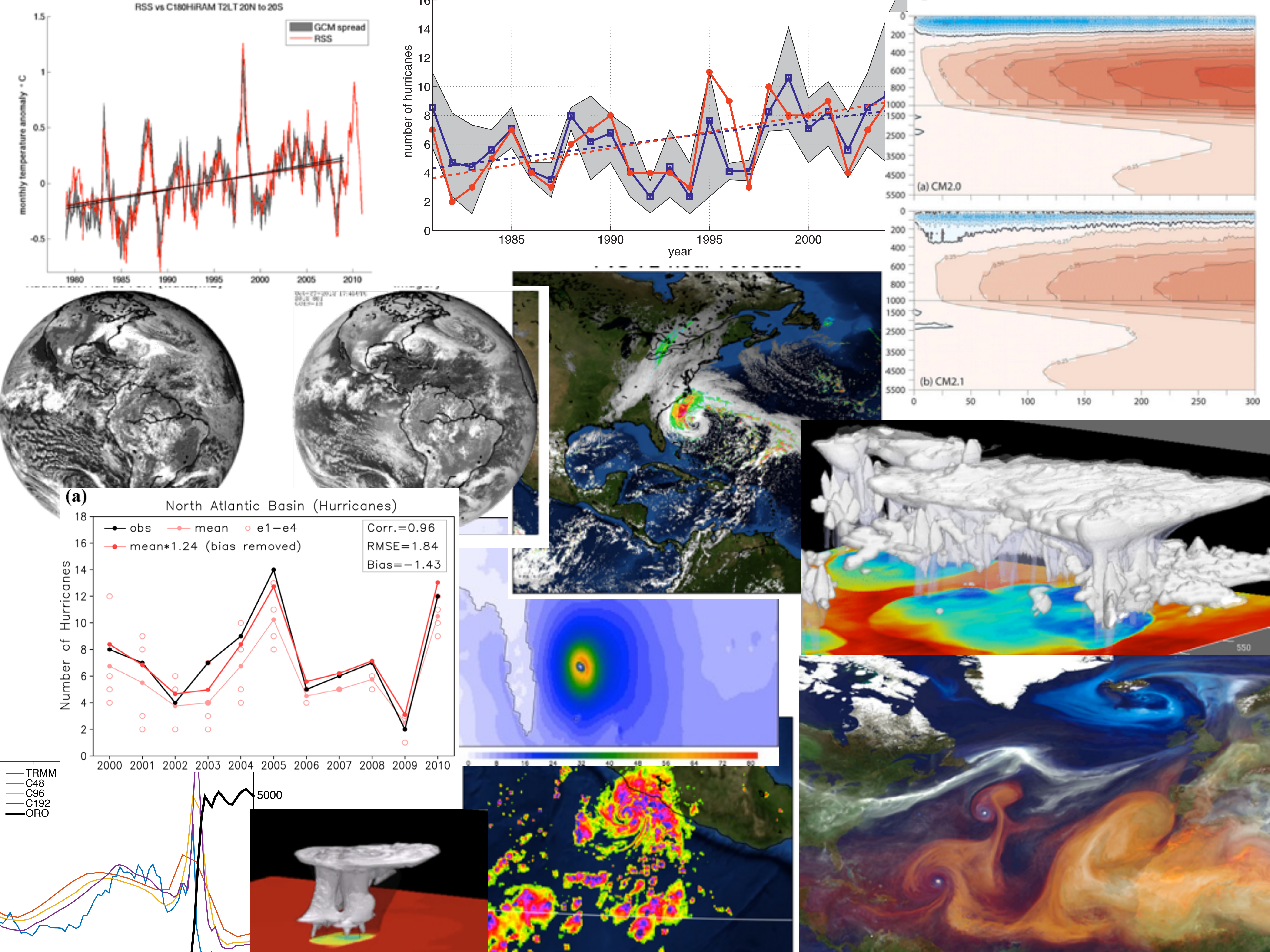
UCAR
UNIVERSITY CORPORATION FOR ATMOSPHERIC RESEARCH
ENGILITY
Engineered to Make a Difference

What is FV^3? FV^3 is:

- **Fully finite volume!** Flux divergences + vertical Lagrangian + integrated PGF
- **Mimetic**: Recovers Newton's and conservation laws with integral theorems
- **Adaptable and Robust**: works with many physics and chemistry packages! AM2/3/4, GOCART, MOZART, CAM, GFS, GEOS, etc.
Also excellent for ocean coupling
- **Flexible**: arbitrary vertical levels, grid refinement by nesting and/or stretching
- **Fast!** A faster model tends to be a better model



- **Proven** effective at all scales. Maintains the large-scale circulation while accurately representing mesoscale and cloud-scale



RSS vs C180HiRAM T2LT 20N to 20S
GCM spread
RSS
monthly temperature anomaly ° C
number of hurricanes
year
(a) CM2.0
(b) CM2.1
(a)
North Atlantic Basin (Hurricanes)
obs
mean
e1−e4
mean*1.24 (bias removed)
Corr.=0.96
RMSE=1.84
Bias=−1.43
Number of Hurricanes
TRMM
C48
C96
C192
ORO
5000

Who uses FV^3?

- FV and FV^3 are among the most widely used global cores in the world, with a large and diverse community of users.

- GFDL models
  - AM4/CM4/ESM4
  - HiRAM
  - CM2.5/2.6
  - FLOR and HiFLOR
  - fvGFS

- CAM-FV^3 (FV is default in CESM)
- LASG FAMIL
- NASA GEOS
- Harvard GEOS-CHEM
- GISS ModelE
- MPI ECHAM (advection scheme)
- JAMSTEC MIROC (adv. scheme)

Development of the FV^3 core

- Lin and Rood (1996, MWR): Flux-form advection scheme
- Lin and Rood (1997, QJ): FV solver
- Lin (1997, QJ): FV Pressure Gradient Force
- Lin (2004, MWR): Vertically-Lagrangian discretization
- Putman and Lin (2007, JCP): Cubed-sphere solver
- Lin (in prep): Nonhydrostatic dynamics
- Harris and Lin (2013) and Harris, Lin, and Tu (2016): Grid refinement

Development of the FV^3 core

- **Lin and Rood (1996, MWR): Flux-form advection scheme**
- Lin and Rood (1997, QJ): FV solver
- Lin (1997, QJ): FV Pressure Gradient Force
- Lin (2004, MWR): Vertically-Lagrangian discretization
- Putman and Lin (2007, JCP): Cubed-sphere solver
- Lin (in prep): Nonhydrostatic dynamics
- Harris and Lin (2013) and Harris, Lin, and Tu (2016): Grid refinement

Lin and Rood (1996, MWR)
Flux-form advection scheme

$$q^{n+1} = \frac{1}{\pi^{n+1}} \left\{ \pi^n q^n + F\left[q^n + \frac{1}{2} g(q^n) \right] + G\left[q^n + \frac{1}{2} f(q^n) \right] \right\}.$$

