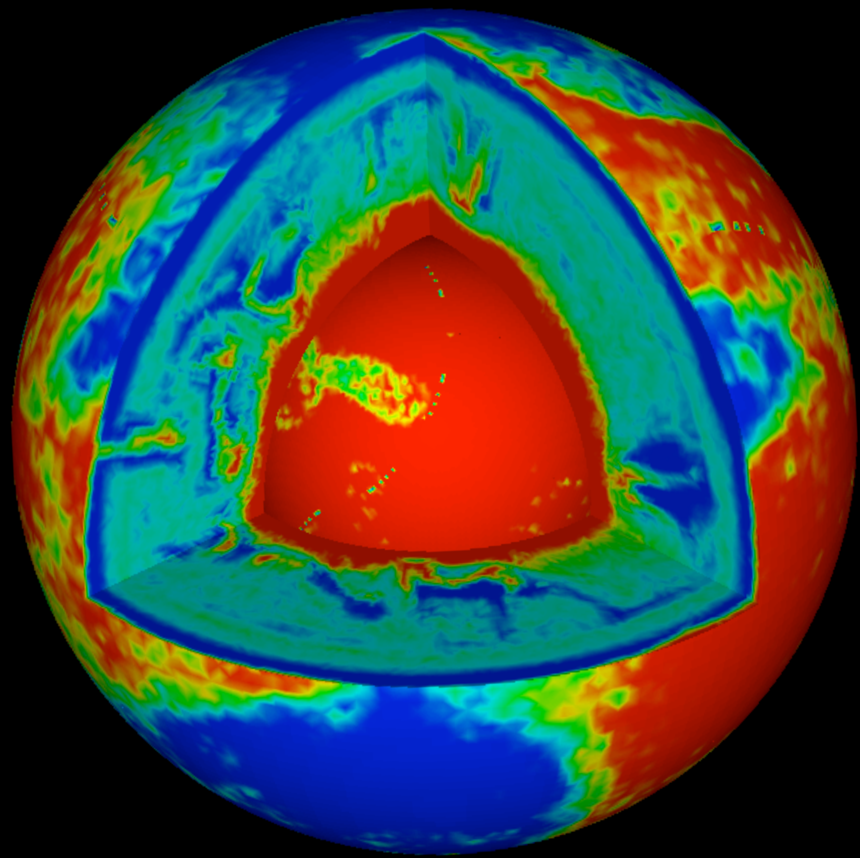
3D Spherical Mantle Convection Calculations using the Yin-Yang grid

Paul Tackley

ETHZ Institut für Geophysik

(contributions from T. Nakagawa (Kyushu U.)

J. Hernlund (UBC), F. Deschamps, J. Connolly)



Outline

- Background
- Technical details
- Examples

See 2 recent PEPI papers for more details:

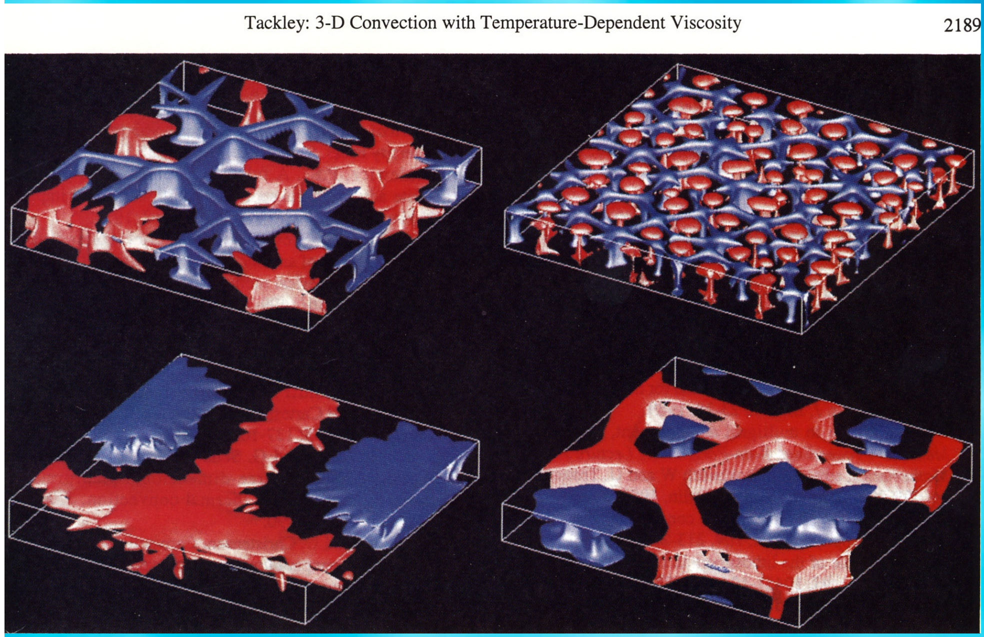
Tackley, P. J., Modelling compressible mantle convection with large viscosity contrasts in a three-dimensional spherical shell using the yin-yang grid

