A Brief Introduction to Icosahedral Grids

Ross Heikes, David Randall

Dept. of Atmospheric Science
Colorado State University

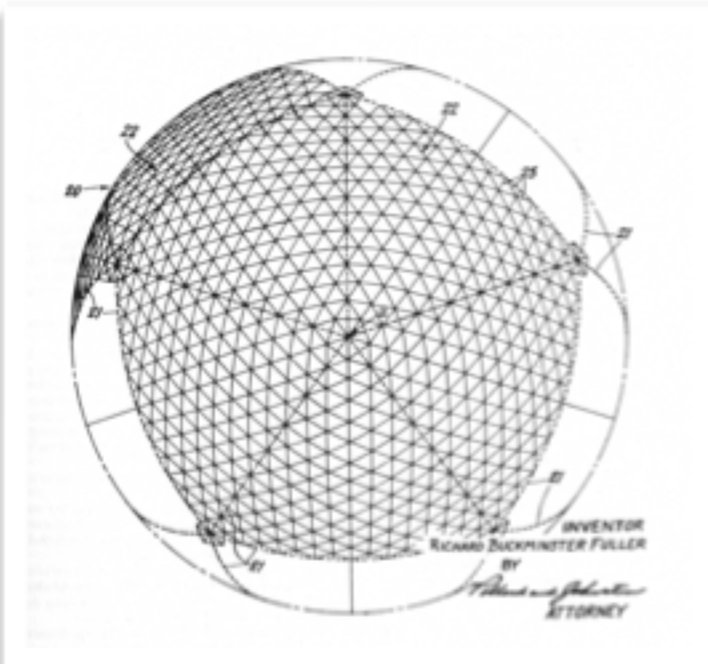
Geodynamo Benchmarking Workshop – February 5, 2015

Spherical Geodesic Grid - The Beginning - Domes 1

- The geodesic dome was invented by Walter Bauersfeld (1879-1959) of the Zeiss Optical Works in 1922.
- Used as a planetarium roof at Zeiss
- He also invented the planetarium projector

Spherical Geodesic Grid - The Beginning - Domes 2

- The geodesic dome was further championed by Buckminster Fuller (1895-1983).
- Fuller was awarded his first of several patents: US patent #2682235 (1954)
- Fuller designed the U.S. Pavilion at Expo'67' -- Montreal World's Fair. 250 feet diameter.



Spherical Geodesic Grid - Numerical Models

- **Sadourny**, R., A. Arakawa, and Y. Mintz, 1968: Integration of the nondivergent barotropic vorticity equation with an icosahedral-hexagonal grid for the sphere. *Mon. Wea. Rev.*, **96,** 351-356.
- **Williamson**, D. L., 1968: Integration of the barotropic model over a spherical geodesic grid. *Tellus*, **20,** 642-653.
- J. R. **Baumgardner**. A Three-Dimensional Finite Element Model for Mantle Convection. PhD thesis, Univ. of California, Los Angeles, 1983.

How to make an icosahedral grid. Projecting to the sphere.

- Starting with an icosahedron (fig. 1)
- We can project the icosahedron onto a unit sphere (fig. 2) forming 20 spherical triangles.