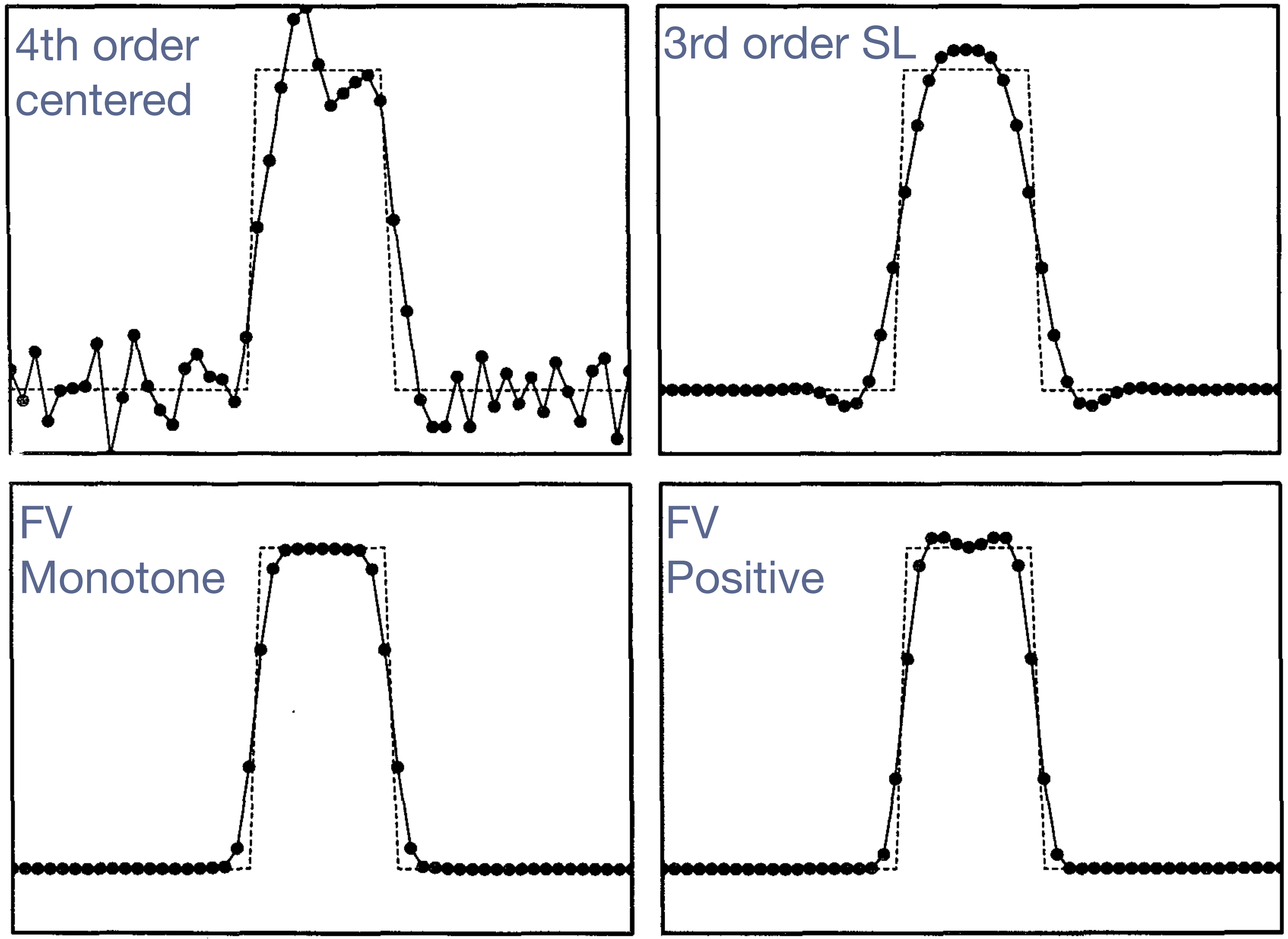
- **Forward-in-time** 2D scheme derived from 1D PPM operators
- Advective-form inner operators (f, g) eliminate leading-order deformation error
    - Allows preservation of constant tracer field under nondivergent flow
    - Ensures forward-in time scheme is stable
    - **Fully 2D**! Stability condition is max(C_x, C_y) < 1
- Flux-form outer operators F, G ensure mass conservation

Lin and Rood (1996, MWR)
Flux-form advection scheme

$$q^{n+1} = \frac{1}{\pi^{n+1}} \left\{ \pi^n q^n + F\left[q^n + \frac{1}{2} g(q^n) \right] + G\left[q^n + \frac{1}{2} f(q^n) \right] \right\}.$$

- PPM operators are upwind biased
  - More physical, but also more diffusive
- Monotonicity constraint to prevent extrema; also option for “linear” (un-limited) non-monotonic scheme. Tracer advection is **always** monotonic.
- Scheme maintains linear correlations between tracers when unlimited or when monotonicity constraint applied (not necessarily so for positivity)

1D Advection Test



Lin and Rood 1996, MWR

Development of the FV^3 core

- Lin and Rood (1996, MWR): Flux-form advection scheme
- **Lin and Rood (1997, QJ): FV solver**
- Lin (1997, QJ): FV Pressure Gradient Force
- Lin (2004, MWR): Vertically-Lagrangian discretization
- Putman and Lin (2007, JCP): Cubed-sphere solver
- Lin (in prep): Nonhydrostatic dynamics
- Harris and Lin (2013) and Harris, Lin, and Tu (2016): Grid refinement

Lin and Rood (1997, QJ) FV solver

- Solves adiabatic layer-averaged vector-invariant equations. δp is the layer mass.