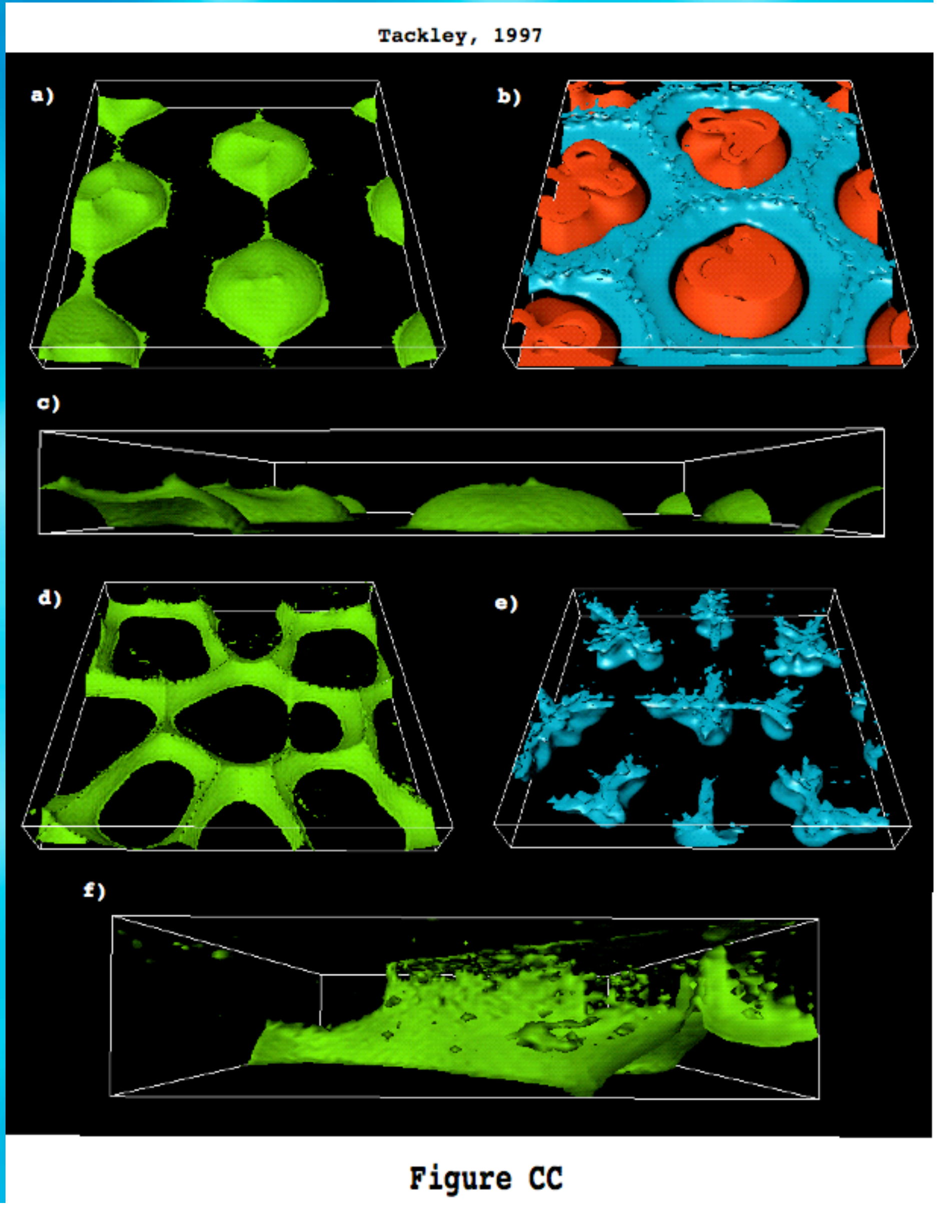
Hernlund, J. W. and P. J. Tackley, Modeling mantle convection in the spherical annulus.