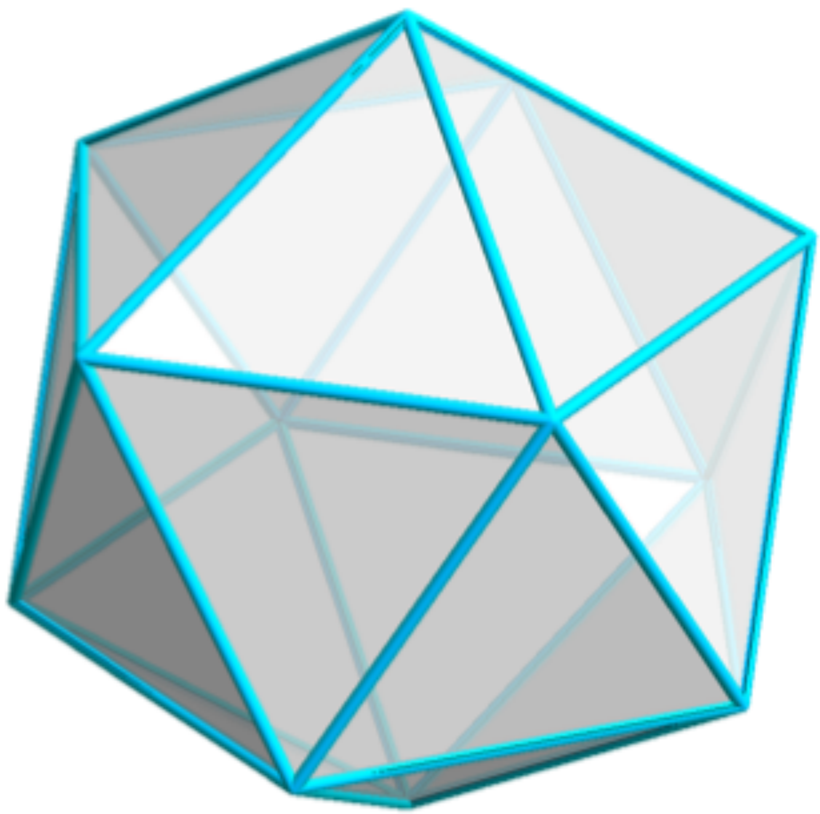


figure 1

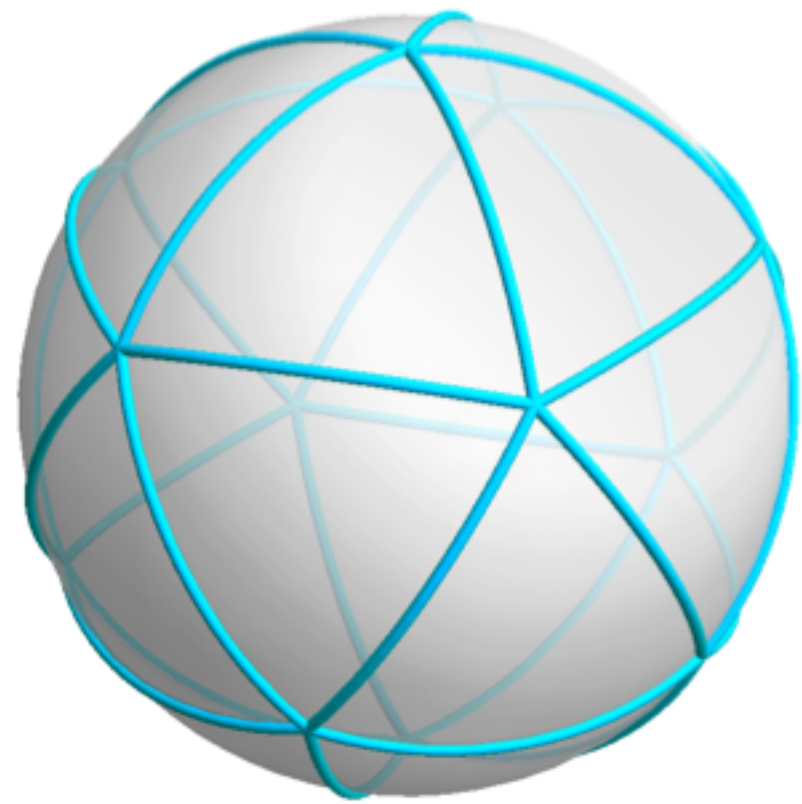
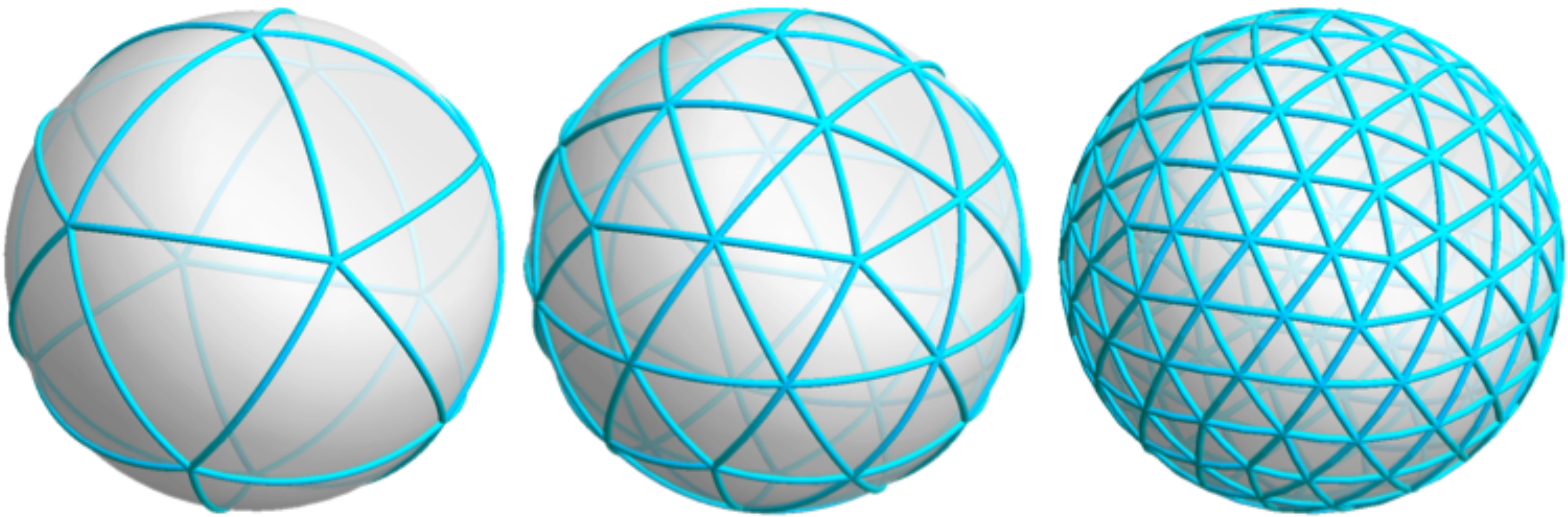


figure 2

How to make an icosahedral grid. Generating polyhedron.

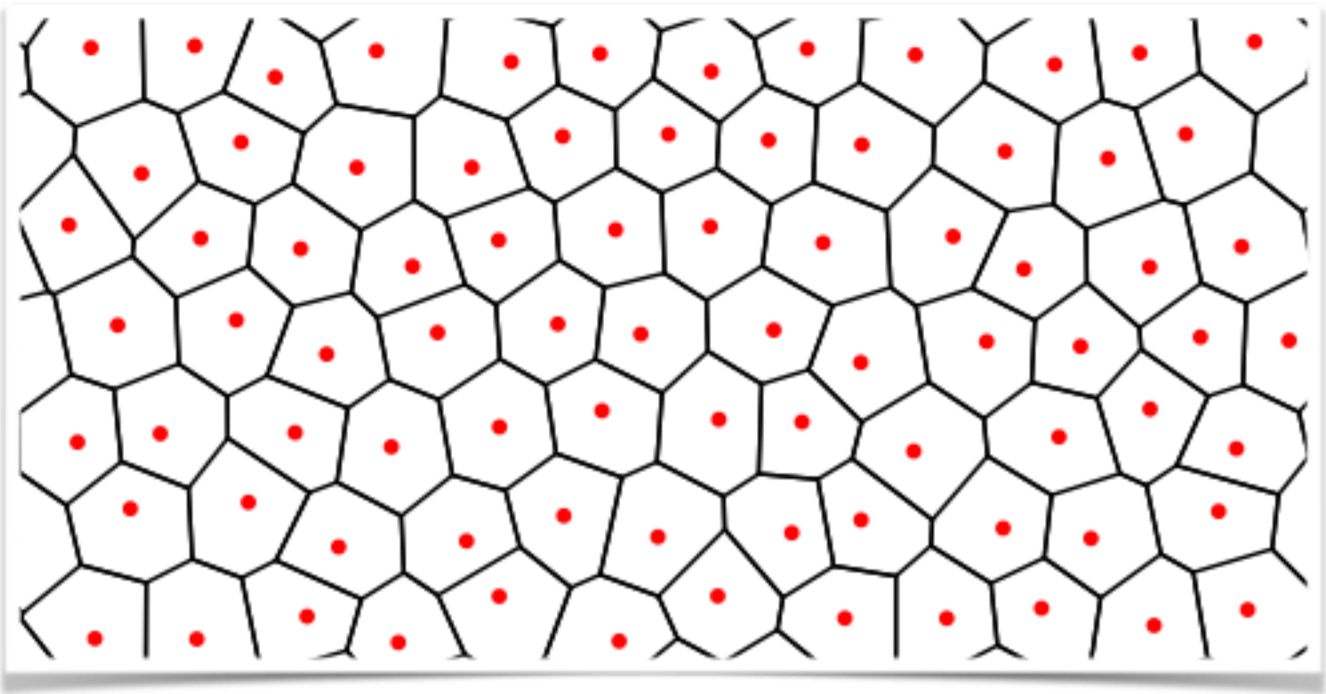
- Each spherical triangles can be further partitioned into four spherical triangles.
- The algorithm can be applied recursively.
- The vertices of these polyhedrons are used to generate the icosahedral grid.