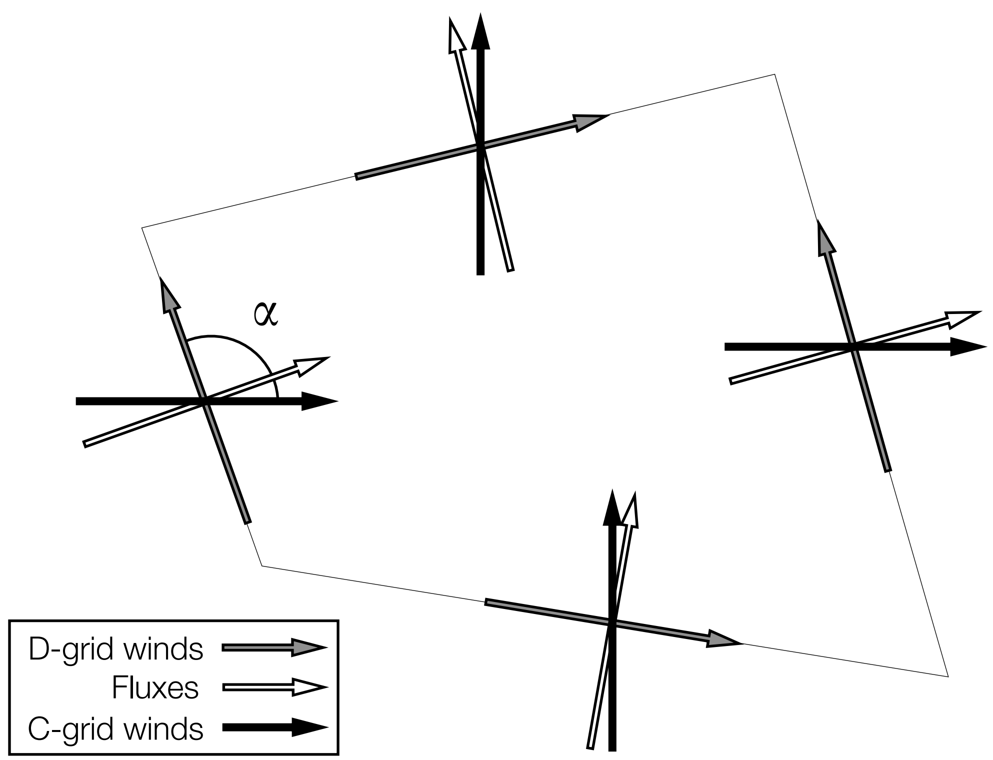
$$\begin{aligned}\frac{\partial \delta p}{\partial t} + \nabla \cdot (\mathbf{V} \delta p) &= 0 \\ \frac{\partial \delta p \Theta}{\partial t} + \nabla \cdot (\mathbf{V} \delta p \Theta) &= 0\end{aligned}$$

- Everything (except the PGF) is a flux! So we use the Lin & Rood advection scheme for forward evaluation.

$$\frac{\partial \mathbf{V}}{\partial t} = -\Omega \hat{k} \times \mathbf{V} - \nabla \left(\kappa + \nu \nabla^2 D\right) - \frac{1}{\rho} \nabla p \Big|_z$$

- PGF evaluated backward with updated pressure and height

- **Question:** how is vertical transport incorporated?

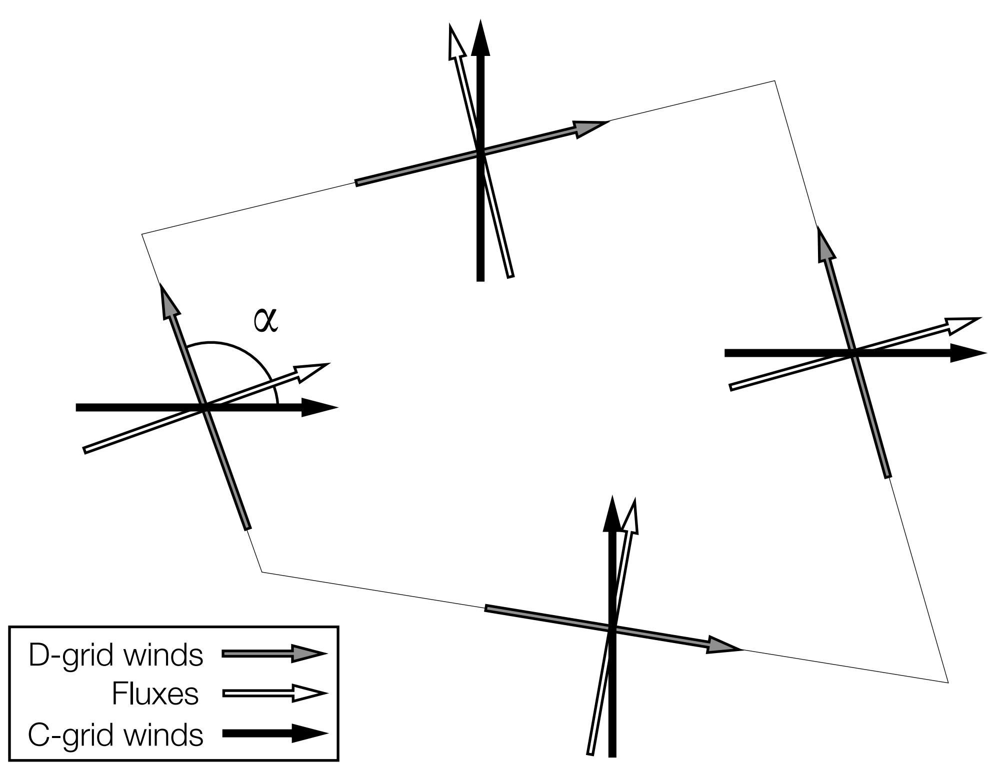


Lin and Rood (1997, QJ) FV solver

- D-grid, with C-grid winds for fluxes
  - C-grid winds advanced a half-timestep—like a simplified Riemann solver. Diffusion due to C-grid averaging is alleviated
- Two-grid discretization and time-centered fluxes avoid computational modes
  - Divergence is invisible to solver: divergence damping is an integral part of the solver

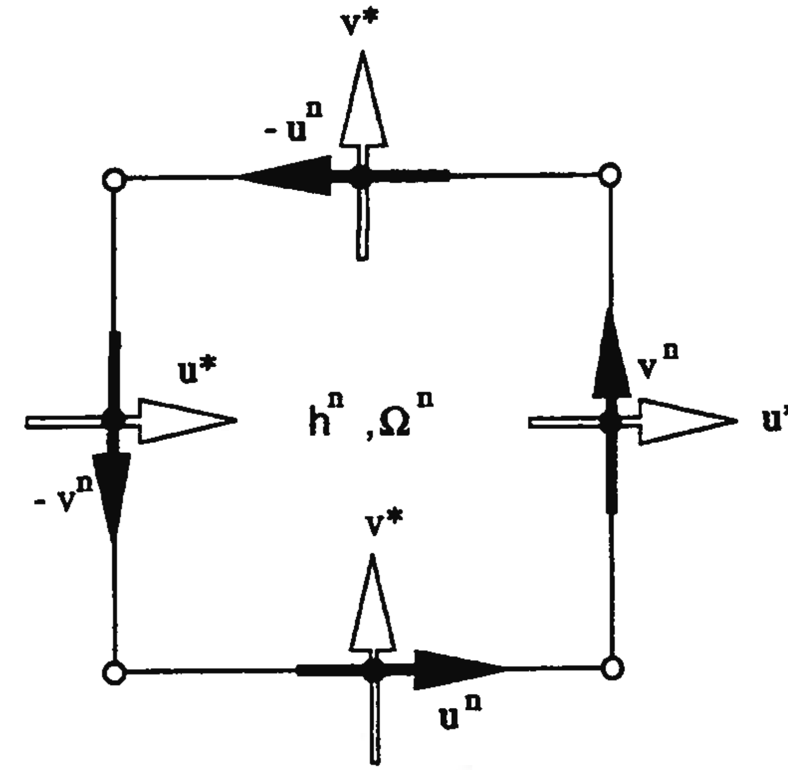
$$\begin{aligned} \frac{\partial \delta p}{\partial t} + \nabla \cdot (\mathbf{V} \delta p) &= 0 \\ \frac{\partial \delta p \Theta}{\partial t} + \nabla \cdot (\mathbf{V} \delta p \Theta) &= 0 \end{aligned}$$

$$\frac{\partial \mathbf{V}}{\partial t} = -\Omega \hat{k} \times \mathbf{V} - \nabla \left(\kappa + \nu \nabla^2 D\right) - \frac{1}{\rho} \nabla p \Big|_z$$



