The GFDL Finite-Volume Cubed-Sphere Dynamical Core

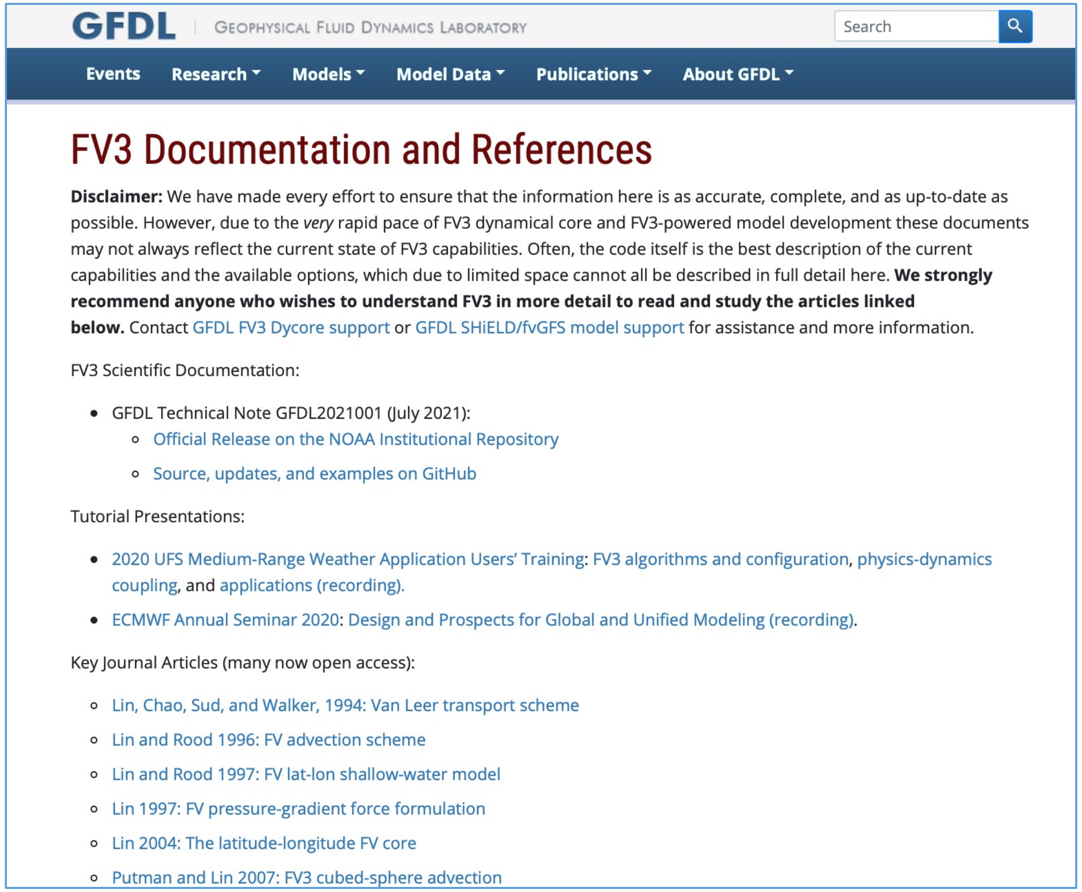
Structure and Usage

Lucas Harris
for the GFDL FV3 Team

UFS SRW Application Training

21 September 2021

FV3 Reference Info



GFDL | Geophysical Fluid Dynamics Laboratory

Search

Events | Research | Models | Model Data | Publications | About GFDL

FV3 Documentation and References

Disclaimer: We have made every effort to ensure that the information here is as accurate, complete, and as up-to-date as possible. However, due to the *very* rapid pace of FV3 dynamical core and FV3-powered model development these documents may not always reflect the current state of FV3 capabilities. Often, the code itself is the best description of the current capabilities and the available options, which due to limited space cannot all be described in full detail here. **We strongly recommend anyone who wishes to understand FV3 in more detail to read and study the articles linked below.** Contact GFDL FV3 Dycore support or GFDL SHiELD/fvGFS model support for assistance and more information.

FV3 Scientific Documentation:

- GFDL Technical Note GFDL2021001 (July 2021):
  - Official Release on the NOAA Institutional Repository
  - Source, updates, and examples on GitHub

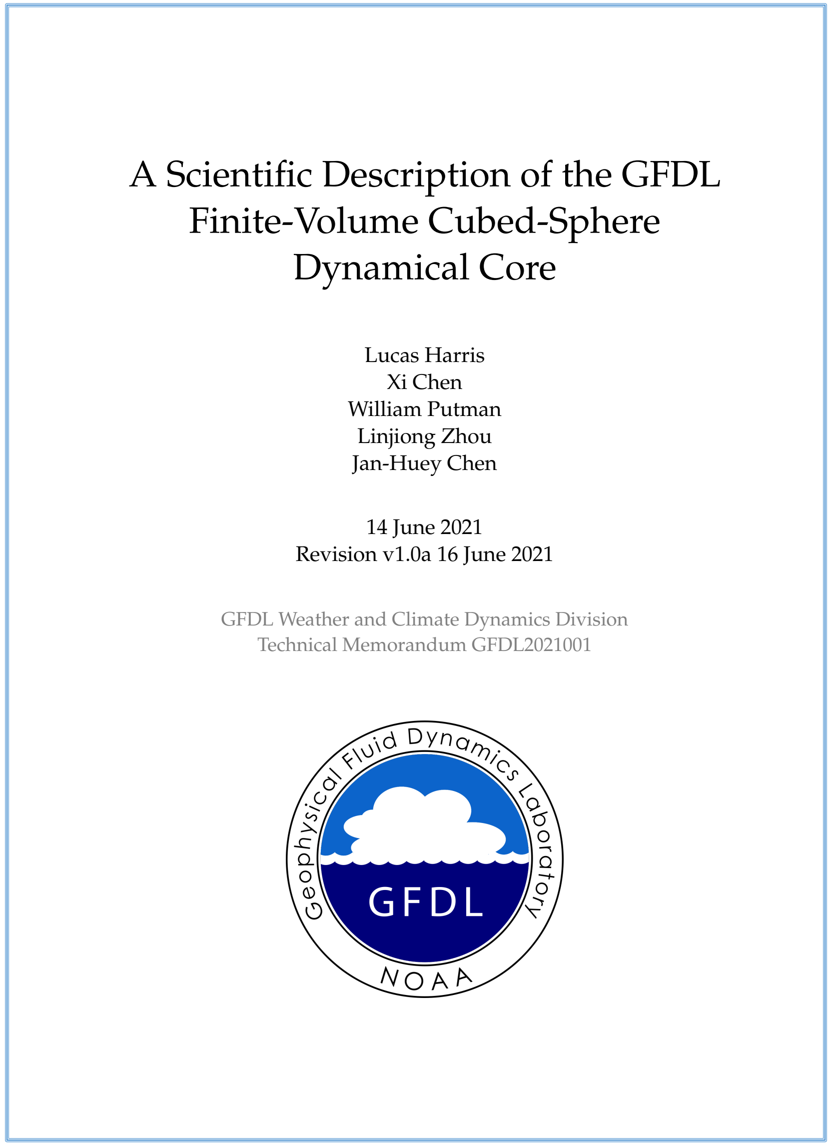
Tutorial Presentations:

- 2020 UFS Medium-Range Weather Application Users' Training: FV3 algorithms and configuration, physics-dynamics coupling, and applications (recording).
- ECMWF Annual Seminar 2020: Design and Prospects for Global and Unified Modeling (recording).

Key Journal Articles (many now open access):

- Lin, Chao, Sud, and Walker, 1994: Van Leer transport scheme
- Lin and Rood 1996: FV advection scheme
- Lin and Rood 1997: FV lat-lon shallow-water model
- Lin 1997: FV pressure-gradient force formulation
- Lin 2004: The latitude-longitude FV core
- Putman and Lin 2007: FV3 cubed-sphere advection

www.gfdl.noaa.gov/fv3/fv3-documentation-and-references/



A Scientific Description of the GFDL Finite-Volume Cubed-Sphere Dynamical Core

Lucas Harris
Xi Chen
William Putman
Linjiong Zhou
Jan-Huey Chen

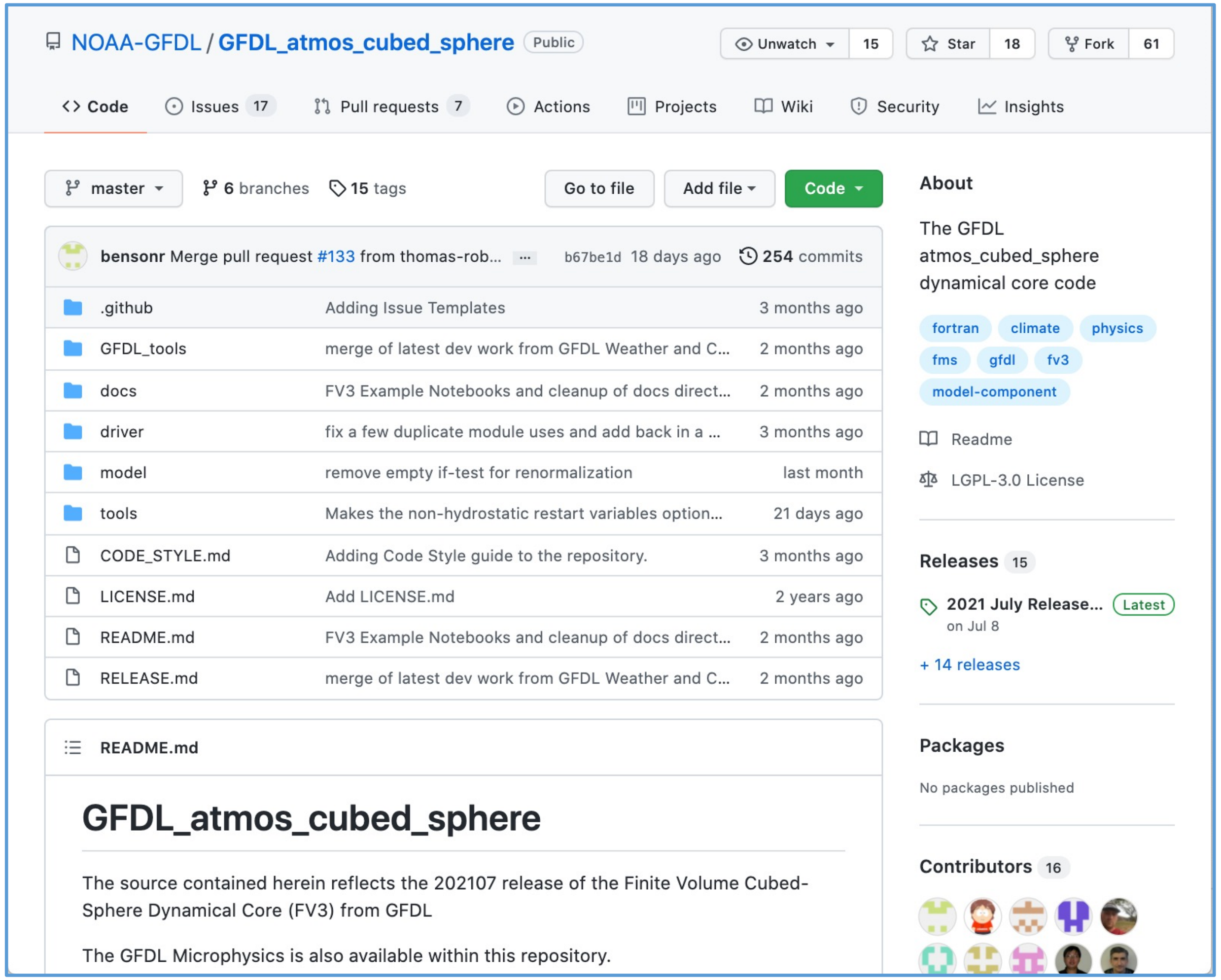
14 June 2021
Revision v1.0a 16 June 2021