Stag3D: 1992-

1998 AGU monograph

1993 GRL

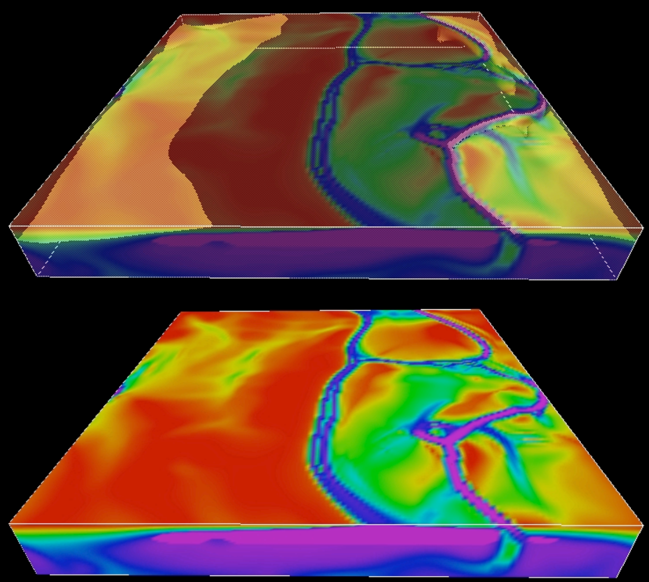


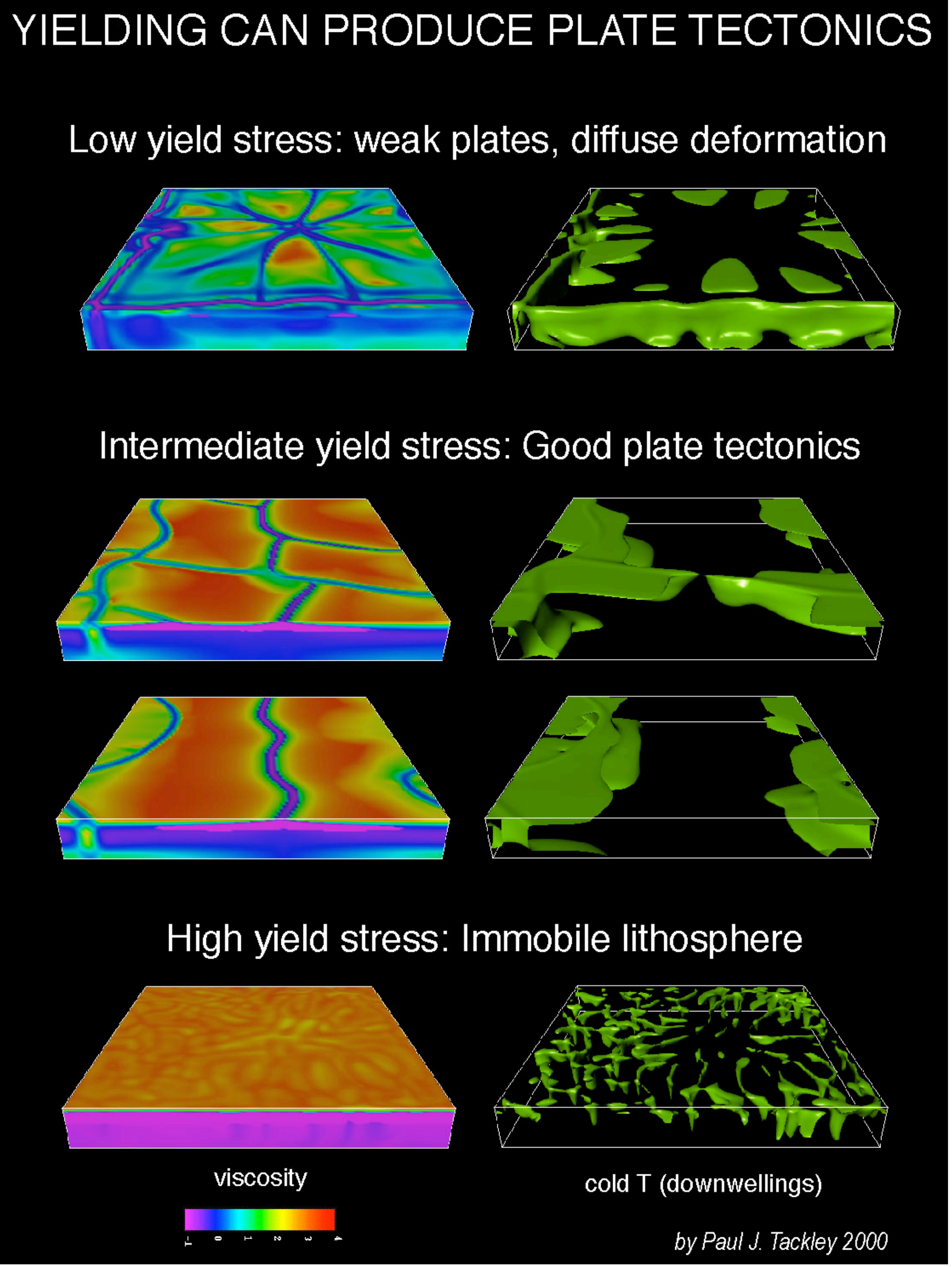