FV solver: Vorticity flux

- Nonlinear vorticity flux term in momentum equation, confounding linear analyses
- D-grid allows exact computation of absolute vorticity—no averaging!
- Vorticity uses same flux as δp: consistency improves geostrophic balance, and SW-PV advected as a scalar!
- **Many** flows are strongly vortical, not just large-scale…



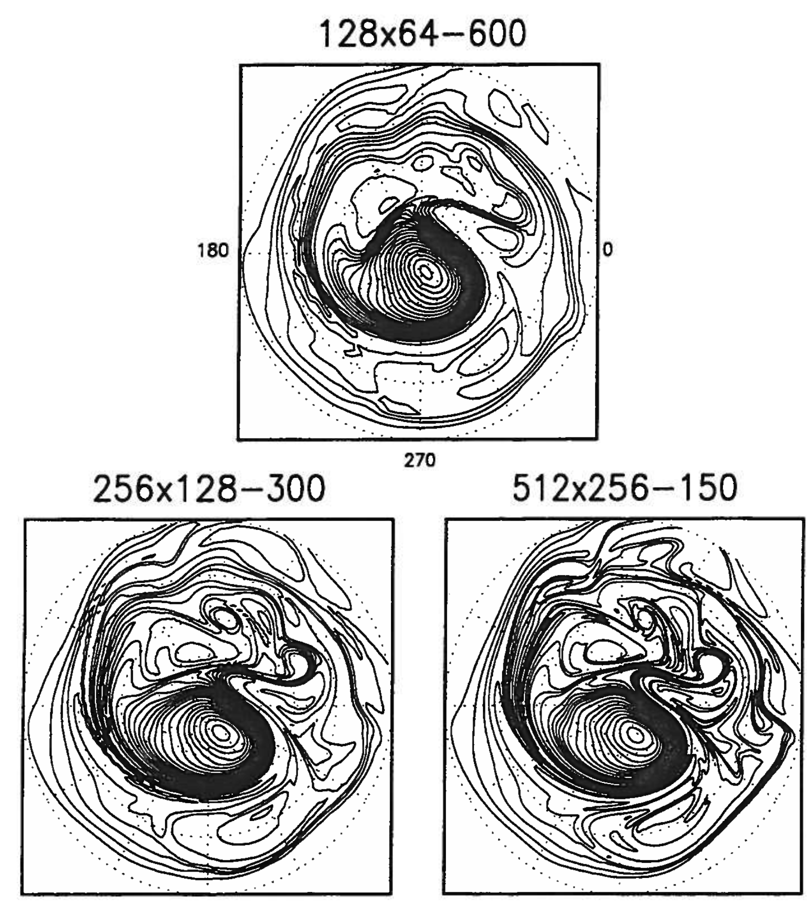


Figure 10. Polar stereographic projection (from the equator to the north pole) of the potential vorticity contours at DAY-24 in the 'stratospheric vortex erosion' test case at three different resolutions.

FV solver:
Kinetic Energy Gradient

- Vector-invariant equations susceptible to Hollingsworth-Kallberg instability if KE gradient not consistent with vorticity flux

- Solution: use C-grid fluxes again to advect wind components, yielding an upstream-biased kinetic energy

$$\kappa^* = \frac{1}{2}\left\{\mathscr{X}(\overline{u^*}^{\theta}, \Delta t; u^n) + \mathscr{Y}(\overline{v^*}^{\lambda}, \Delta t; v^n)\right\}.$$

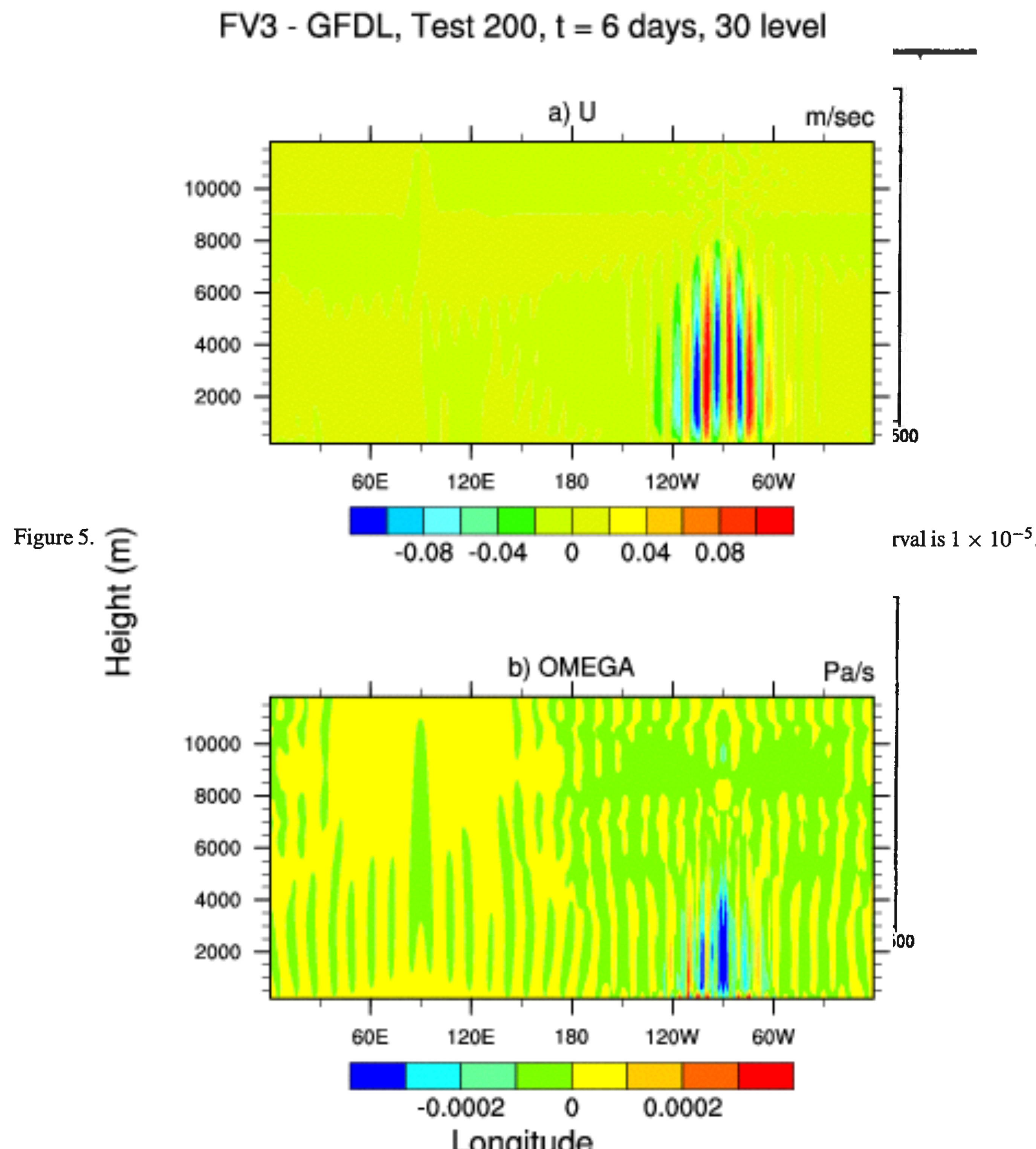
  - Consistent advection again!