GFDL Weather and Climate Dynamics Division
Technical Memorandum GFDL2021001

Harris et al. (2021)
Comprehensive FV3 Scientific Documentation
on GitHub and NOAA Institutional Repository

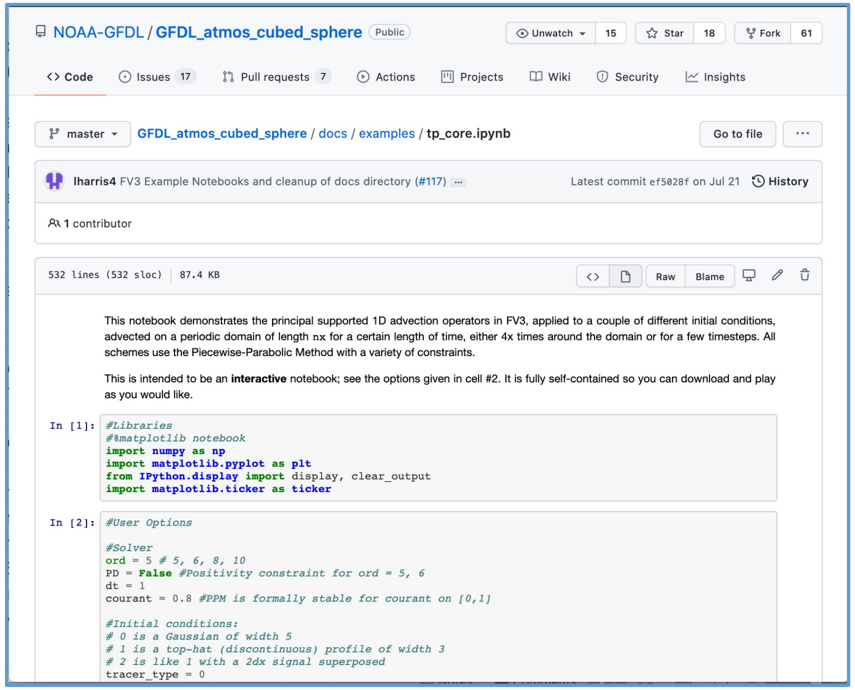
FV3 Community GitHub

Fork FV3 on GitHub



github.com/NOAA-GFDL/GFDL_atmos_cubed_sphere

Official site for FV3 releases, examples, issue tracking, documentation, and more



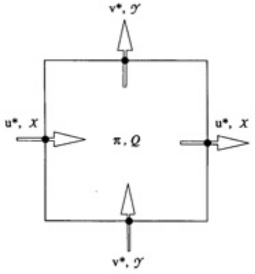
Examples directory: Jupyter notebooks demonstrating FV3 capabilities. Updates released regularly.

FV3: The GFDL Finite-Volume Cubed-Sphere Dynamical Core

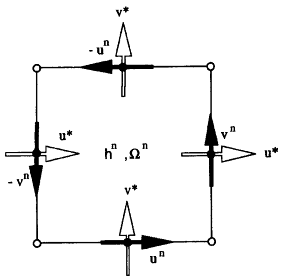
The FV3 Way

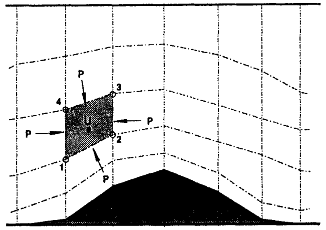
- Physical consistency
- Fully-FV numerics
- Component coupling
- Computational efficiency

Lin & Rood 1996
Efficient 2D high-order conservative FV transport



Lin & Rood 1997
FV horizontal solver focusing on nonlinear vorticity dynamics

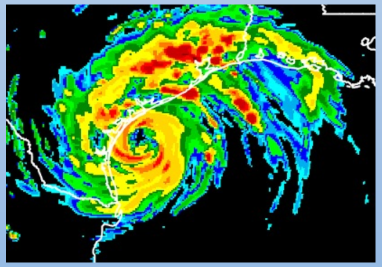


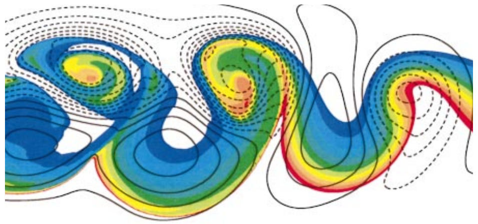


Lin 1997 Efficient, mimetic FV PGF

FV3 for the 2020s

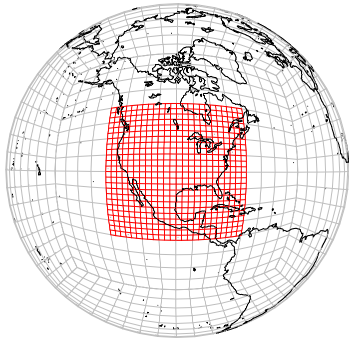
Rigorous Thermodynamics
Flexible dynamics
Adaptable physics interface
Variable-resolution techniques
Regional & periodic domains
Powerful initialization, DA, and nudging functions

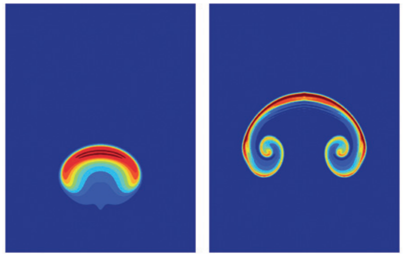




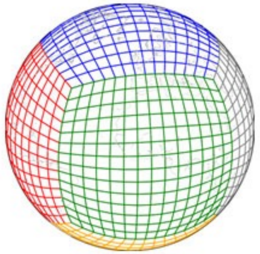
Lin 1998–2004 FV core with “floating” Lagrangian vertical coordinate

Harris & Lin 2013, 2016
Variable resolution with two-way nesting and Schmidt grid stretching





Lin 2006, X Chen & Lin et al 2013
Consistent Lagrangian nonhydrostatic dynamics



Putman & Lin 2007
Scalable cubed-sphere grid, doubly-periodic domain

Finite-Volume Dynamical Cores

FV3 Documentation
Section 1.2

- All variables are 3D cell- or face-means…not gridpoint values
- We solve not the differential Euler equations but their cell-integrated forms using integral theorems
  - Everything is a flux, including the momentum equation. **Fully FV!**
  - Mass conservation ensured to rounding error
  - C-D grid: Vorticity computed *exactly*; accurate divergence computation
  - Mimetic: Physical properties recovered by discretization, particularly Newton's 3rd law
  - Fully compressible: calculation is horizontally local
  - Flow-following Lagrangian vertical coordinate
- FV3 is a fully forward-in-time solver with backwards PGF and acoustic terms

FV3 time integration sequence

- FV3 is a forward-in-time solver with multiple levels of time-integration
  - Flux-divergence terms and physics tendencies evaluated forward-in-time
  - Pressure-gradient and sound-wave terms evaluated backward-in-time for stability
  - HEVI: Everything is *explicit* in the horizontal but *implicit* in the vertical
- *Lagrangian vertical coordinate*: flow constrained along time-evolving Lagrangian surfaces. This greatly simplifies the inner “acoustic” or “Lagrangian dynamics” timestep.

dt_atmos
Physics timestep

fv_dynamics()
FV3 solver

[physics]

fv_update_phys()
Consistent field update

k_split
“remapping” loop

dyn_core()
Lagrangian dynamics

fv_tracer2d()
Sub-cycled tracer transport
OpenMP on k

Lagrangian_to_Eulerian()
Vertical Remapping
GFDL In-line Microphysics
(i,k) OpenMP on j

n_split
“acoustic” loop

c_sw(), etc.
C-grid solver

d_sw()
Forward Lagrangian dyn.
OpenMP on k

update_dz_d()
Forward δz evaluation
OpenMP on k

riem_solver()
Backwards vertical PGF,
sound wave processes
(i,k) OpenMP on j

one_grad_p()/nh_p_grad()
Backwards horizontal PGF
OpenMP on k

dt_atmos	Physics Timestep fv_dynamics calling frequency
dt_atmos ÷ k_split	Tracer calling timestep Vertical remapping In-line microphysics
dt_atmos ÷ (k_split x n_split)	Gravity and acoustic processes

FV3 Documentation

Chapter 2

Time integration: Namelist Options

- `dt_atmos`: Timestep for the full FV3 solver and physics.
  - Should be motivated by physics design.
    For GFS Physics recommend 150–225; smaller for regional physics
- `k_split`: Number of vertical remappings per long timestep.
  - More `k_splits` can improve stability but slow down model
- `n_split`: Number of acoustic timesteps per remapping timestep.
  - Recommend values between 5–10.
  - The acoustic timestep is equal to `dt_atmos` / (`k_split` x `n_split`)
- `hydrostatic`: whether to use the hydrostatic solver