20 triangles | 80 triangles | 360 triangles

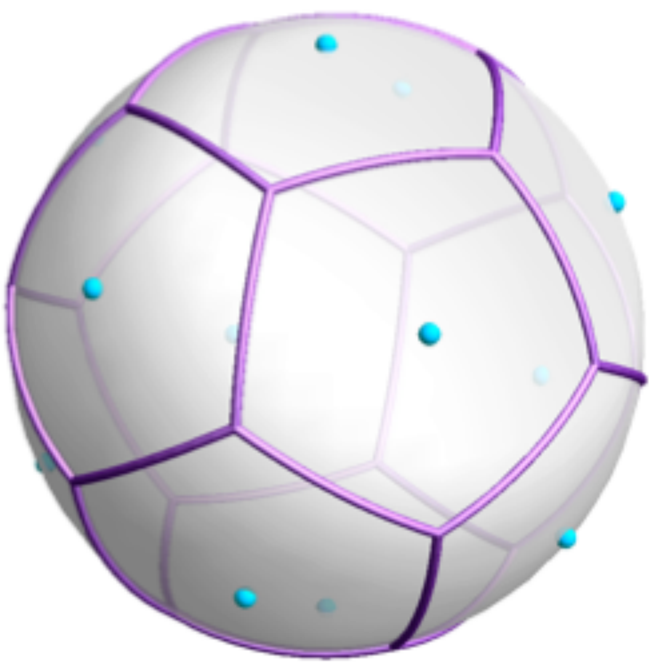
Define Voronoi Grid

- Consider an arbitrary set of grid points (red dots).
- For each grid point, there is an area which consists of all points closer to that grid point than to any other grid point.
- Each area is called a **cell**. One cell for each grid point.
- These cells are called a **Voronoi** grid.

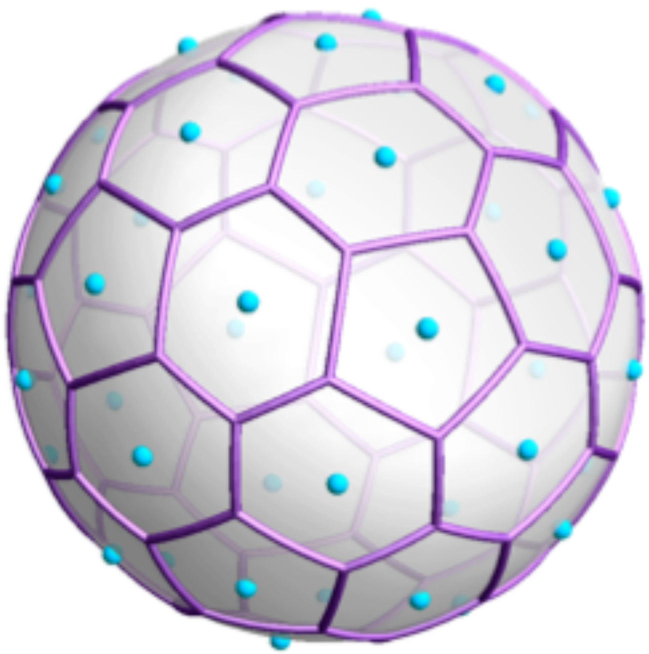


How to make an icosahedral grid. Generating polyhedron.

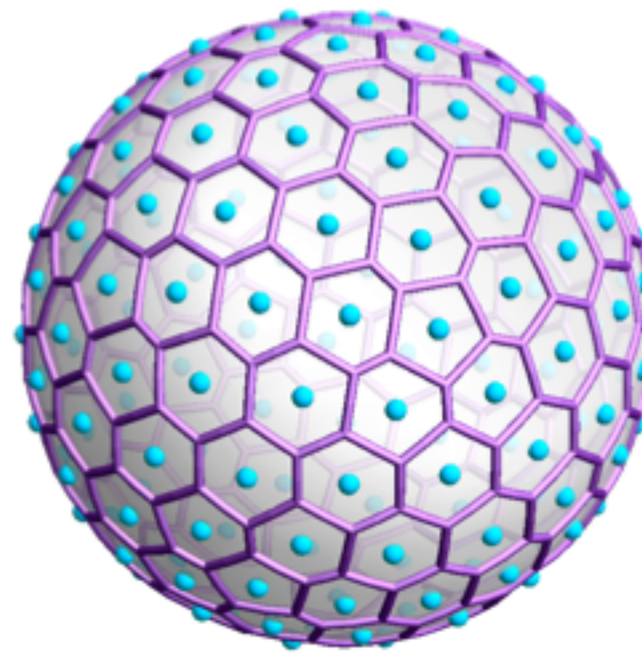
- The vertices of the previous polyhedrons (shown here as blue points) are used to generate the icosahedral grids. The vertices are called generating points.
- An area (Voronoi cell) on the sphere is associated with each generating point.
- This algorithm allows for an isotropic and homogeneous tiling of the sphere to arbitrarily high resolution.
- The positions of the grid points can be fine tuned to enhance desired numerical properties of the grid.



12 cells



42 cells

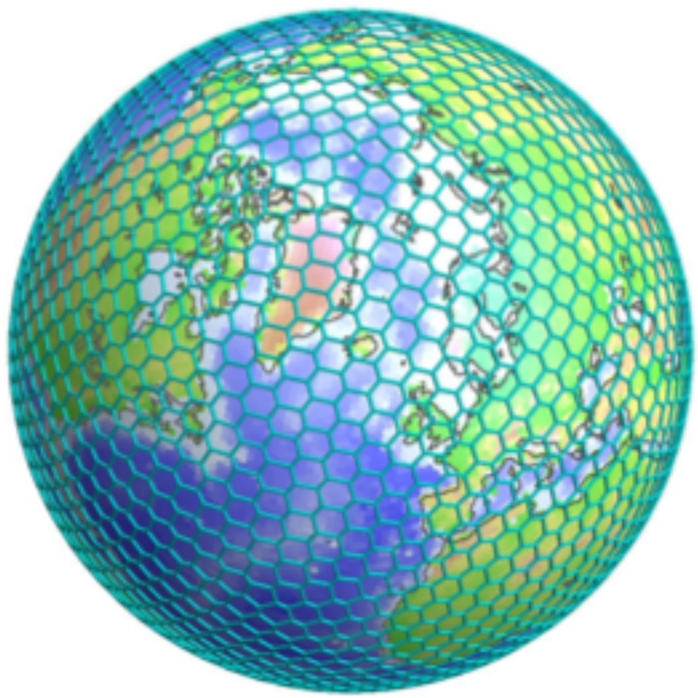


162 cells

Counting the cells

Counting the cells

- Let r denote the number of applications of the subdivision algorithm, that is partitioning one triangle into four triangles.
- Our target resolutions are:



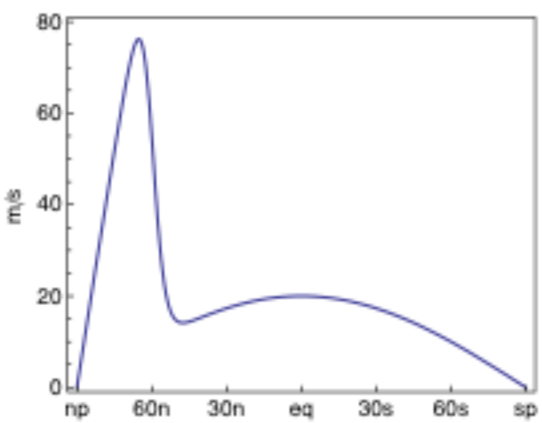
resolution 3
642 cells

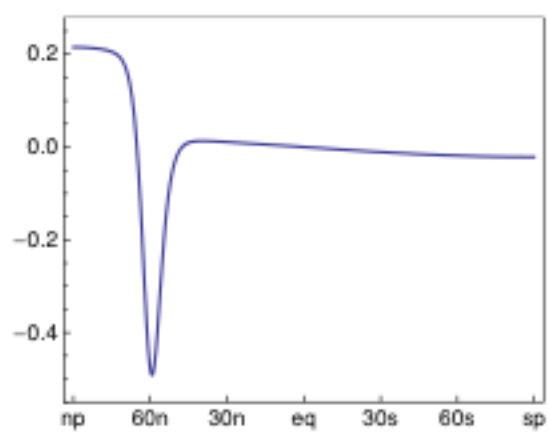
resolution (r)	number of cells	global grid point spacing (km)
9	2,621,442	14.99
10	10,485,762	7.495
11	41,943,042	3.747
12	167,772,162	1.874

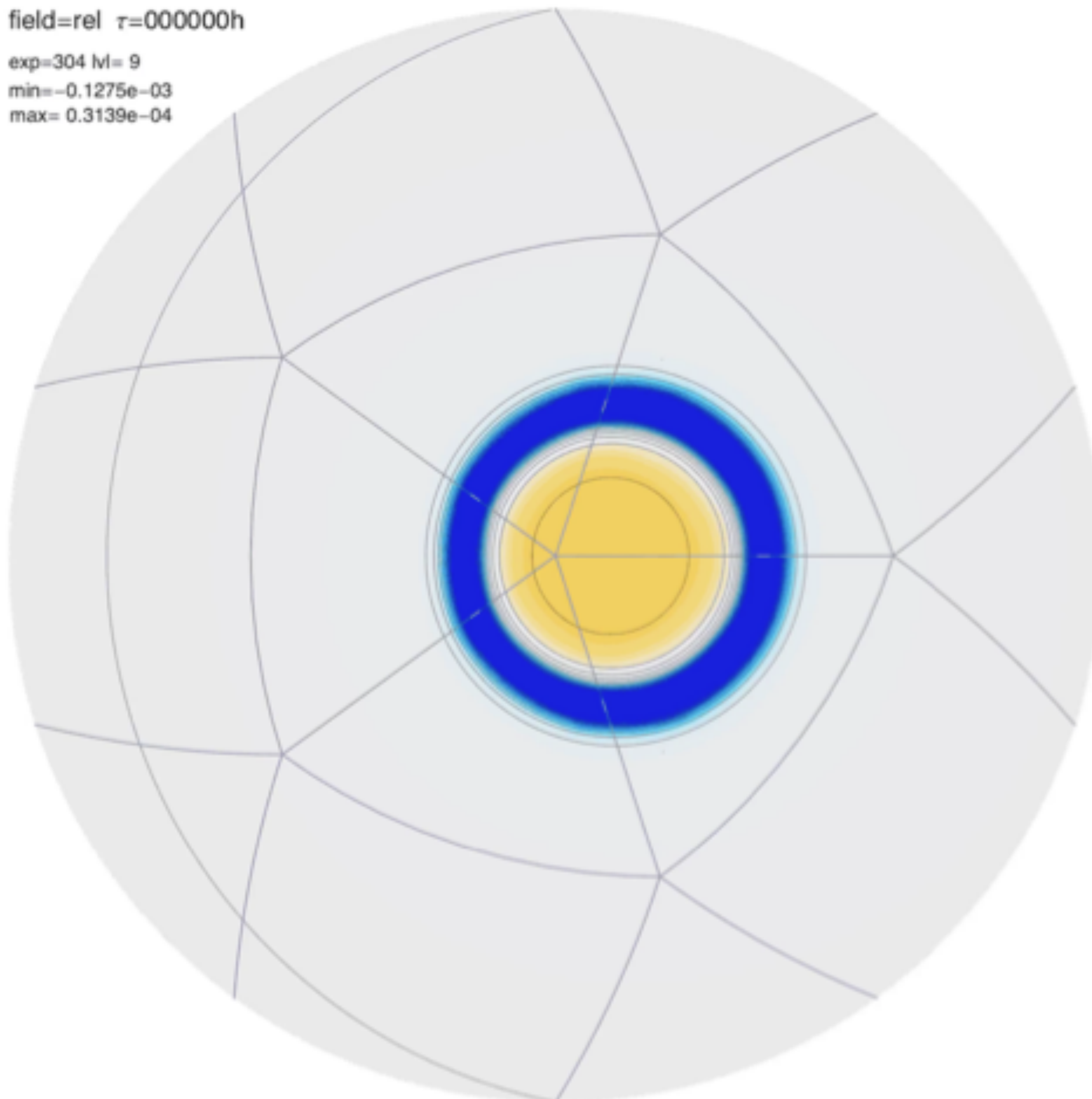
A barotropic vorticity test case

A barotropic vorticity test case

- Two superimposed solid body rotation.
- The faster is slightly offset from vertical.
- Polar stereographic projection



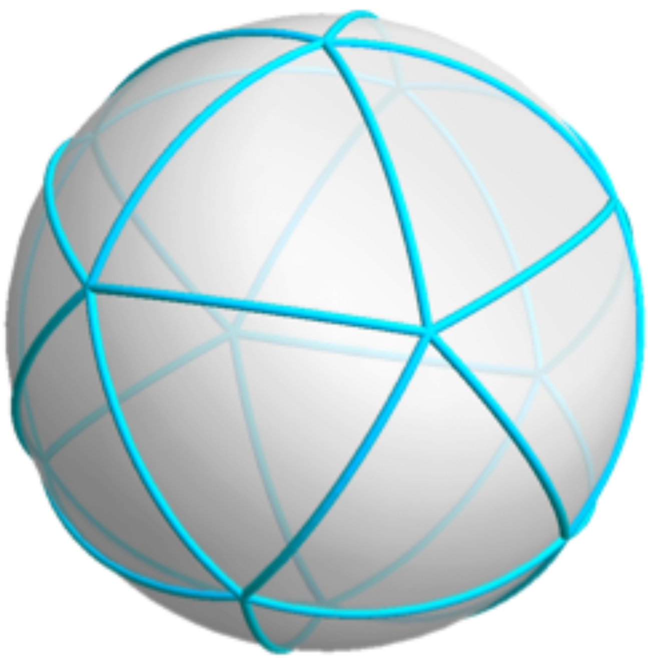




Parallel domain decomposition

Parallel domain decomposition

- An algorithm similar to the grid generation algorithm is used to partition the sphere into quadrilateral regions.
- This domain decomposition is used to assign pieces of the grid to MPI tasks.