Development of the FV^3 core

- Lin and Rood (1996, MWR): Flux-form advection scheme
- Lin and Rood (1997, QJ): FV solver
- **Lin (1997, QJ): FV Pressure Gradient Force**
- Lin (2004, MWR): Vertically-Lagrangian discretization
- Putman and Lin (2007, JCP): Cubed-sphere solver
- Lin (in prep): Nonhydrostatic dynamics
- Harris and Lin (2013) and Harris, Lin, and Tu (2016): Grid refinement

Lin (1997, QJ)
Finite-Volume Pressure Gradient Force

- Computed from Newton's second and third laws, and Green's Theorem
- Errors lower, with much less noise, compared to a finite-difference pressure gradient evaluation
- Easily carries over to nonhydrostatic solver



Development of the FV^3 core

- Lin and Rood (1996, MWR): Flux-form advection scheme
- Lin and Rood (1997, QJ): FV solver
- Lin (1997, QJ): FV Pressure Gradient Force
- **Lin (2004, MWR): Vertically-Lagrangian discretization**
- Putman and Lin (2007, JCP): Cubed-sphere solver
- Lin (in prep): Nonhydrostatic dynamics
- Harris and Lin (2013) and Harris, Lin, and Tu (2016): Grid refinement

Lin (2004, MWR)
Vertically-Lagrangian Discretization

- Equations of motion are vertically integrated to yield a series of layers, which deform freely during the integration

- **Truly Lagrangian!** All flow follows the Lagrangian surfaces, including vertical motion. Vertical transport is *entirely* implicit, so…

  - **No** vertical Courant number restriction!! This is **critical** for high vertical resolution in the boundary layer

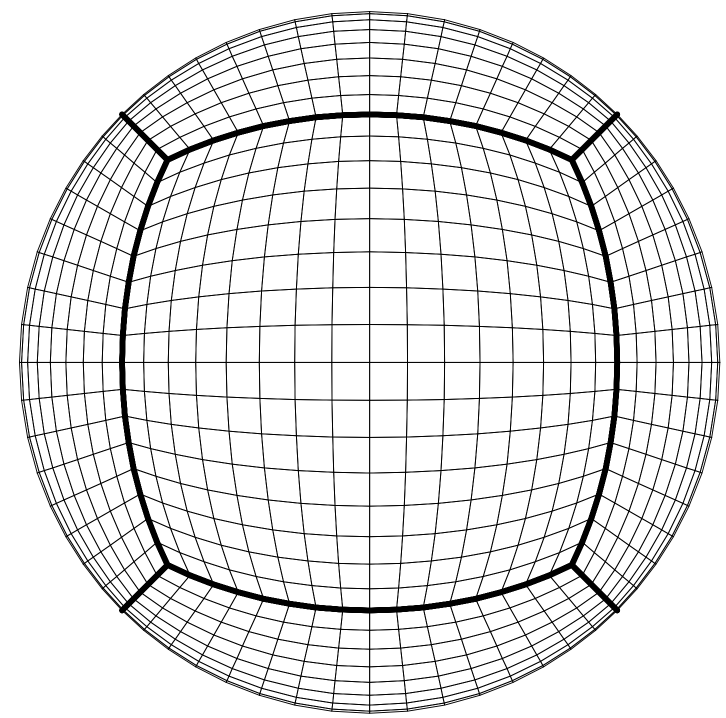
- To avoid layers from becoming infinitesimally thin, vertical remapping to "Eulerian" layers is periodically performed