The Cubed-Sphere Grid

The 3 in FV3

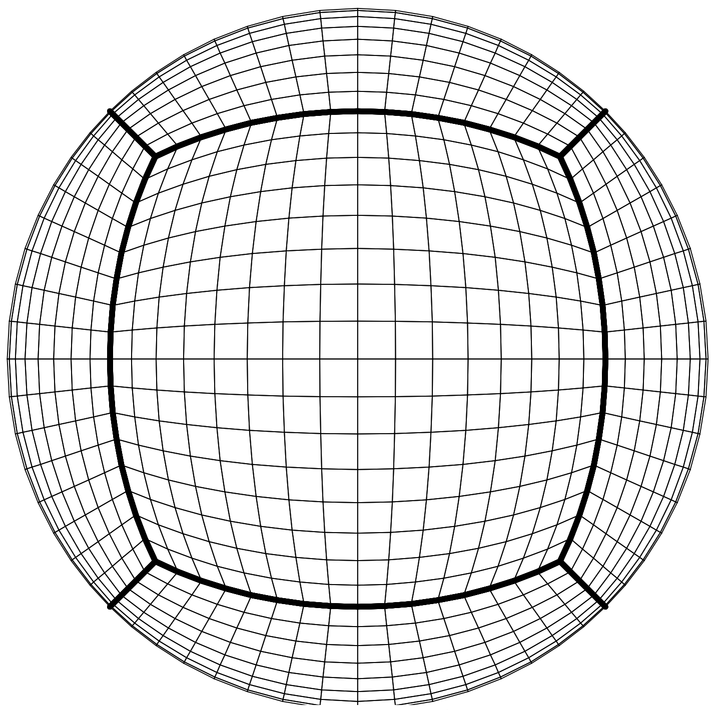
- Gnomonic cubed-sphere grid: coordinates are great circles but non-orthogonal
  - Solution winds are covariant, advection is by contravariant winds
- Winds u and v are defined internally in the local coordinate; output is *always* rotated to earth-relative coords
- Special handling at edges and corners

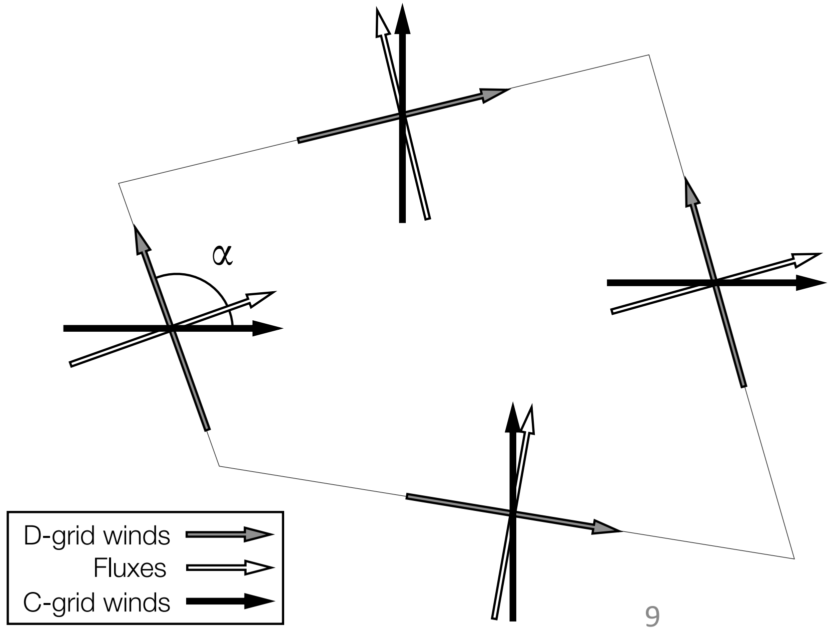
FV3 Documentation

Chapter 3

Literature

Putman + Lin 2007





Grid Namelist Options

- `npx`, `npy`: Number of grid *corners* in each direction.
  - A global cubed sphere **must** use the same in both directions.
  - Nested, regional, or doubly-periodic domains **do not**.
- `ntiles`: Number of tiles on a domain.
  - For the cubed-sphere this must be 6
  - For regional and nested domains this should be 1
- `layout`: 2-element array for the number of MPI domain partitions in each direction on each tile.
  - These values should be divisors of (`npx` - 1) and (`npy` - 1)
  - Total number of cores is `layout`(1) x `layout`(2) x `ntiles`
- `grid_type`: -1 to read from disk, 0 for on-line gnomonic grid (global, regional, nested), 4 for doubly-periodic cartesian domain

FV Advection

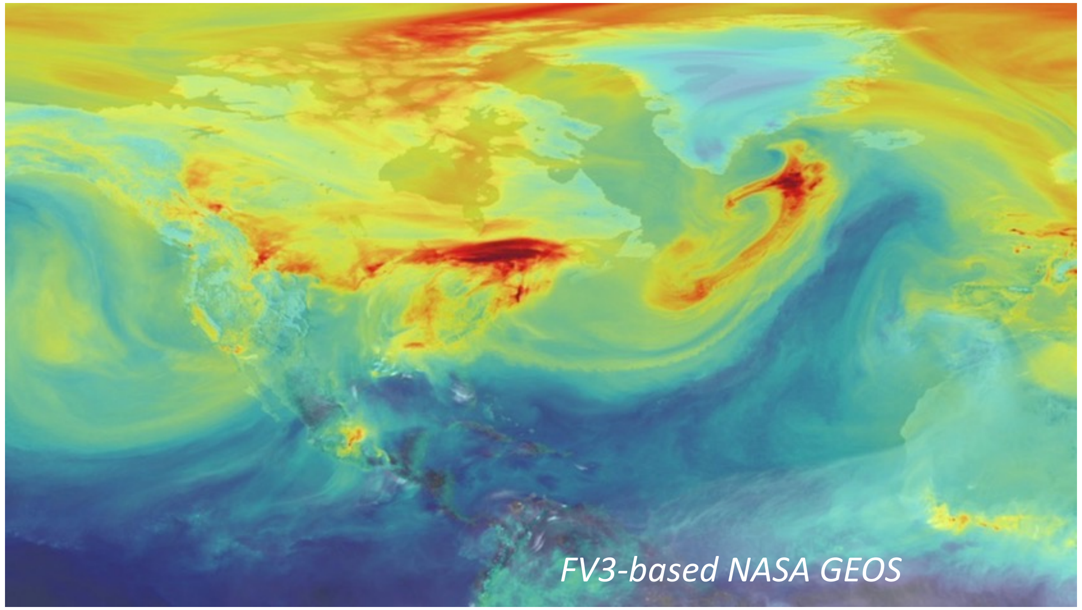
- "Reverse-engineered" forward-in-time 2D scheme constructed from 1D Piecewise-Parabolic Method (PPM) operators
  - Mass-conservative
  - Correlation-preserving for monotonic limiter
  - Cancels splitting error
  - Separate Courant number limit in x and y
  - Upwinding preserves hyperbolicity and causality
- Tracers are advected with a longer, adaptive timestep using the accumulated mass fluxes
- *All* quasi-horizontal processes, except PGF, can be represented as advection
- Highly adaptable: Positive-definite tracer advection greatly improves hurricane structure

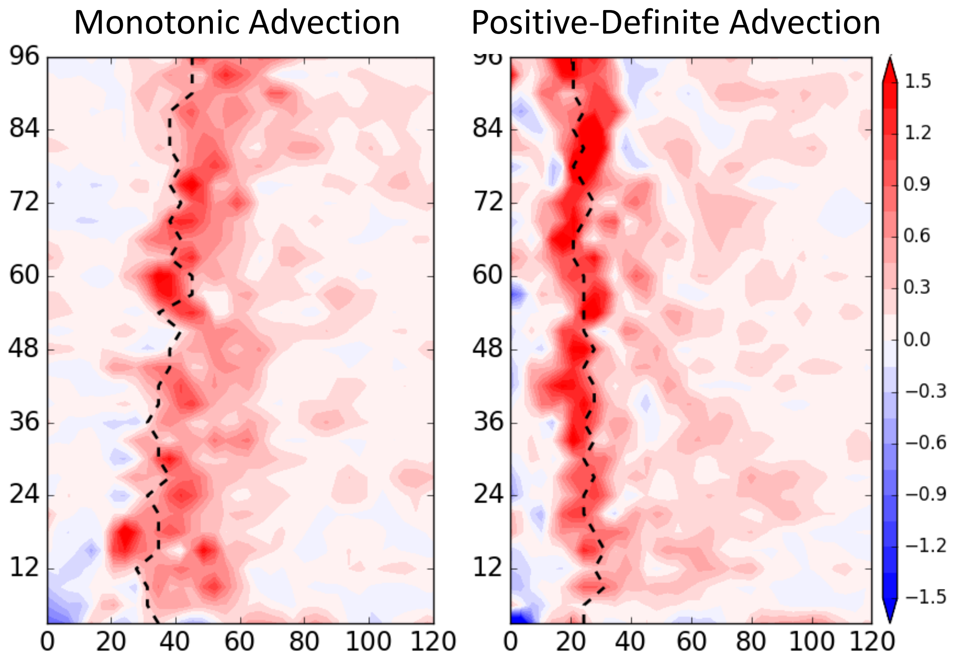
FV3 Documentation

Chapter 4

Literature

Lin and Rood 1996
Putman and Lin 2007





Axisymmetric 5-km W in Hurricane Irma
Gao et al. 2021, JAS

Advection Schemes: Namelist Options

`hord_mt`	KE gradient term
`hord_vt`	Vorticity and w fluxes
`hord_tm`	Potential temperature
`hord_dp`	δp, δz
`hord_tr`	Tracers

Strongly recommend `hord_mt`, `hord_vt`, and `hord_tm` use the same scheme

`hord_tr` *must* use a monotone (8, 10) or positive-definite scheme (-5)

`hord`	
5	"Virtually-inviscid" unlimited "fifth-order" scheme with weak 2Δx filter; fastest and least diffusive
6	Minimally-diffusive unlimited scheme with intermediate-strength 2Δx filter.
8	Monotonic Lin 2004 PPM constraint
10	Hunyh constraint: more expensive but less diffusive than #8
-5	#5 with a positive-definite constraint

FV3 Documentation

Section 4.1

Notebooks

RHwave, tp_core

Advection scheme recommendations

500 m–25 km grid-spacing

- Dynamical quantities (`hord_mt, vt, tm, dp`)
  - “Virtually-inviscid” 5 : Better small-scale structure, convection can develop very rapidly. (Best for short-term forecasting?)
  - “Minimally-diffusive” 6 : Better larger-scale skill, more diffusive though
  - Monotonic (8, 10) more suited for climate or Implicit LES
- Tracer advection (`hord_tr`):
  - **Recommend** positive-definite -5 scheme for weather prediction
  - 8 and 10 preserve correlations exactly and may be better for chemistry applications