20 triangles

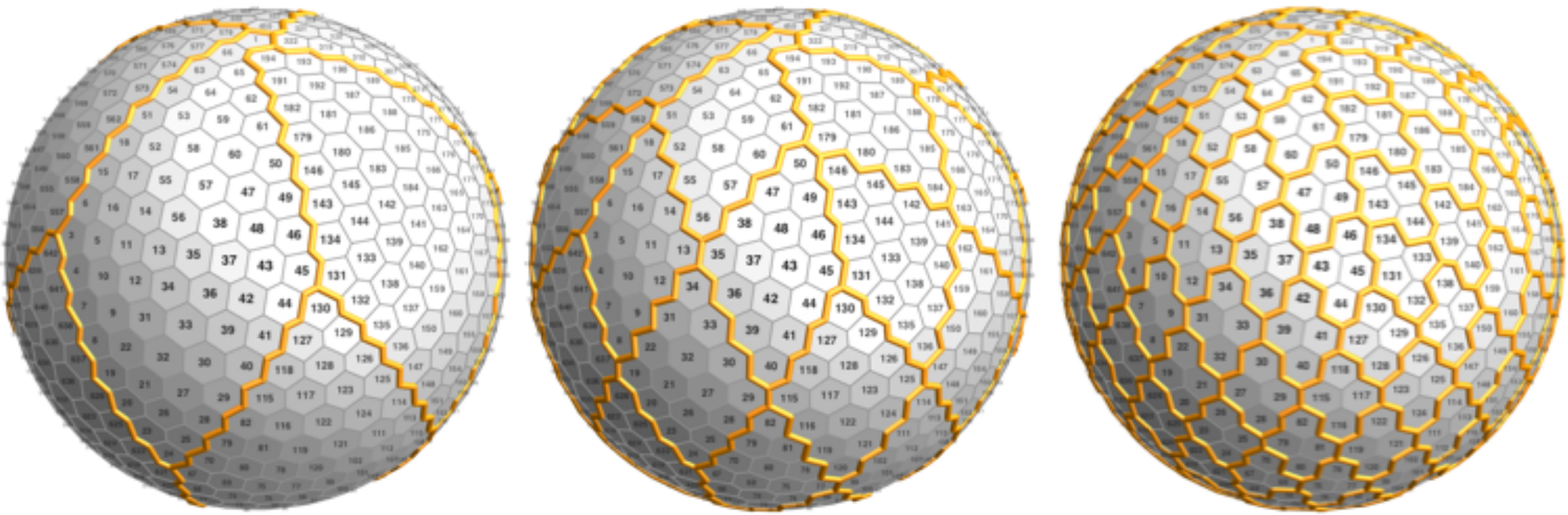

10 pieces


40 pieces

Icosahedral grid. Parallel domain decomposition. Distribution to MPI tasks.

Icosahedral grid. Parallel domain decomposition. Distribution to MPI tasks.

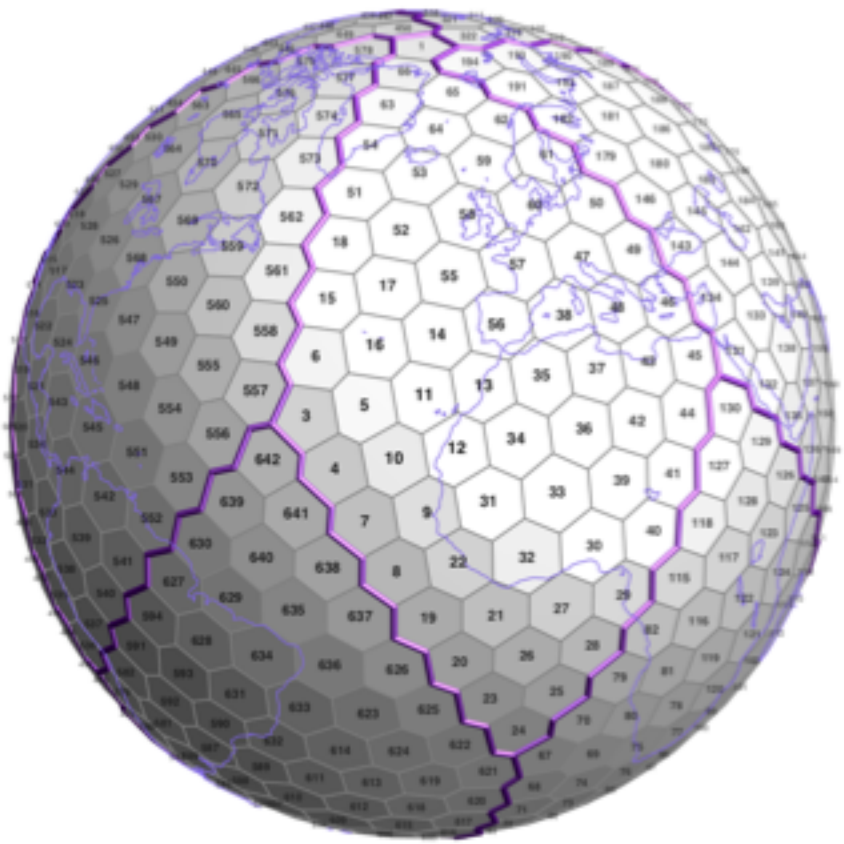
- Pieces of the grid are assigned to MPI tasks.
- MPI non-blocking sends/receives are used to update ghost regions (halo regions) with data from neighboring processes.