Compressible TALA

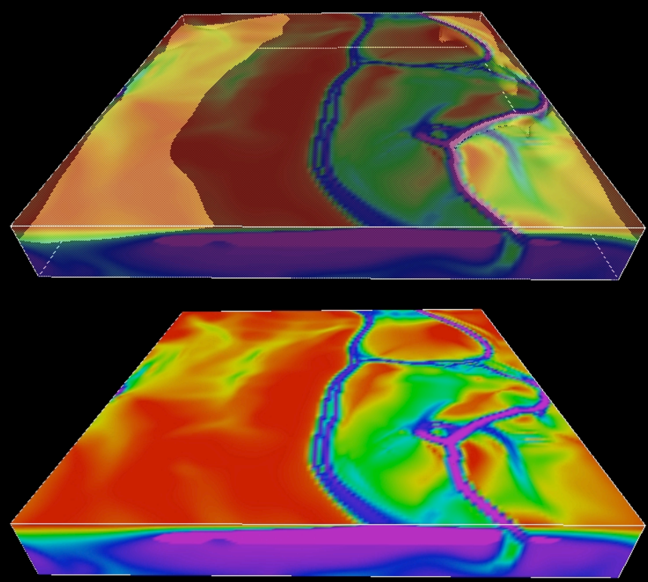
3D cartesian
2D cartesian, axisymmetric
or cylindrical