Literature
Gao et al. 2021

Notebook
RHwave.ipynb

Lagrangian Dynamics in FV3

$$\frac{\partial \delta p^*}{\partial t} + \nabla \cdot (\mathbf{V} \delta p^*) = 0$$
$$\frac{\partial \Theta_v \delta p^*}{\partial t} + \nabla \cdot (\mathbf{V} \delta p^* \Theta_v) = 0$$
$$\frac{\partial w \delta p^*}{\partial t} + \nabla \cdot (\mathbf{V} \delta p^* w) = -\delta p'$$
$$\frac{\partial u}{\partial t} = \Omega v - \frac{\partial}{\partial x}\mathcal{K} - \frac{1}{\rho}\frac{\partial p}{\partial x}\bigg|_z$$
$$\frac{\partial v}{\partial t} = -\Omega u - \frac{\partial}{\partial y}\mathcal{K} - \frac{1}{\rho}\frac{\partial p}{\partial y}\bigg|_z$$

$$\frac{Dz}{Dt} = w = \frac{\partial z}{\partial t} + \mathbf{V} \cdot \nabla z,$$

- FV3 transforms the Euler equations of motion into a *Lagrangian vertical coordinate*, constraining the flow along quasi-horizontal surfaces
- Lagrangian surfaces deform during the integration. Vertical motion and advection is “free”
- Requires layer thickness δp (and δz for nonhydro) to be a prognostic variable

FV3 Documentation	Literature
Chapter 5	Lin 2004

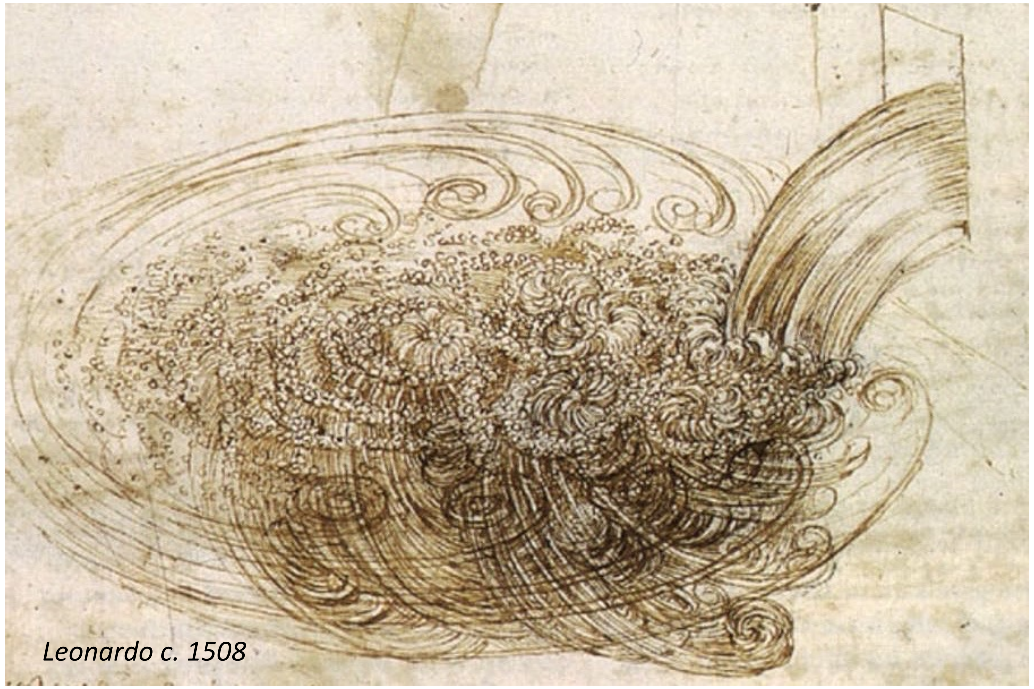
Prognostic Variables

δp	Total air mass (including vapor and condensates) Equal to *hydrostatic* pressure depth of layer
θ_v	Virtual potential temperature
u, v	Horizontal D-grid winds in local coordinate (defined on cell faces)
w	Vertical winds (nonhydrostatic)
δz	Geometric layer depth (nonhydrostatic)
q_i	Passive tracers

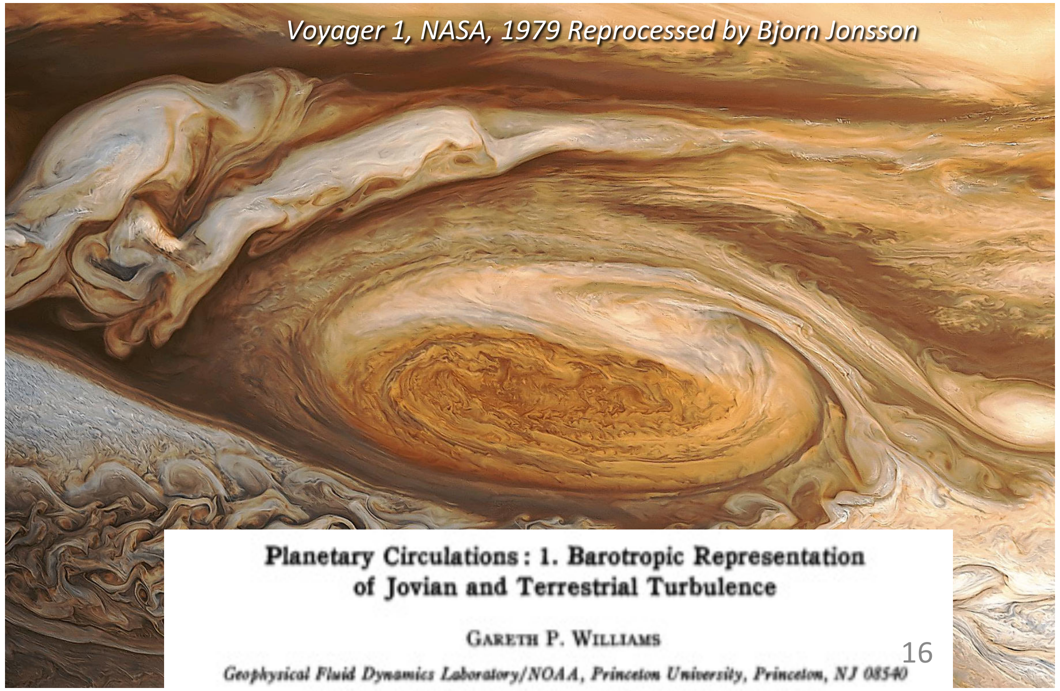
Cell-mean pressure, density, divergence, and specific heat are all *diagnostic* quantities
All variables are layer-means in the vertical: **No vertical staggering**

Vorticity Dynamics

- *Fluids are strongly vortical at all scales.* Vortical motions are especially critical in geophysical flows
- FV3's discretization emphasizes vorticity dynamics:
  - Vector-invariant equations: vorticity computed *exactly*
  - C-D Grid Discretization
  - Consistent advection of derived vorticial quantities



Leonardo c. 1508



Voyager 1, NASA, 1979 Reprocessed by Bjorn Jonsson

Planetary Circulations: 1. Barotropic Representation of Jovian and Terrestrial Turbulence

Gareth P. Williams

Geophysical Fluid Dynamics Laboratory/NOAA, Princeton University, Princeton, NJ 08540

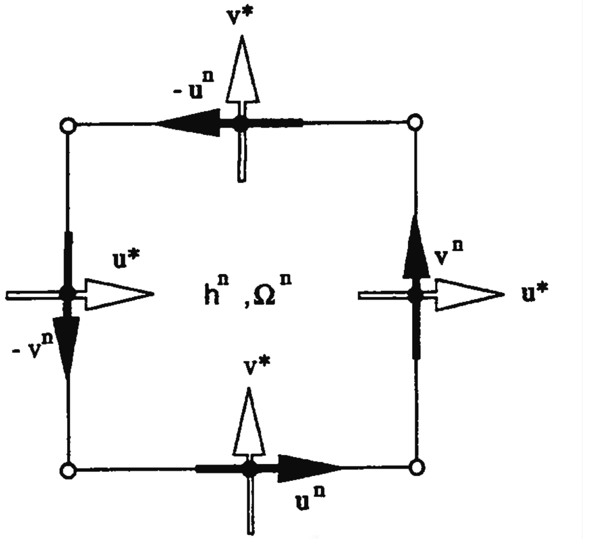
(Manuscript received 2 December 1975, in final form 13 April 1978)

FV3 Documentation	Literature	Notebooks
Chapter 6	Lin et al. 2017	BTwave, BCwave, TornadicSupercell

Momentum equation

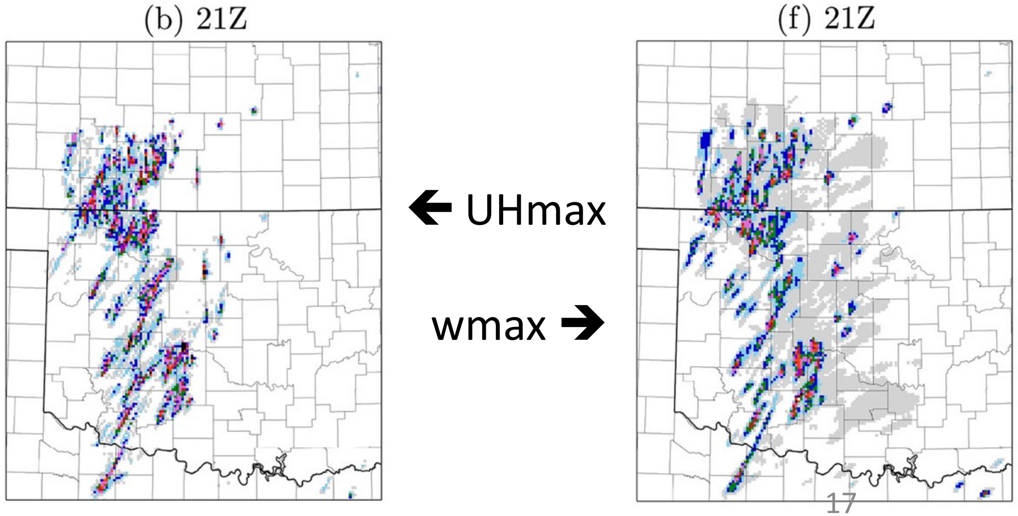
$$\frac{\partial \mathbf{V}}{\partial t} = -\Omega \hat{k} \times \mathbf{V} - \nabla \left(\kappa + \nu \nabla^2 D\right) - \frac{1}{\rho} \nabla p \Big|_z$$