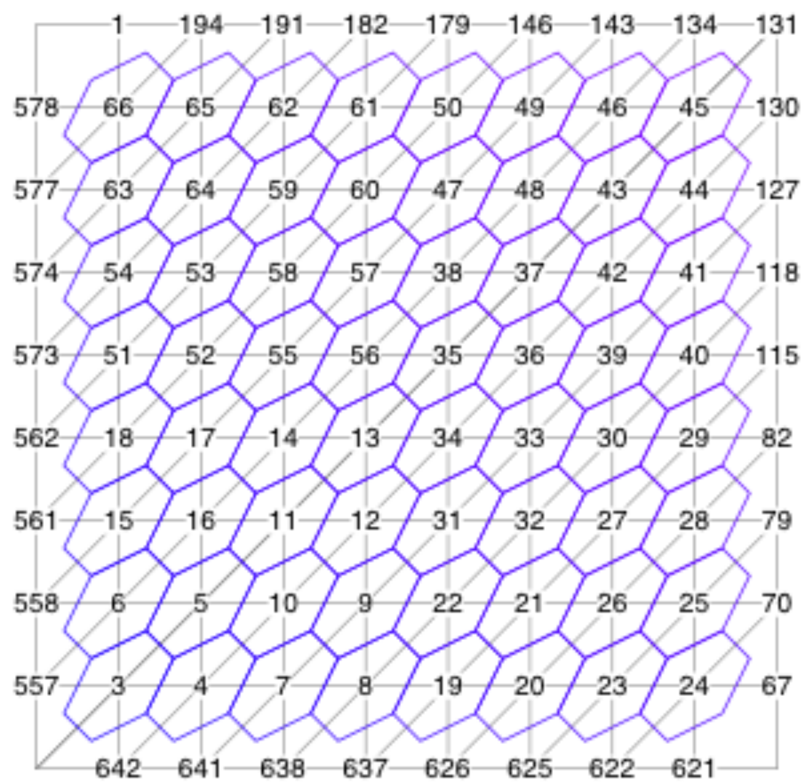


Simple 2D array storage

Simple 2D array storage

- The grid maps conveniently onto 2D data structures
- MPI non-blocking sends/receives are used to update ghost regions (halo regions) with data from neighboring processes.





Masuda and Ohnishi shallow water equations

Masuda and Ohnishi shallow water equations

- Masuda, Y., and H. Ohnishi, 1986: An integration scheme of the primitive equation model with an icosahedral-hexagonal grid system and its application to the shallow-water equations. Short- and Medium-Range Numerical Weather Prediction. Japan Meteorological Society, Tokyo, 317-326.
- Prognostic

$$\frac{\partial \eta}{\partial t} - J(\eta, \psi) + \nabla \cdot (\eta \nabla \chi) = 0$$

$$\frac{\partial D}{\partial t} - J(\eta, \chi) - \nabla \cdot (\eta \nabla \psi) + \nabla^2 (K + \phi) = 0$$

$$\frac{\partial \phi}{\partial t} - J(\phi, \psi) + \nabla \cdot (\phi \nabla \chi) = 0$$

- Diagnostic

$$\nabla^2 \psi = \eta - f \qquad \nabla^2 \chi = D$$

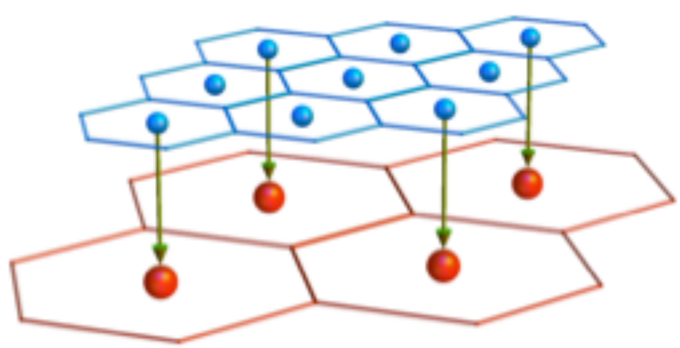
- Three operators: Jacobian, divergence and Laplacian
- We will mostly discuss the Jacobian but the approach could easily apply to the others.

- Our prognostic equations require solving a 2D Poisson equation every time step in each model layer.
- The recursive structure of the grid facilitates the use of multigrid methods.
- There are two main parts to the multigrid algorithm:

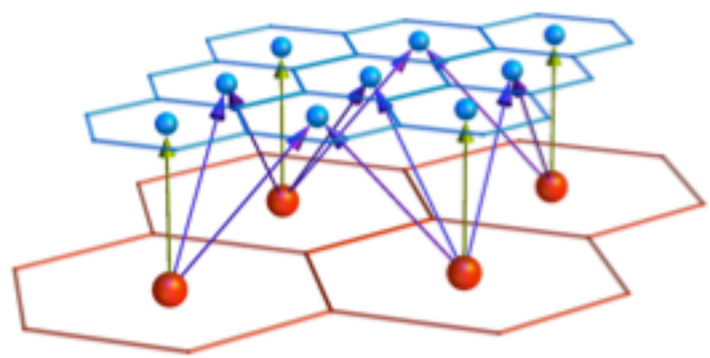
(1) Relaxation sweep. Similar to a standard Jacobi iteration. Most expensive.

$$\alpha_i = \sum_j \omega_{i,j} \alpha_{i,j} - \omega_i \beta_i \quad \text{for all} \quad i = 1, 2, \ldots, N$$

(2) Transferring information between grid resolutions. Less expensive.