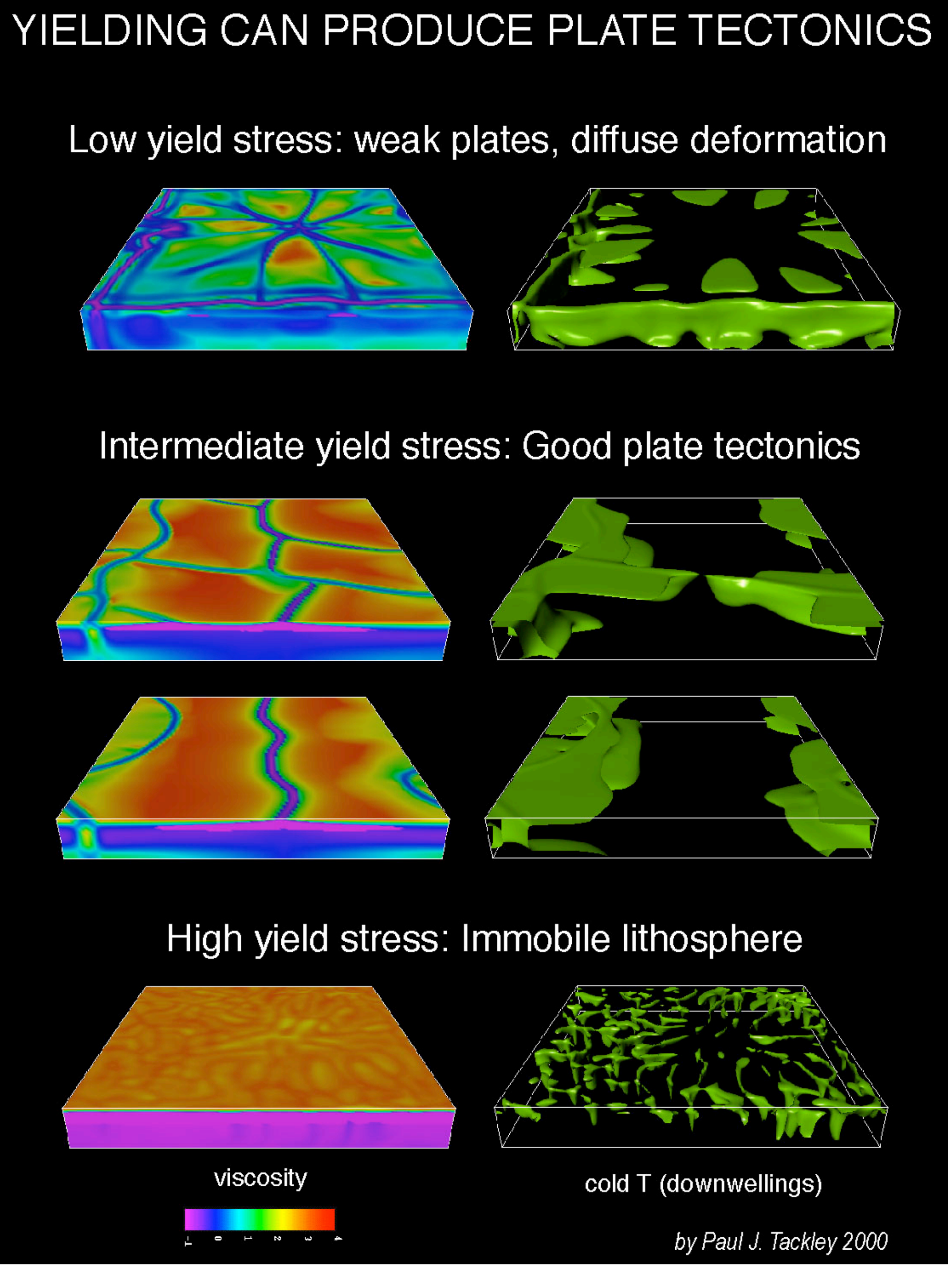
Self-consistent plate tectonics (2000ab)