- FV3 solves the nonlinear flux-form vector invariant equations. One of the terms is the absolute vorticity flux Ωv, -Ωu
- D-grid allows *exact* computation of absolute vorticity Ω using Stokes' theorem—no averaging!
- The cell-integrated vorticity is advected as a scalar, using the same fluxes as other variables. Products are also advected as scalars!
  - ex: Updraft helicity wΩ



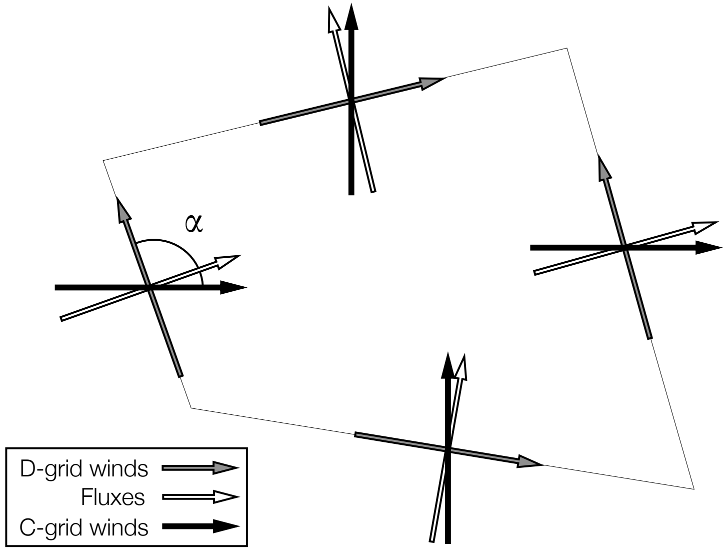
Literature

Lin & Rood 1996
Harris et al. 2019



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

The C-D grid solver

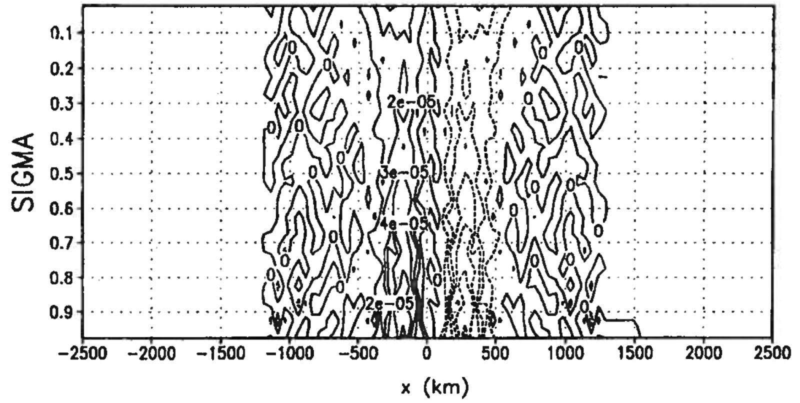


- Flux evaluation requires face-normal and time-mean fluxes.
- The C-grid winds are interpolated and then advanced a half-timestep. These are used to compute the fluxes.
- Upstream flux also allows consistent computation of the KE gradient term, avoiding the Hollingsworth-Kallberg instability
- Two-grid discretization and time-centered upwind fluxes avoid computational modes, giving FV3 high accuracy and low noise

FV3 Documentation	Literature
Section 6.2	Lin & Rood 1997

Backward horizontal pressure gradient force

- Computed from Newton's second and third laws, and Green's Theorem
- Errors lower, with much less noise, compared to traditional evaluations
    - Purely horizontal: **no** along-coordinate projection
    - PGF equal and opposite—3rd law! Momentum is conserved
    - Curl-free in the absence of density gradients



The pressure gradient (m s^{-2}) computed by the Arakawa–Suarez method. Contour interval is 1×10^{-5}.

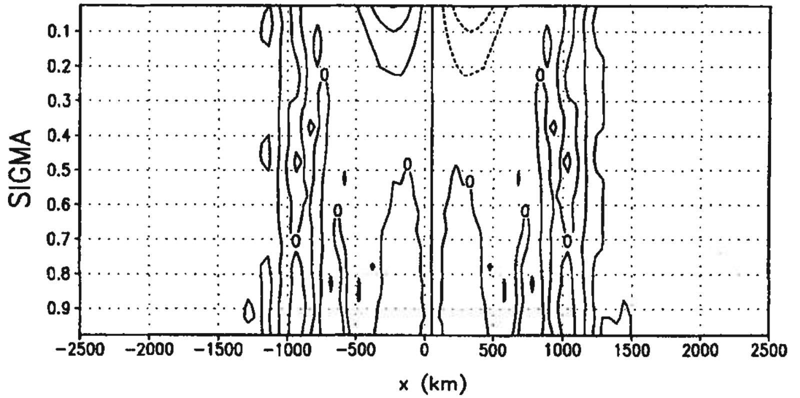
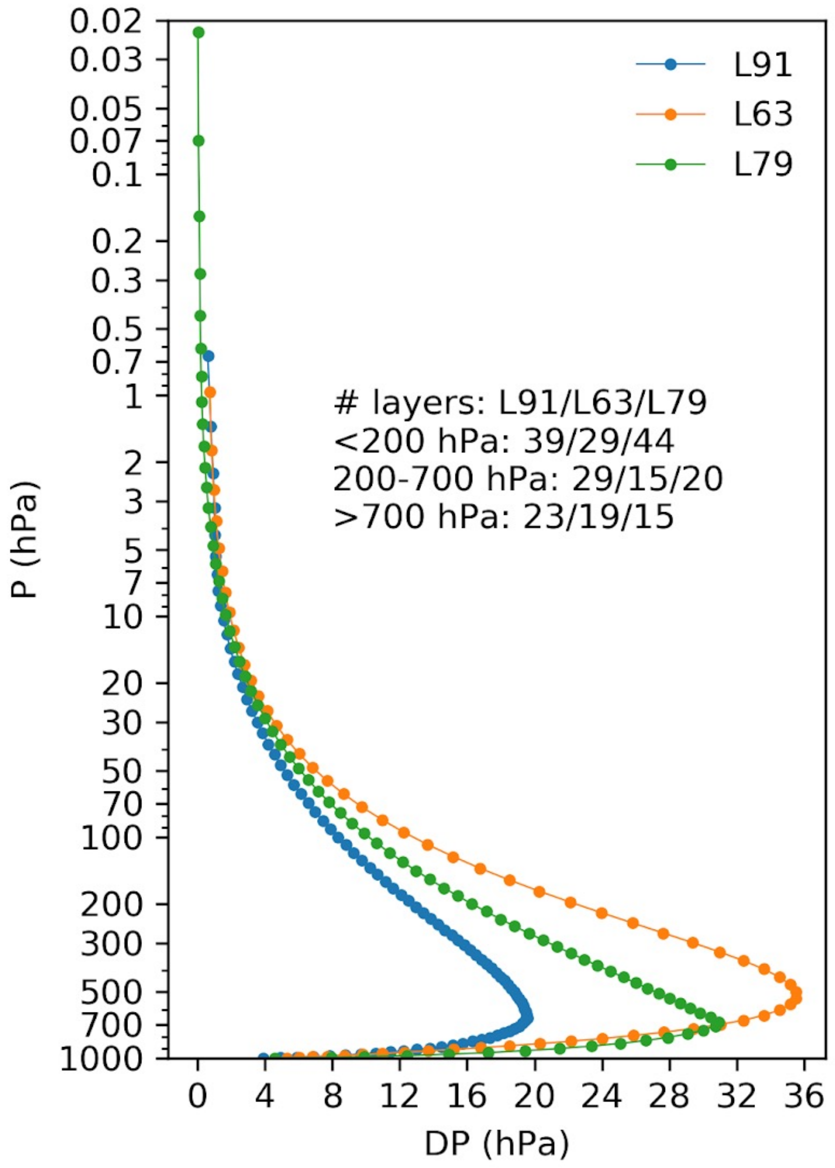


Figure 6. As in Fig. 5, but for the finite-volume method.

FV3 Documentation	Literature	Notebook
Section 6.6	Lin 1997	mtn_rest_100km, mtn_wave_tests

Vertical Levels in FV3

- Reference interface pressure: $p_k = a_k + b_k * p_s$
  - Top level is k=1
- A library of vertical level setups are in fv_eta.F90 with differing `npz`, ptop, and level positions.
  - These are carefully built to avoid instability and place levels where they are needed. *Be careful when creating new sets of vertical levels.*
- `npz`: number of vertical levels
- `npz_type`: specific choice of level, depends on `npz`



GFS-like vertical level setups for SHiELD
Courtesy Linjiong Zhou

The Lagrangian Vertical Coordinate

Literature

Lin 2004

FV3 Documentation

Sec. 5.1, 5.3; Chap 7

- Vertical motion and advection is *implicit* through the deformation of quasi-horizontal layers.
  - **No** Courant number restriction or time-splitting
  - Computing δp and δz is sufficient for vertical advection.
- Periodically, a high-order conservative remapping back to the reference "Eulerian" coordinate is done to avoid δp → 0

```
 for cosz calculations: nswr,deltim,deltsw,dtswh =          10
 180.000000000000        1800.00000000000        0.500000000000000
 anginc,nstp =  1.308996938995747E-002          11
      2016           8           4           5           0           0
ZS        6849.180       -412.0000        231.8707
PS max =    1054.338       min =    441.9276
Mean specific humidity (mg/kg) above 75 mb=   3.994439
Total surface pressure (mb) =    985.8600
mean dry surface pressure =    983.2388
Total Water Vapor (kg/m**2) =    26.56078
--- Micro Phys water substances (kg/m**2) ---
Total cloud water=  6.7240514E-02
Total rain  water=  2.4596382E-02
Total cloud ice  =  3.9533786E-02
Total snow       =  2.1293228E-02
Total graupel    =  1.5756246E-02
---------------------------------------------
TE ( Joule/m^2 * E9) =   2.631335
UA_top max =    144.7256       min =   -42.04172
UA max =    144.7256       min =   -52.63876
VA max =    81.54002       min =   -75.08163
W  max =    37.20443       min =   -10.95220
Bottom w max =    6.904078       min =   -8.178501
Bottom: w/dz max =   0.2614583       min =  -0.3034391
DZ (m) max   20.60121       min   5908.555
Bottom DZ (m) max =   -20.60121       min =   -32.27457
TA max =    317.2014       min =    175.0712
OM max =    94.72382       min =   -140.0013
ZTOP      40.88114       34.03699        39.23885
SLP max =    1097.546       min =    928.5116
ATL SLP max =    1023.925       min =    1000.363
fv_GFS Z500    5696.136       5786.775       5447.827       5867.303
Cloud_top_P (mb) max =   1.0000000E+10  min =    52.47940
Surf_wind_speed max =    28.65420       min =   6.4052401E-05
```

3.25-km X-SHiELD: remap dt = 36 s

→ vertical courant number = [-10.9,9.36]

Vertical Remapping Schemes: Namelist Options

`kord_mt`	u and v
`kord_wz`	w
`kord_tm`	Temperature (< 0, recommended) or potential temperature (available in forthcoming release); and density
`kord_tr`	Tracers

- **Strongly** recommend all options use the same remapping scheme
- Remapping temperature is *geopotential conserving*
- Remapping does not preserve correlations but 8 does so approximately

`kord`	
4	Monotone PPM
6	Vanilla PPM
7	PPM with Huynh's monotonicity constraint (more expensive but less diffusive)
8	Cubic spline (Zerroukat, 2006), Huynh (1996) monotonicity constraint everywhere
9, 10	8 but with conditional application of Huynh constraint
11	Cubic spline with 2Δz oscillations removed
16	Perfectly-linear scheme (experimental; be careful!!)