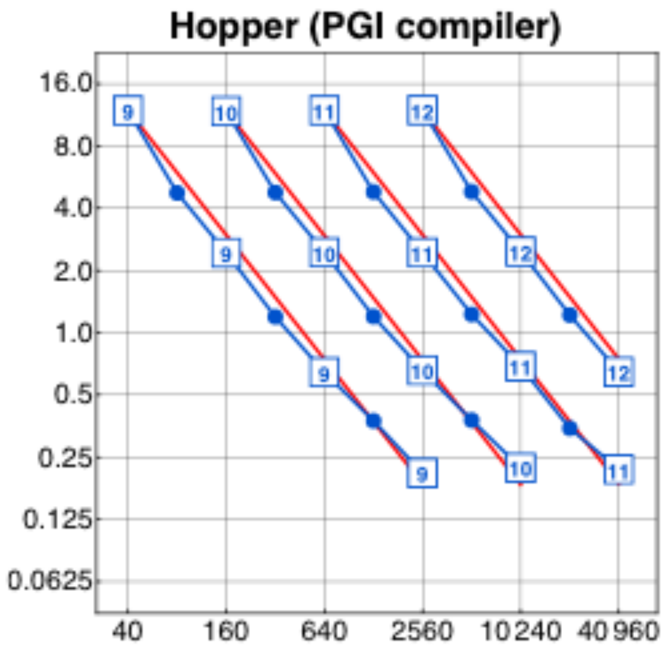
restriction



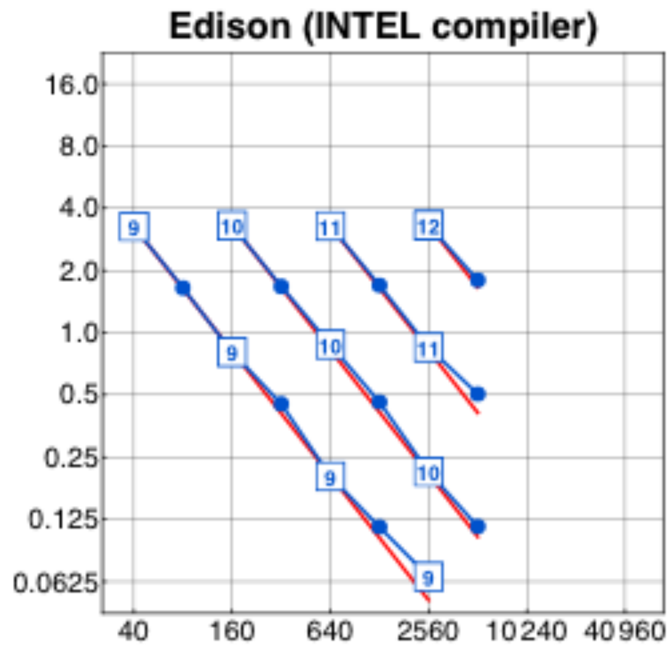
prolongation

Parallel scaling with MPI.

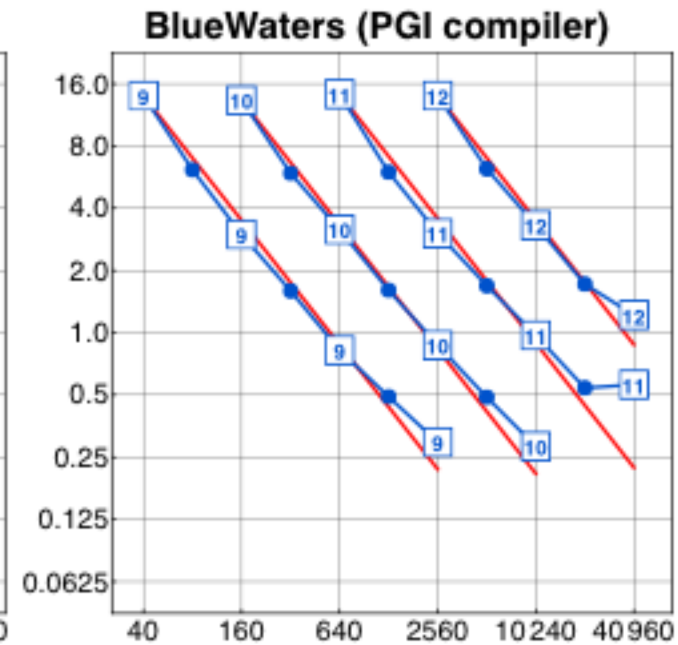
- Plot show the time to do 10 multigrid v-cycles
- X-axis is number of MPI tasks. Y-axis is time. Both are log scale.
- Each blue line indicates a particular grid resolution. Grids 09, 10, 11 and 12.
- The red line is the idealized speed-up.
- For each resolution the red line and the blue line should be coincident.
- All the same code with no heroic optimization



NERSC Cray XE6, with a peak performance of 1.28 Petaflops/sec, 153,216 compute cores.



NERSC Cray XC30, with a peak performance of 2.57 petaflops/sec, 133,824 compute cores.



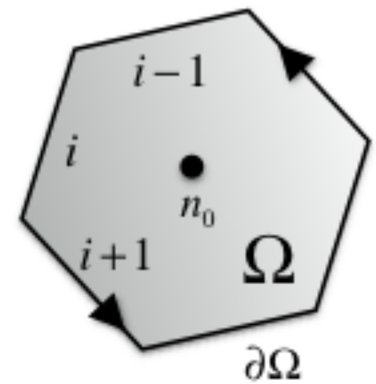
NCSA Cray XE6/XK7 system consisting of more than 22,500 XE6 compute nodes. And 4200 XK7 compute nodes (AMD Interlagos, NVIDIA GK110)

A finite-difference curl operator

- Stokes (Kelvin-Stokes) theorem: The integral of the curl of a vector field over some area equals the line integral of the vector field around the boundary of the area.

$$\int_{\Omega}\left[\mathbf{k}\cdot(\nabla\times\mathbf{F})\right]d\Omega=\oint_{\partial\Omega}(\mathbf{F}\cdot\boldsymbol{\tau})dS$$

- Considering a cell and with the following approximations:

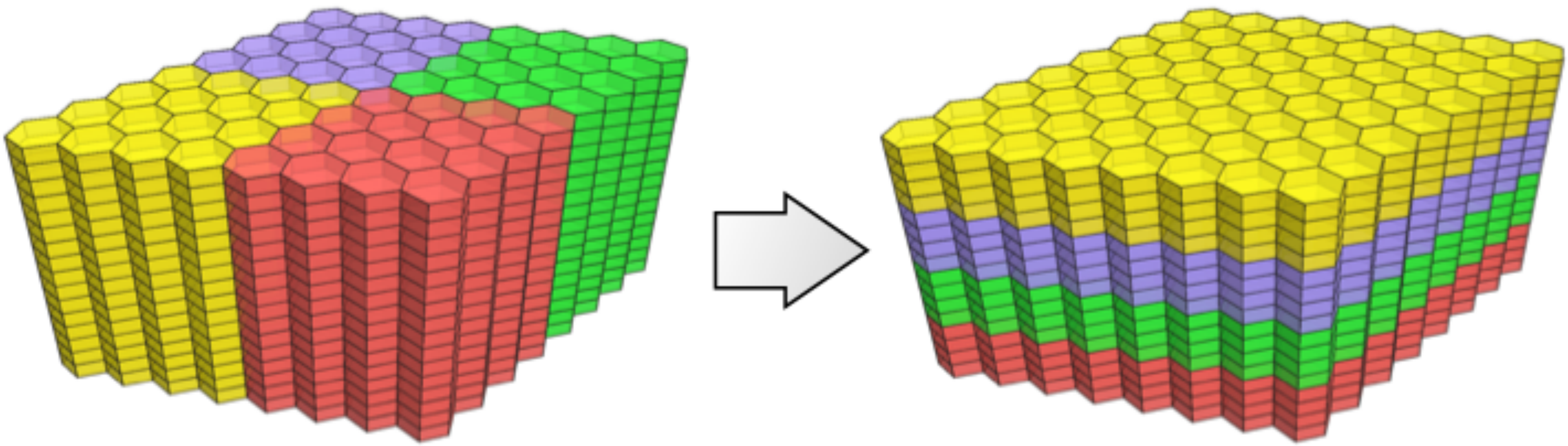


1. $\int_{\Omega}\left[\mathbf{k}\cdot(\nabla\times\mathbf{F})\right]d\Omega\approx\left[\mathbf{k}\cdot(\nabla\times\mathbf{F})\right]_0 A_0$
2. $\oint_{\partial\Omega}(\mathbf{F}\cdot\boldsymbol{\tau})dS\approx\sum_i (F_\tau)_i l_i$

- The finite-difference **curl** operator is given by

$$\left[\mathbf{k}\cdot(\nabla\times\mathbf{F})\right]_0 A_0\approx\sum_i (F_\tau)_i l_i$$

- Transpose the model blocks and columns so that blocks become bigger.
- Hopefully the speed-up will outweigh the additional communication.