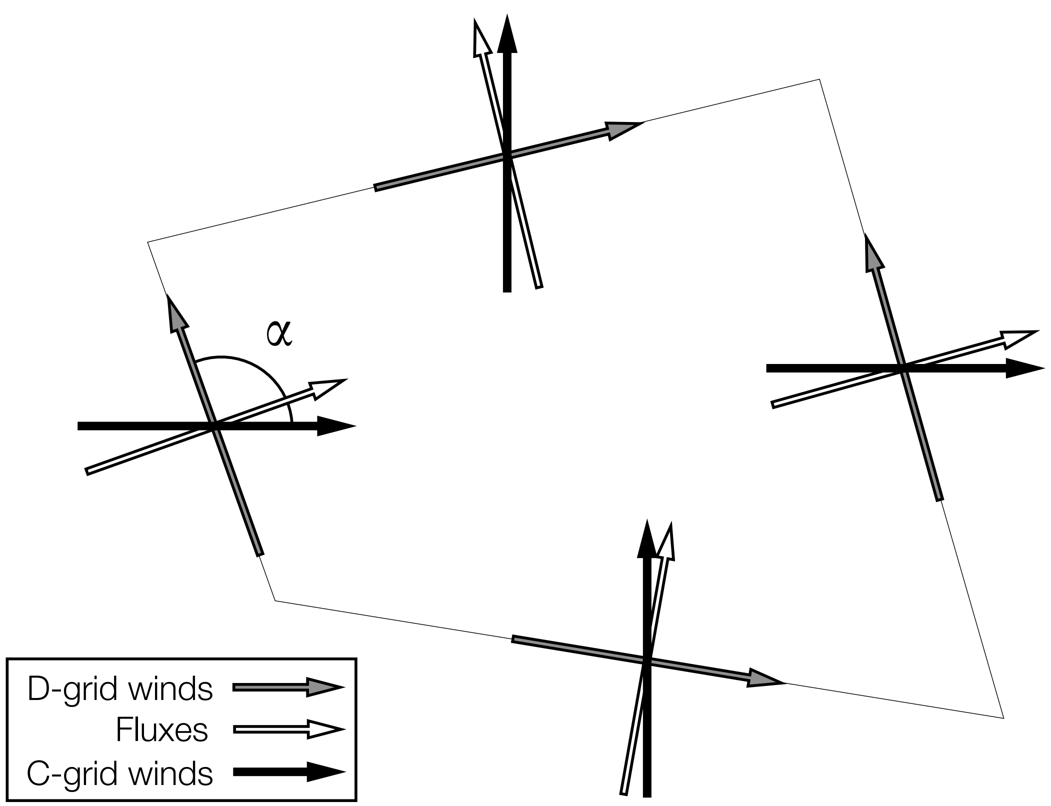
Development of the FV^3 core

- Lin and Rood (1996, MWR): Flux-form advection scheme
- Lin and Rood (1997, QJ): FV solver
- Lin (1997, QJ): FV Pressure Gradient Force
- Lin (2004, MWR): Vertically-Lagrangian discretization
- **Putman and Lin (2007, JCP): Cubed-sphere solver**
- Lin (in prep): Nonhydrostatic dynamics
- Harris and Lin (2013) and Harris, Lin, and Tu (2016): Grid refinement

Putman and Lin (2007, JCP) Cubed-sphere solver

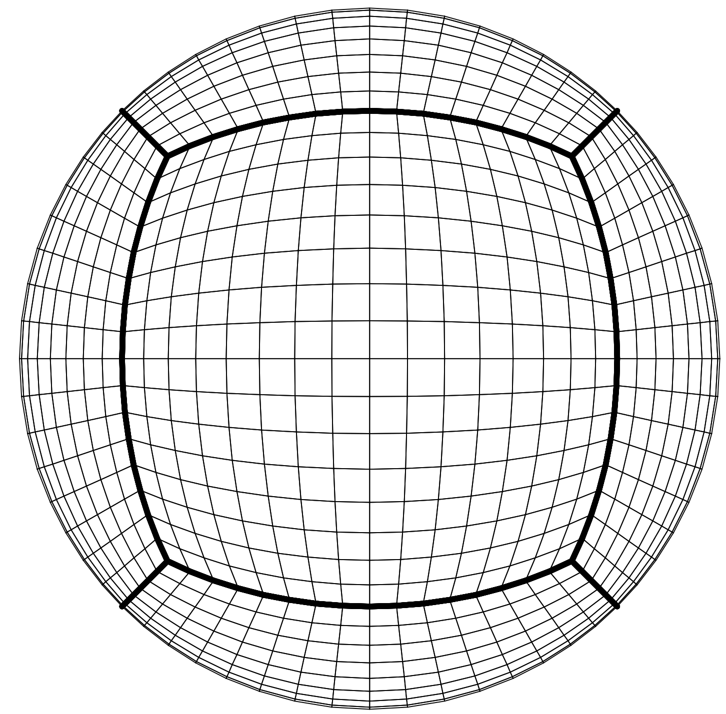
- Gnomonic cubed-sphere grid: coordinates are great circles
- Widest cell only √2 wider than narrowest
    - More uniform than conformal, elliptic, or spring-dynamics cubed spheres
- Tradeoff: coordinate is non-orthogonal, and special handling needs to be done at the edges and corners.

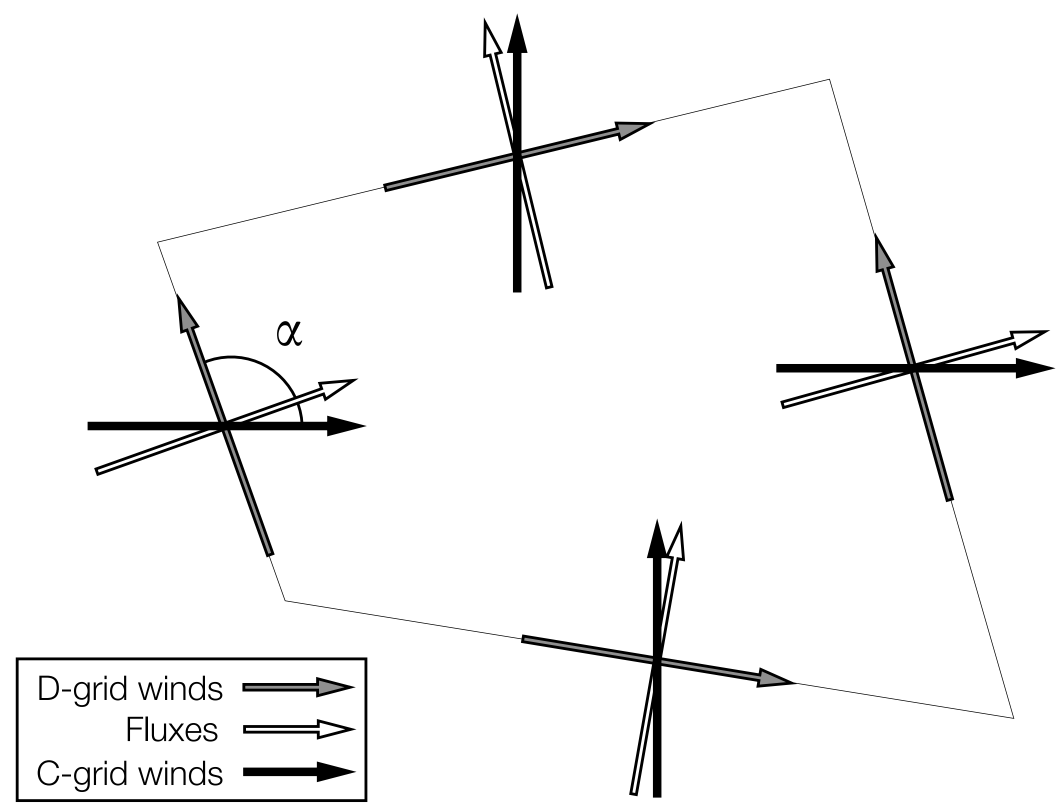




Putman and Lin (2007, JCP)
Non-orthogonal coordinate

- Gnomonic cubed-sphere is non-orthogonal
- Instead of using numerous metric terms, use covariant and contravariant winds
    - Solution winds are covariant, advection is by contravariant winds
    - KE is product of the two





Development of the FV^3 core

- Lin and Rood (1996, MWR): Flux-form advection scheme
- Lin and Rood (1997, QJ): FV solver
- Lin (1997, QJ): FV Pressure Gradient Force
- Lin (2004, MWR): Vertically-Lagrangian discretization
- Putman and Lin (2007, JCP): Cubed-sphere solver
- **Lin (in prep): Nonhydrostatic dynamics**
- Harris and Lin (2013) and Harris, Lin, and Tu (2016): Grid refinement

Nonhydrostatic FV^3

- **Goal**: Maintain hydrostatic circulation, while accurately representing non-hydrostatic motions in the fully-compressible Euler equations

- Introduce new prognostic variables: w and δz (height thickness of a layer), from which density (and thereby nonhydrostatic pressure) is computed

- Traditional semi-implicit solver for handling fast acoustic waves

  - **True nonhydrostatic!** Explicit w into vertically-Lagrangian solver

- Vertical velocity w is the 3D cell-mean value. Vorticity is also a cell-mean value, so **helicity** can be computed without averaging!