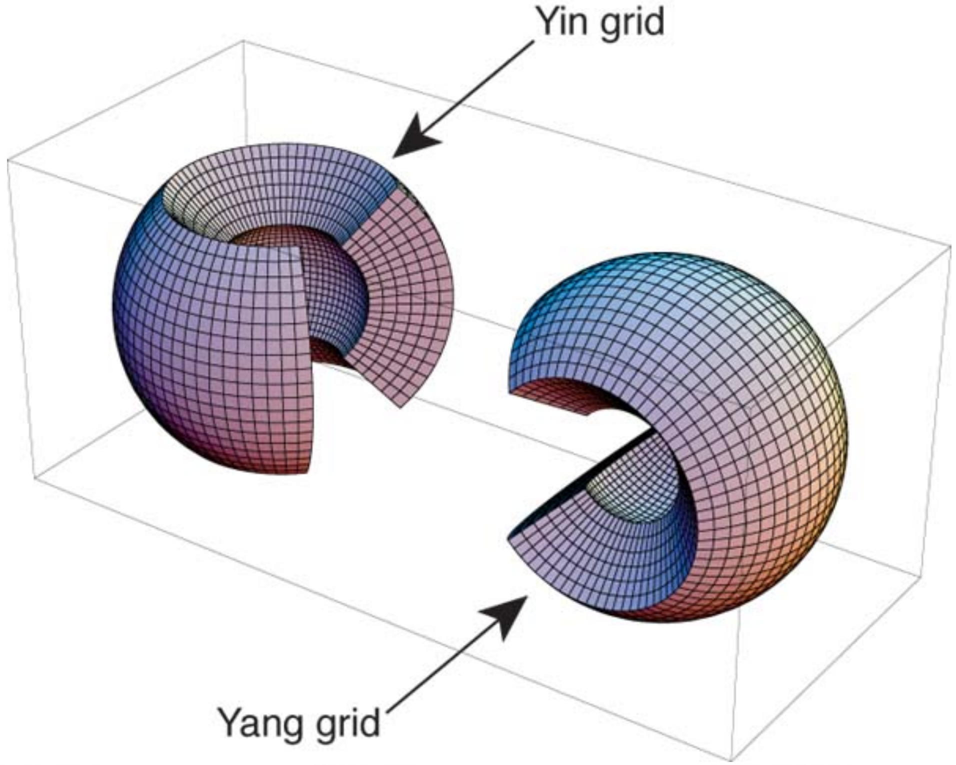


Cartesian: how to make spherical?