FV3 Documentation

Section 5.3

Semi-implicit nonhydrostatic solver

- Semi-implicit solver cleanly extends FV Lagrangian dynamics into nonhydrostatic regime
- Start with advected w*, z*
  - *Consistent* with other variables
- Vertical pressure gradient and non-advective changes to layer depth δz are solved by semi-implicit solver
  - Simultaneous solution for w and δz through diagnosed p'
  - p' accurately interpolated to interfaces using cubic spline
- Vertically-propagating sound waves weakly damped. That's OK.

$$\frac{\partial}{\partial t}\delta z^* = \delta w^*.$$

$$\frac{\partial}{\partial t}(w^*\delta m) = \delta p',$$

$$\frac{\partial p'}{\partial t} = \gamma p \frac{\delta w^*}{\delta z^*}.$$

$$p = \left(\frac{\delta m}{\delta z} R_d \Theta_v\right)^\gamma$$

$$p = p^* + p'$$

FV3 Documentation

Section 7.1

Lagrangian nonhydrostatic dynamics, how do they work?

- Recall that FV3 uses a hybrid-pressure coordinate. Cell mass δp is constant during sound wave processes.
- Nonhydrostaticity creates pressure perturbation
  - p computed by ideal gas law, incorporating heating
  - p* computed through δp above
- Vertical gradients in p' create vertical accelerations, deforming the Lagrangian interfaces
- Elastic straining (expansion/compression) of the Lagrangian layers alters δz
- Adiabatic changes to δz changes p'…

$$\frac{\partial}{\partial t}\delta z^* = \delta w^*.$$

$$\frac{\partial}{\partial t}(w^*\delta m) = \delta p',$$

$$\frac{\partial p'}{\partial t} = \gamma p \frac{\delta w^*}{\delta z^*}.$$

$$p = \left(\frac{\delta m}{\delta z} R_d \Theta_v\right)^\gamma$$

$$p = p^* + p'$$

FV3 Documentation

Section 7.2

Notebook

DPsupercell

Numerical Diffusion and Physical Dissipation

- *All useful atmospheric models have grid-scale motions removed by numerical diffusion (whether they know it or not).*
- Energy cascades to grid scales and **must** be removed since dissipative scales (O(1 cm)) are not explicitly resolved
- Models aren't perfect, noise and errors must be removed
  - C-grids produce particularly prodigious noise at discontinuities
- Diffusion is also a powerful tool to **improve** simulations
  - Tompkins and Semie 2017; Pressel et al. 2017; see also *Implicit LES*

FV3 Documentation
Sections 8.1, 8.2

Literature
Zhao et al. 2012
X Chen et al. 2018

Damping in FV3

FV3 Documentation

Sections 8.3, 8.4, 8.5

- FV3's physical consistency produces very few computational modes and thus can be minimally-diffusive.
  But well-configured diffusion can give *improved* results
- FV3 applies **no** direct implicit diffusion to divergent modes which cascade to grid scale *unimpeded*.
  Scale-selective divergence damping represents their physical dissipation.
- Rotational modes can be damped implicitly by monotonic advection or explicitly by vorticity damping.
  - For consistency also damps δp, δz, θ_v, w.
    No explicit damping for tracers.
- Note that **all** implicit (except vertical remapping) and explicit diffusion is **along** Lagrangian surfaces.

Numerical Damping: Namelists

nord	Damping order	
	Divg.	Vort.
1	4^{th}	4^{th}
2	6^{th}	6^{th}
3	8^{th}	6^{th}

- FV3 has a host of damping parameters to fit its many use cases.
- `nord`: Controls order of damping.
  - Higher values mean more scale-selectivity
- `d4_bg`: Nondimensional divergence damping coefficient.
  - Values between 0.1 and 0.15 recommended
  - Use smaller values for nord = 1, 2
- `do_vort_damp`: Logical flag for enabling vorticity damping.
- `vtdm4`: Nondimensional coefficient for vorticity damping. Should be much smaller than `d4_bg`.
  - Values between 0.0 and 0.06 are recommended.

Numerical Damping: Namelists

- `d_con`: Fraction of damped KE restored as **heat**, conserving energy.
  - Set to 0.0 to disable this conversion; 1.0 restores all energy.
- `delt_max`: Limit on heating from damped KE [K/s].
  - Values between 0.002 and 0.008 are recommended. Does **not** affect diabatic heating.
- `dddmp`: Coefficient for simplified nonlinear Smag damping
  - Optional, but values between 0.0 and 0.5 are potentially useful
- `sg_cutoff`: Level [Pa] above which 2Δz energy-momentum-mass conserving filter is applied.
  - Replaces older `n_sponge` option
  - Recommend applying to layers above 100 mb
- `fv_sg_adj`: Timescale [s, smaller is stronger] of 2Δz filter.
  - Use values larger than `dt_atmos` to avoid interfering with the PBL scheme.

Upper Boundary: Namelist Options

- FV3's flexible upper boundary significantly reduces gravity-wave reflection, requiring fewer sponge layers.
- `d2_bg_k1`: Strength of second-order damping in top layer (k=1).
  - Values between 0.15 and 0.2 recommended.
- `d2_bg_k2`: …in second layer (k=2).
  - Recommend values between 0.02 and 0.1.
- `rf_cutoff`: Level [Pa] above which Rayleigh damping is applied to u, v, w.
  - If `d2_bg_k1` ≤ 0 then 2nd-order damping layer is applied above this level (not just to k=1 and 2)
- `tau`: Timescale [days, smaller is stronger] of Rayleigh damping.
  - Recommend 5 for 13-km, 3 or 1.5 for 3-km, as a first try

Rigorous Thermodynamics and Physics-Dynamics Coupling

- Mass δp in FV3 includes water vapor and all condensates. Condensate loading and moist-mass effects are baked-in.
- FV3 incorporates the heat content of water vapor and condensates in adiabatic processes and diabatic heating
- Diabatic heating is applied consistently with the dynamics
  - c_p in hydrostatic: δp constant, δz dependent on T (hypsometric equation)
  - c_{vm} in nonhydrostatic: δz constant, p dependent on T (ideal gas law)

Literature

J-H Chen and Lin 2013
Zhou et al. 2019
Harris et al. 2020

FV3 Documentation

Sec. 7.2, Chapter 9
See also Linjiong Zhou's 2020 MRW lecture

A few debugging and diagnostic options

- `print_freq`: frequency (in hours if > 0; in timesteps if < 0) of diagnostic outputs (max/min/ave, global integrals, etc.)
- `range_warn`: whether to check the ranges of values at different places in the core, and print out location of bad values
- `fv_debug`: print great volumes of solver information
- `no_dycore`: turn OFF the dynamics
    - Good for debugging or testing “single column”.
- `fv_diag_column_nml`: controls column output for either forecast soundings or for debugging
- Restart files and advanced diag_manager capabilities also very useful!

A few thoughts on model development

- **Physics and dynamics form a holistic system**. Schemes, tuning parameters, and damping must be designed *together*.
  - Check physics tendencies and use the debugging options to hunt down instabilities.
- For weather modeling and storm modeling we recommend using `nord` = 2 or 3 and `vtdm4` of 0.03 or less.
  - It may take some work to find the best parameters for your application.
- More damping may not mean greater stability. Overdamped flows can lead to intense convection and explosive events, causing crashes.
- Carefully examine new vertical level setups to ensure discontinuities in δp, δz, and their derivatives do not appear, especially over high terrain.
- Weird things happen in the stratosphere. Often judicious choice of upper-level parameters (especially the $2\Delta z$ filter) can greatly suppress instabilities

Usage Guide

FV3 is a dynamical core and not a model.

- Correct: “FV3 is the dynamical core of the GFDL Modeling Suite and other UFS Configurations”
- Correct: “FV3 uses a Lagrangian vertical coordinate and the Putman and Lin (2007) advection scheme”
- Incorrect: “The microphysics and land surface in FV3 have been updated.”
- ????: “FV-3 [sic]…an inferior model [sic] which will lead to decades of isolation.”