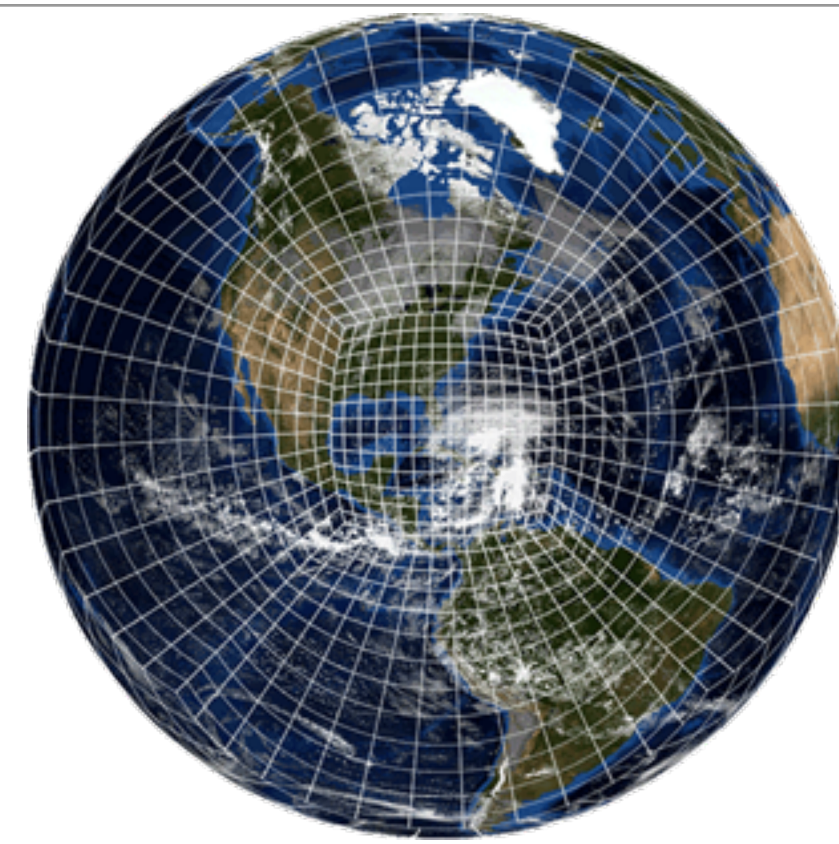
Development of the FV^3 core

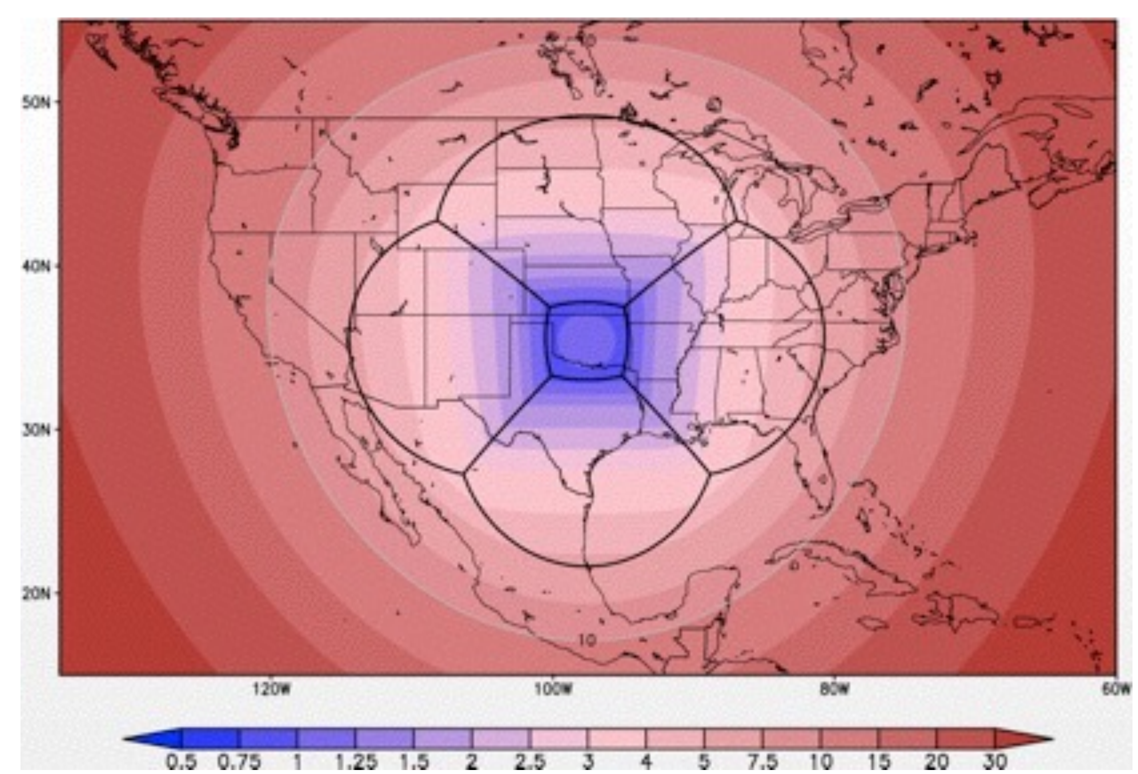
- Lin and Rood (1996, MWR): Flux-form advection scheme
- Lin and Rood (1997, QJ): FV solver
- Lin (1997, QJ): FV Pressure Gradient Force
- Lin (2004, MWR): Vertically-Lagrangian discretization
- Putman and Lin (2007, JCP): Cubed-sphere solver
- Lin (in prep): Nonhydrostatic dynamics
- **Harris and Lin (2013) and Harris, Lin, and Tu (2016): Grid refinement**

Stretched grid

The simple, easy way to achieve grid refinement

- Smooth deformation! And requires no changes to the solver
- Smooth grid has no abrupt discontinuity, and **greatly** reduces need for scale-aware physics
- Capable of **extreme** refinement (80x!!) for easy storm-scale simulations on a full-size earth