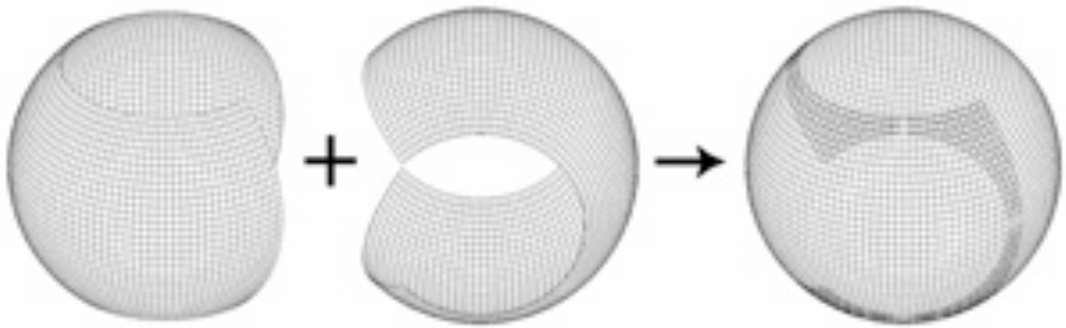




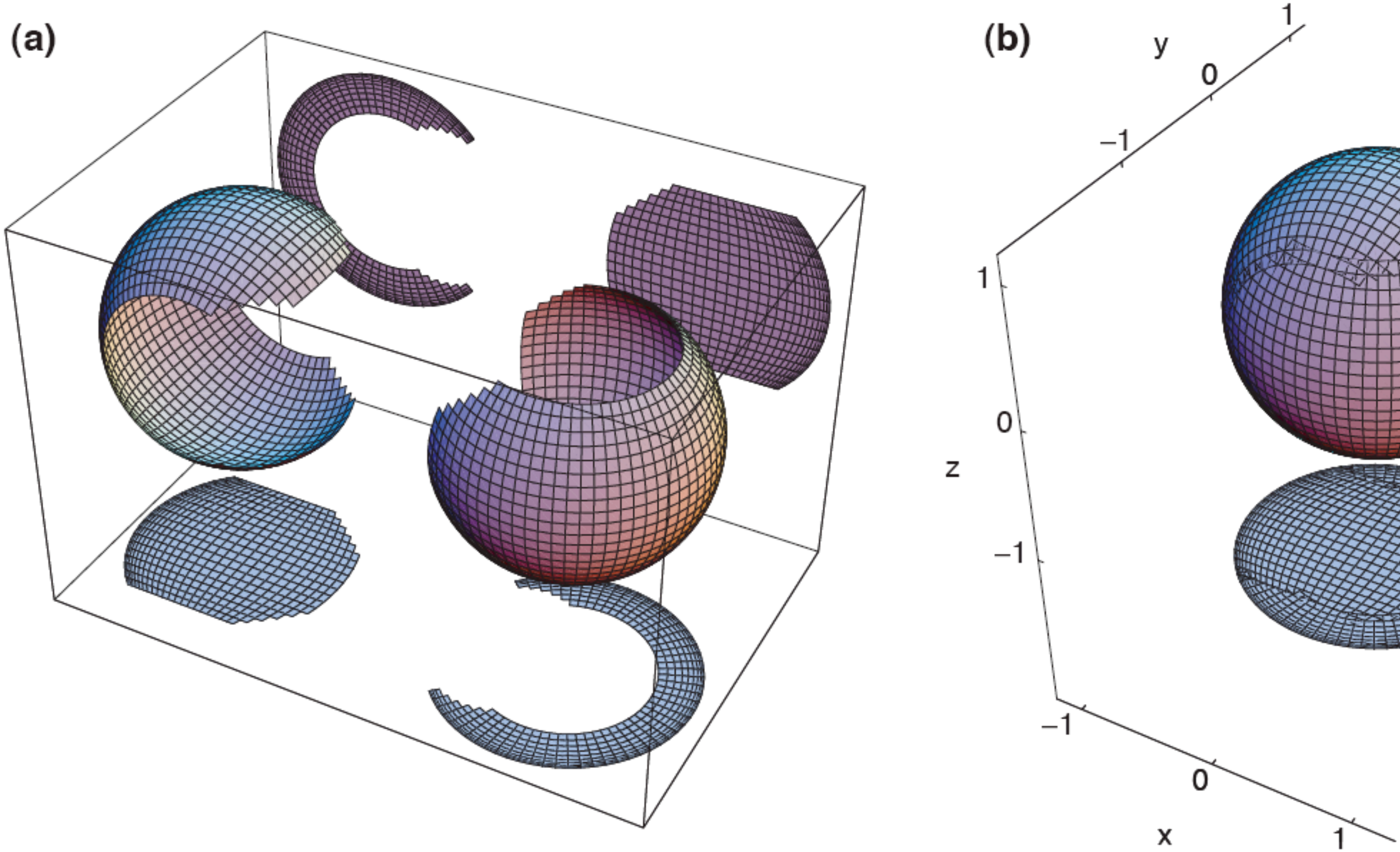
‘Yin-Yang’ grid (Kageyama, JAMSTEC ESC)



- Orthogonal => simple finite-differences possible
- Overlapping region (6% of total)