The Global FV3 Community

Past, present, future earth and beyond

See presentation at 10:45 MDT Friday for more applications

UFS
UNIFIED FORECAST SYSTEM

GFDL
AM4 CM4 ESM4
SHiELD SPEAR

NATIONAL WEATHER SERVICE
GFSv15 v16 GEFSv12
MRW SRW HAFS …

NOAA RESEARCH
NATIONAL OCEANIC & ATMOSPHERIC ADMINISTRATION

GFDL FV3

NASA
GEOS, DAS, MERRA(2)
Ames Mars GCM

HARVARD
GEOS Chem
GEOS-Chem High-Performance

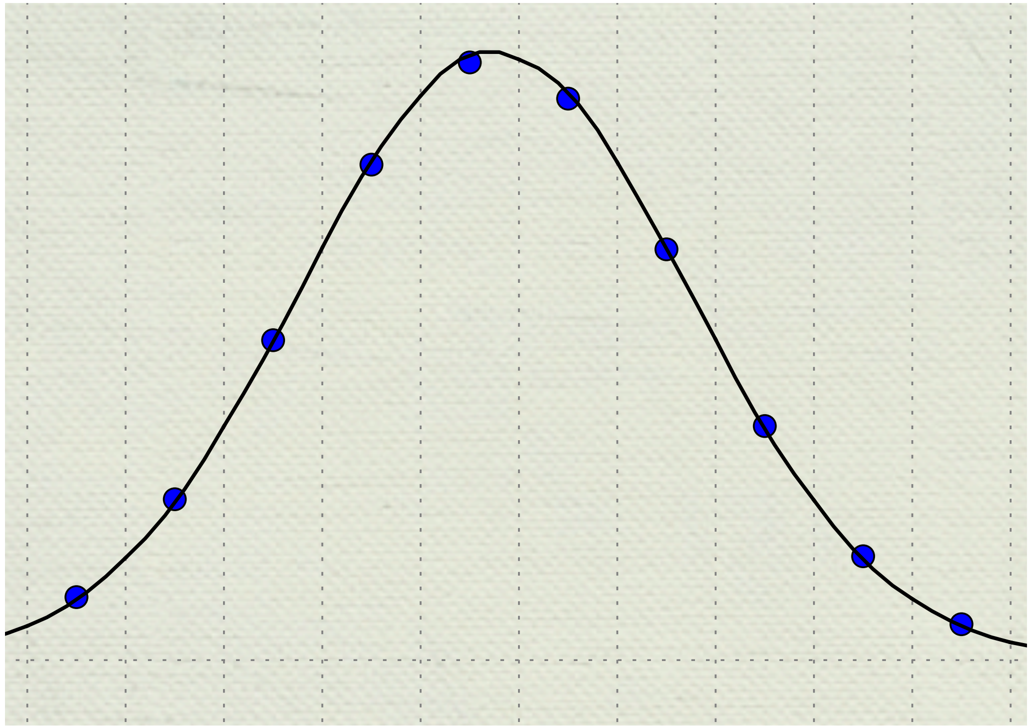
CAM-FV
CAM-FV3
NCAR

IAP
Chinese Academy of Sciences
LASG FAMIL, F-GOALS

Taiwan Central Weather Bureau
CWBGFS

The Piecewise-Parabolic Method: The cornerstone of FV numerics

- Extension to higher order of the Van Leer piecewise-linear method, itself an extension of Godunov's first-order finite-volume scheme
- The internal variation of each grid cell is approximated by a parabola, from which the fluxes through each cell interface can be integrated



Literature

Collela & Woodward 1984
Van Leer 1971–1979

FV3 Documentation

Section 4.1

The Piecewise-Parabolic Method: The cornerstone of FV numerics

- “Vanilla” PPM reconstruction is formally 4th order if Δx is constant.
- But you are free to do much more with your degrees of freedom. You can flatten or steepen or …?
- This is useful for shape-preservation (monotonicity, positive-definite) or for simply eliminating undesirable $2\Delta x$ noise

“Imagine PPM as something akin to the Toll House chocolate chip cookie recipe. The cookies you get by following the package exactly are really, really good. At the same time, you can modify the recipe to produce something even better while staying true to the basic framework. The basic cookies will get you far, but with some modification you might just win contests or simply impress your friends. PPM is just like that.”

wjrider.wordpress.com/2017/11/17/the-piecewise-parabolic-method-ppm/

Which solution is the best?

“Accuracy” analyses assume continuous sinusoid modes. They cannot incorporate discontinuities.

Centered-differencing schemes produce a lot of noise at discontinuities!! And staggered grids *preserve* the junk!

Monotonic schemes are more diffusive—but PPM gives you the freedom to balance shape-preservation with accuracy