Each task has a 4×4×16 block

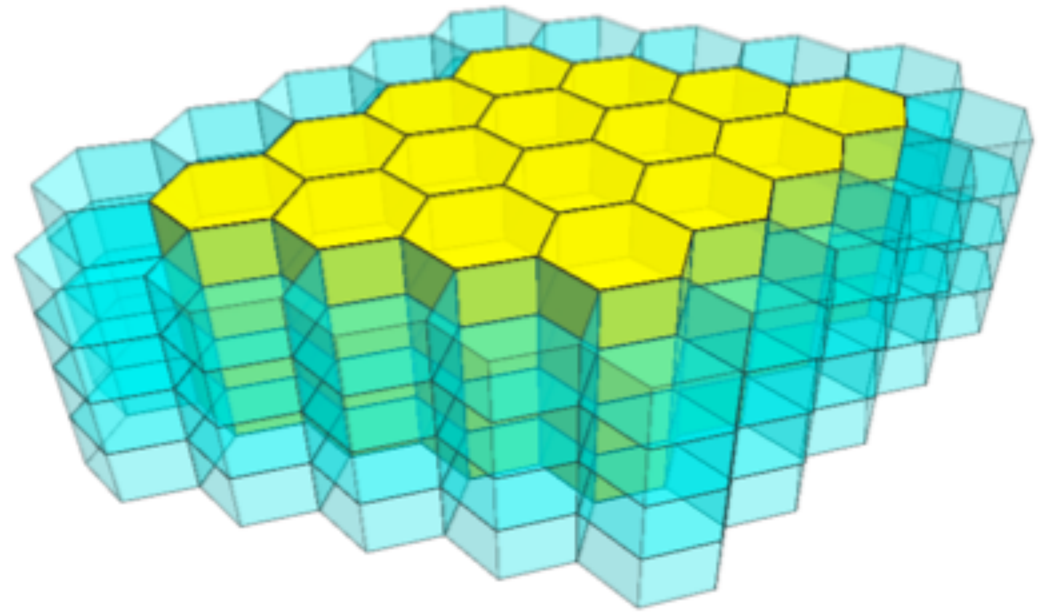
Each task has a 16×16×4 block

Parallel efficiency

- Each grid block requires information from neighboring subdomains to fill ghost cells.
- We can define **parallel efficiency** to be:

$$\text{parallel efficiency} \approx \frac{\text{number of local cells}}{\text{number of ghost cells}}$$

- **Larger parallel efficiency is better**. More useful work is done per ghost cells.

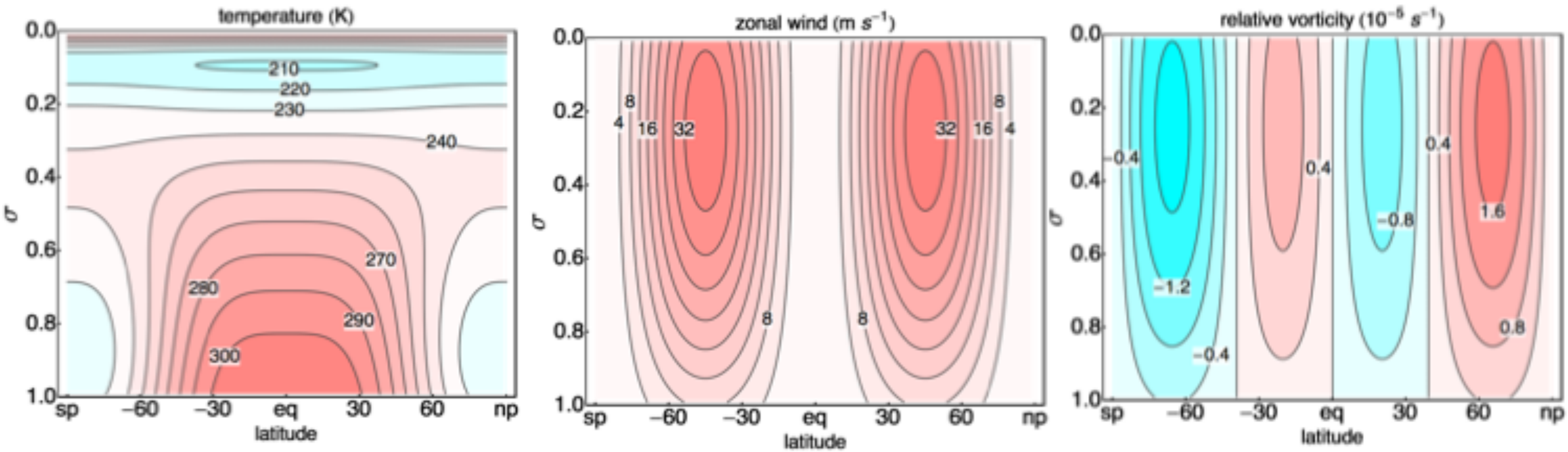


Yellow cells *belong to the local process*

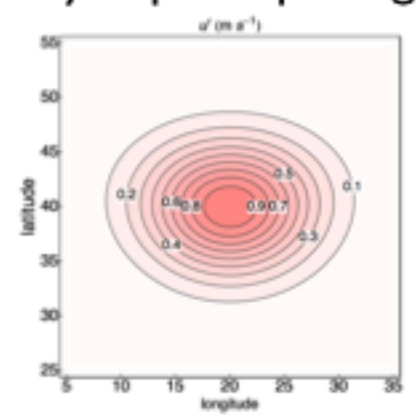
Blue cells *are ghost cells filled from neighboring process*

Extratropical cyclone in the 3D models

- Jablonowski and Williamson (2006) *Quart. J. Roy. Meteor. Soc.*, **132**, 2943-2975
- Prescribed **zonally symmetric** analytic formulas for initial prognostic variables:



- A baroclinic wave is triggered in the balanced initial conditions by superimposing a perturbation (1 m s^{-1}) in zonal wind at each model level.



Height-coordinate. Jablonowski-Williamson.

- Surface theta is plotted
- Grid 7 (60 km)
- Day 10