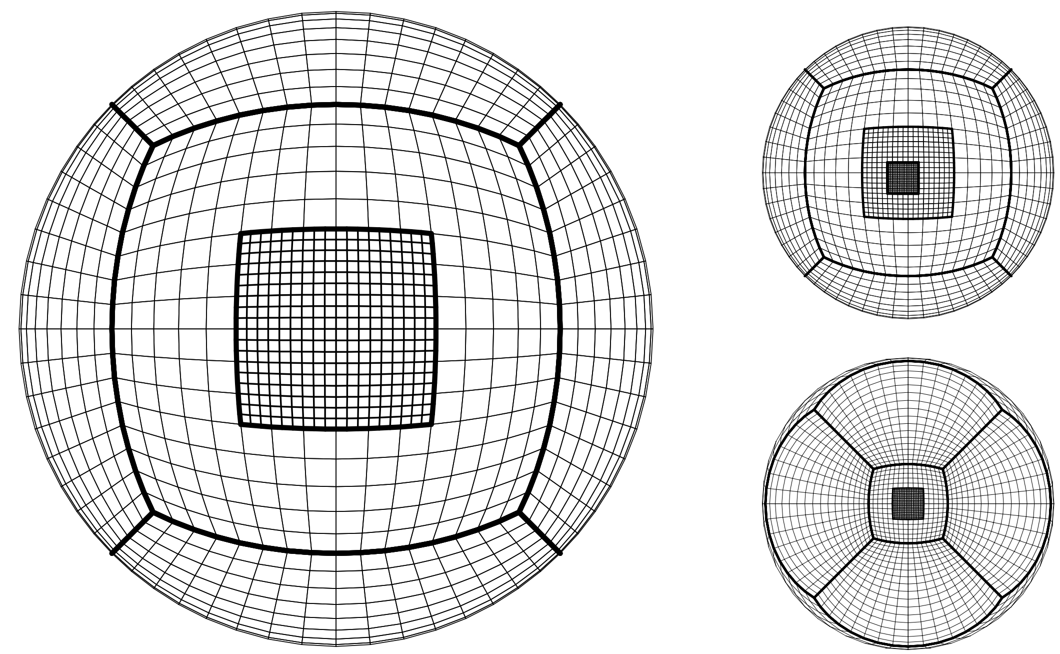


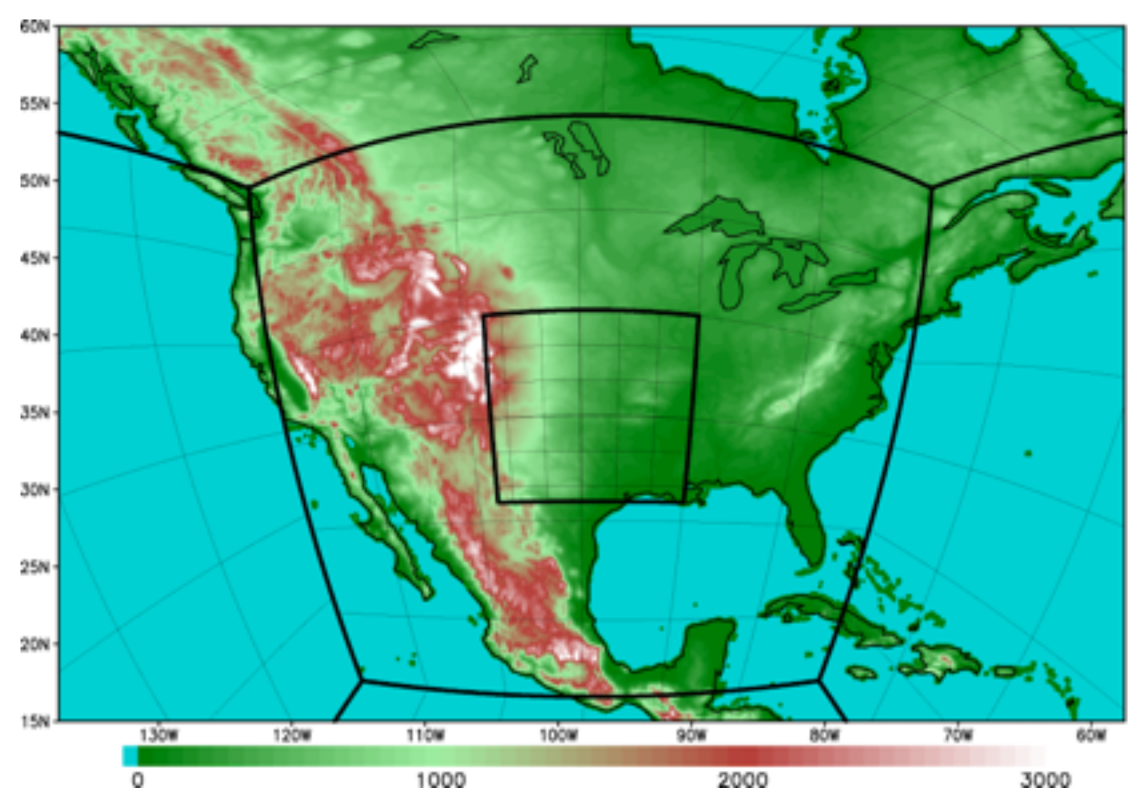


Harris, Lin, and Tu, 2016

Two-way grid nesting

- **Simultaneous** coupled, consistent global and regional solution. No waiting for a regional prediction!
- Different grids permit different parameterizations; **doesn't need a "compromise" or scale-aware physics** for high-resolution region
- Coarse grid can use a longer timestep: **more efficient** than stretching!
- **Very flexible!** Combine with stretching for very high levels of refinement





Harris and Lin, 2013, 2014

FV solver:
Time-stepping procedure

- Interpolate time t^n D-grid winds to C-grid
- Advance C-grid winds by one-half timestep to time $t^{n+1/2}$
- Use time-averaged air mass fluxes to update δp and θ_v to time t^{n+1}
- Compute vorticity flux and KE gradient to update D-grid winds to time t^{n+1}
- Use time t^{n+1} δp and θ_v to compute PGF to complete D-grid wind update

FV^3 **nonhydrostatic** solver: Time-stepping procedure

- Interpolate time t^n D-grid winds to C-grid
- Advance C-grid winds by one-half timestep to time $t^{n+1/2}$
- Use time-averaged air mass fluxes to update δp and θ_v, **and to advect w and δz**, to time t^{n+1}
- Compute vorticity flux and KE gradient to update D-grid winds to time t^{n+1}
- **Solve nonhydrostatic terms for w and nonhydrostatic pressure perturbation using vertical semi-implicit solver**
- Use time t^{n+1} δp, **δz**, and θ_v to compute PGF to complete D-grid wind update

Mass conserving two-way nesting

- Usually quite complicated: requires flux BCs, conserving updates, and precisely-aligned grids
- Update only winds and temperature; not δp, δz, or tracer mass
  - Two-way nesting overspecifies solution anyway
- **Very simple**: works regardless of BC and grid alignment

★ δp is the vertical coordinate: need to remap the nested-grid data to the coarse-grid's vertical coordinate

- Option: "renormalization-conserving" tracer update