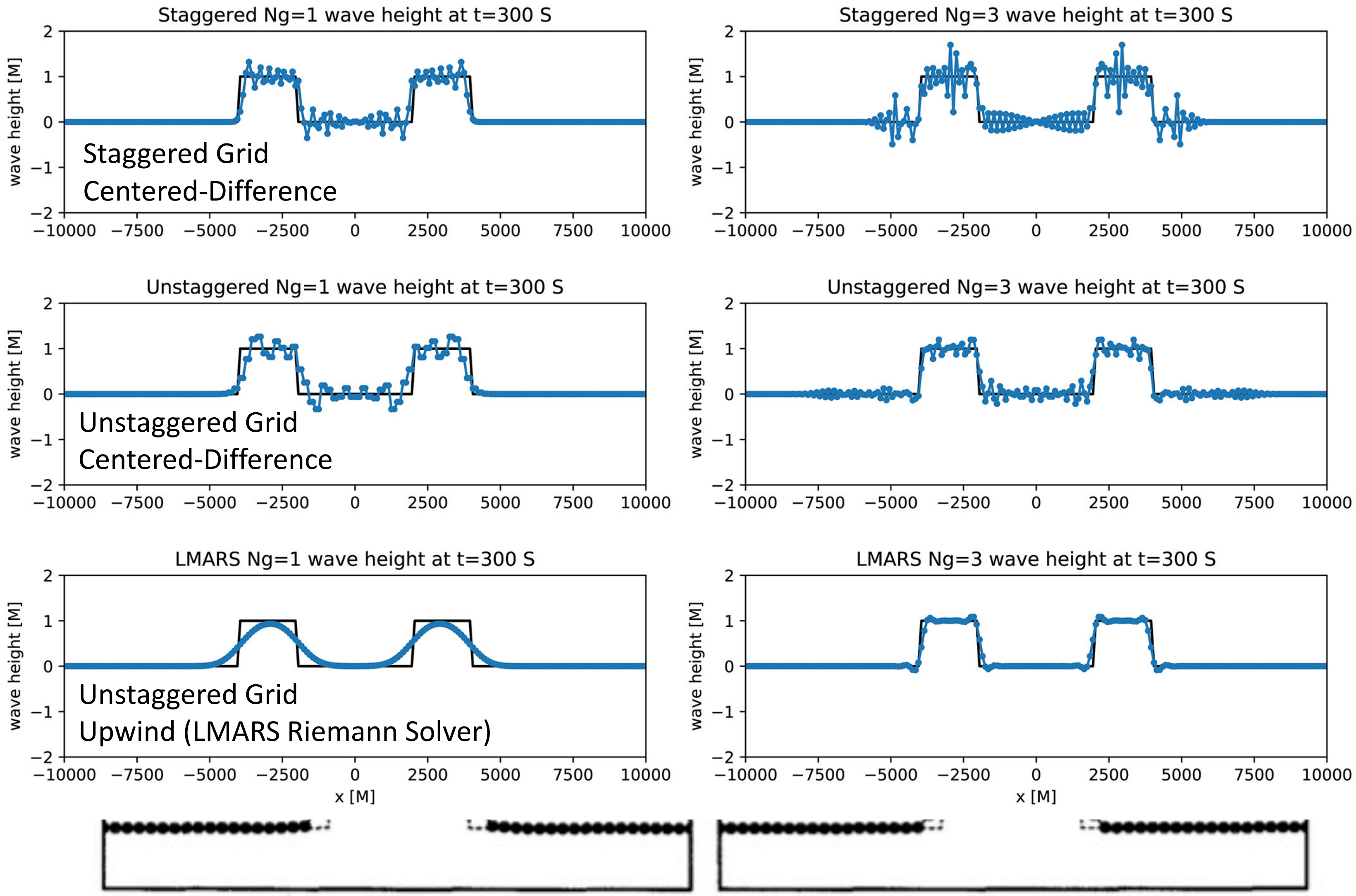
Literature

Lin and Rood 1996

X Chen et al. 2018

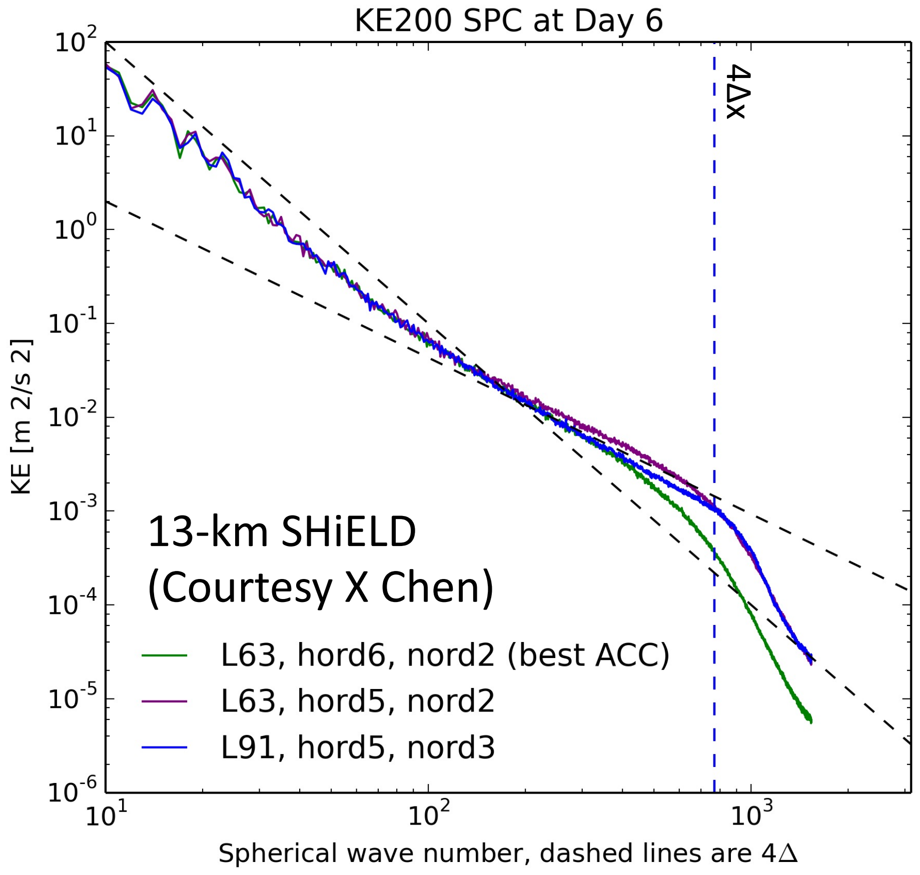
Low-order Methods

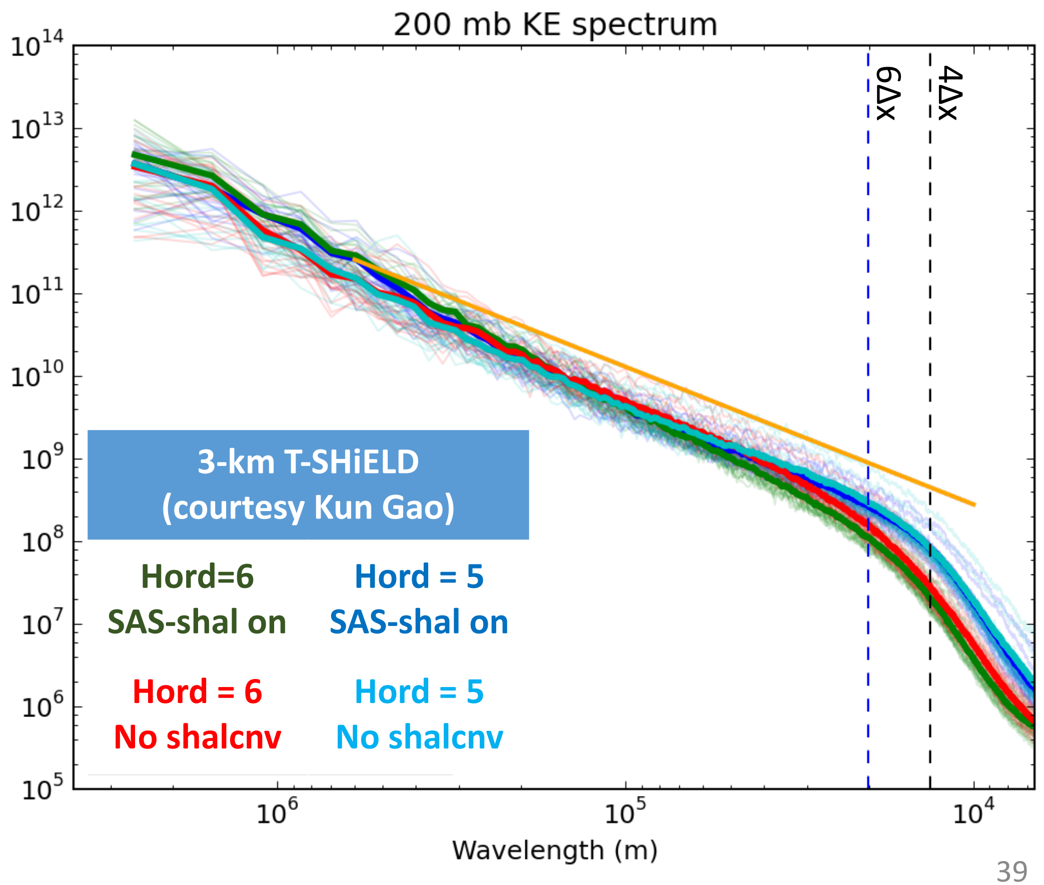
High-order Methods



Effect of advection options: 200-mb KE spectra

WARNING
Variance spectra depend on many factors, show case-to-case variability, and may not depict scientifically-credible features. Parental discretion is advised.





Tracer advection and sub-cycling

FV3 Documentation

Section 4.2

- Tracers are advected with a longer timestep than the dynamics
  - $U_{max} \approx 200$ m/s but $U_{max}+c_s \approx 540$ m/s
  - Split-explicit methods that assume $U \ll c_s$ struggle in the stratosphere
- Free-stream preservation: FV3 accumulates mass fluxes during the acoustic timesteps. These fluxes are then used to advect the tracers.
  - One or two sub-cycled timesteps is usually enough for stability.
  - Adaptively determined timestep from domain-maximum wind speed
- Tracer advection is *always* monotone or positive definite to avoid new extrema. Explicit diffusion is not used.

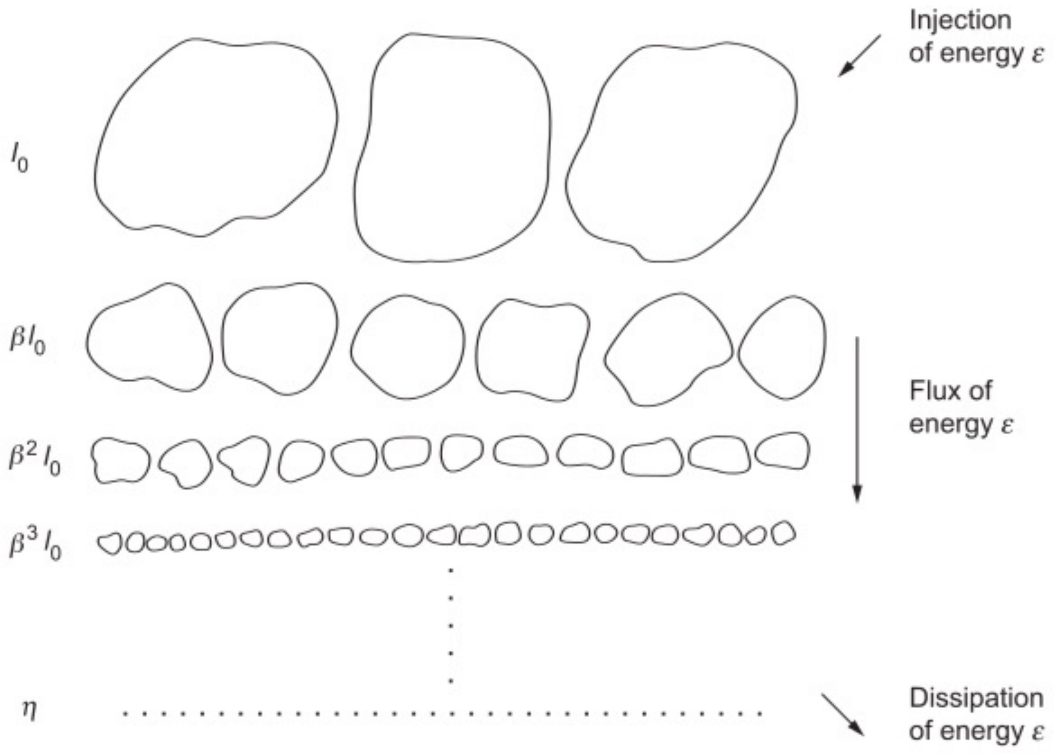
The Turbulent Energy Cascade

"Big whirls have little whirls
that feed on their velocity,
And little whirls have lesser whirls
and so on to viscosity"
—*Lewis F. Richardson, 1922*

Kinetic energy cascades from the large energy-containing scale to increasingly small-wavelength modes.

In a continuous fluid, the cascade continues until molecular diffusion can dissipate kinetic energy to heat.

In large-scale flows this is complicated by a second *upscale* turbulent cascade.

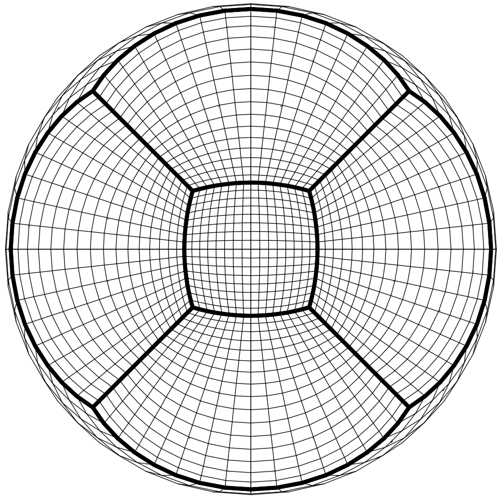


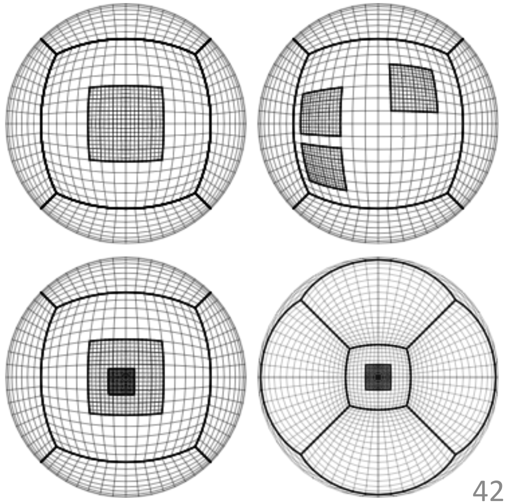
Ecke: "The Turbulence Problem" (2005)

Variable-resolution techniques

FV3 Documentation
Chapter 10

- Variable-resolution is the future of convective-scale modeling (C-SHiELD, HAFS-B)
- Stretched global grid is the easy, simple way to grid refinement.
- Two-way nesting is flexible and highly configurable.
  - Inflow BCs “baked-in” to numerics
  - Nesting methodology designed to be consistent with numerics
  - Concurrent nesting is extremely efficient:
    Run as many grids as you want at the same time!





Initialization: Namelist Options

FV3 Documentation
Chapter 9

Literature
J-H Chen et al. 2018

- `external_ic`: enable module for reading ICs from external file
- `nggps_ic`: Read regridded GFS ICs. Does no horizontal interpolation.
- `ecmwf_ic`: Read lat-lon ECMWF ICs, including horizontal interpolation
- `read_increment`: whether to read a DA increment from an external file and apply it in an FV-consistent way
- `res_latlon_dynamics`: input file for `ecmwf_ic` or increments
- `na_init`: # of forwards-backwards initialization iterations.
  - Spins up nonhydrostatic state when init from hydrostatic ICs
  - Set to 0 for GFSv15 or later ICs

When restarting

- Restarting FMS-based models is easy. Simply move the restart files from the RESTART/ directory to the INPUT/ directory.
  - Make sure to include `coupler.res`.
- Restart files are a great debugging too.
  - Run to just before the crash and inspect the NetCDF restart files.
- Be sure to set these options to ensure cross-restart reproducibility:
  - `na_init = 0`
  - `external_ic = .false.`
  - `make_nh = .false.`
  - `mountain = .true.`
  - `n_zs_filter = 0`
  - `full_zs_filter = .false.`
  - `warm_start = .true.`

Restarts: Namelist Options

- `external_eta`: read vertical level coefficients (ak, bk) from restarts instead of hard-coded values
- `agrid_vel_rst`: write out interpolated A-grid winds to restart files; very useful for DA cycling
- `npz_rst`: number of vertical levels in a restart file, if different from npz; FV3 will remap to the correct level spacing
- `make_nh`: Whether to re-generate nonhydrostatic fields from existing hydrostatic restarts. Not used for `nggps_ic`.