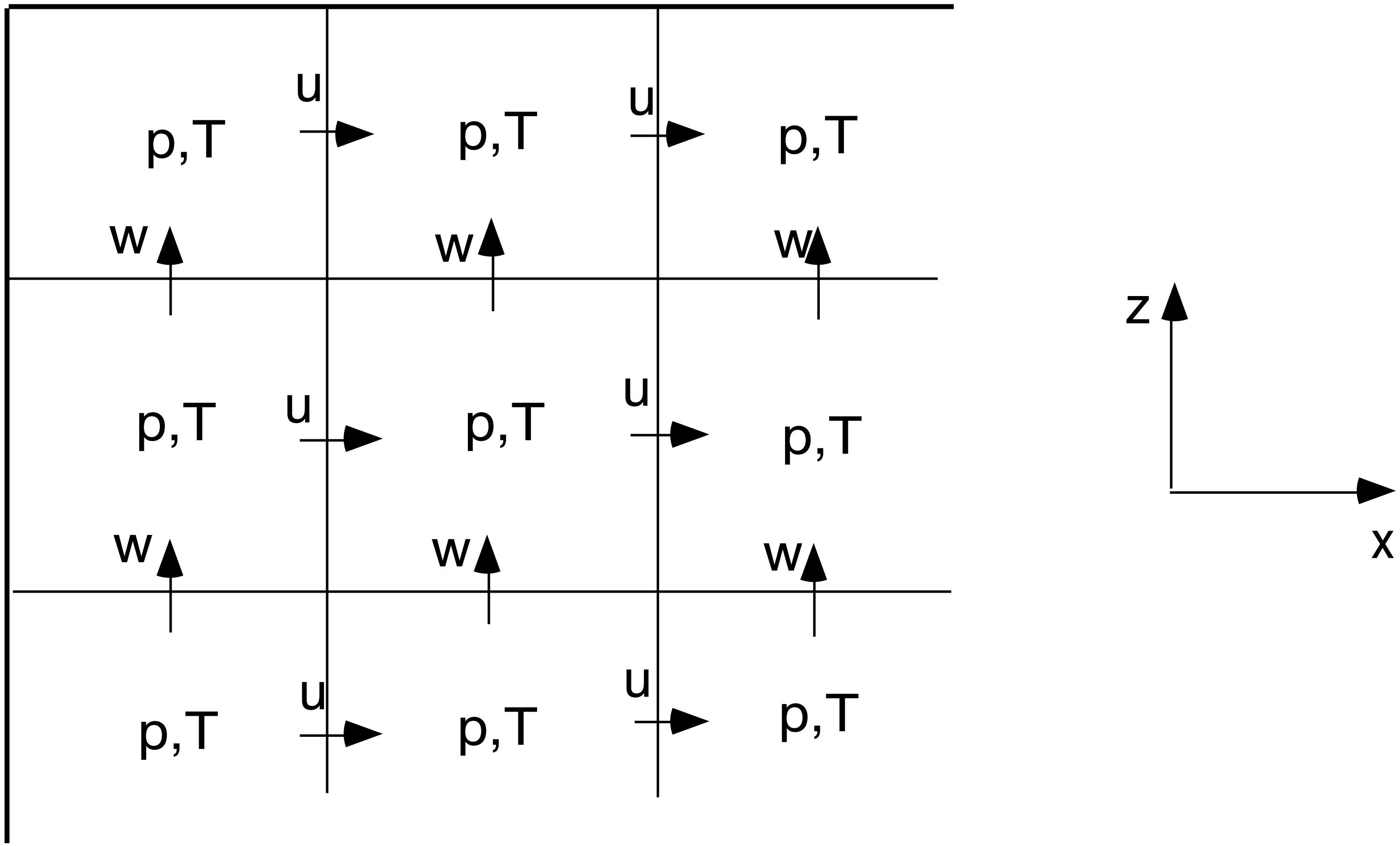


Minimum overlap YY grid



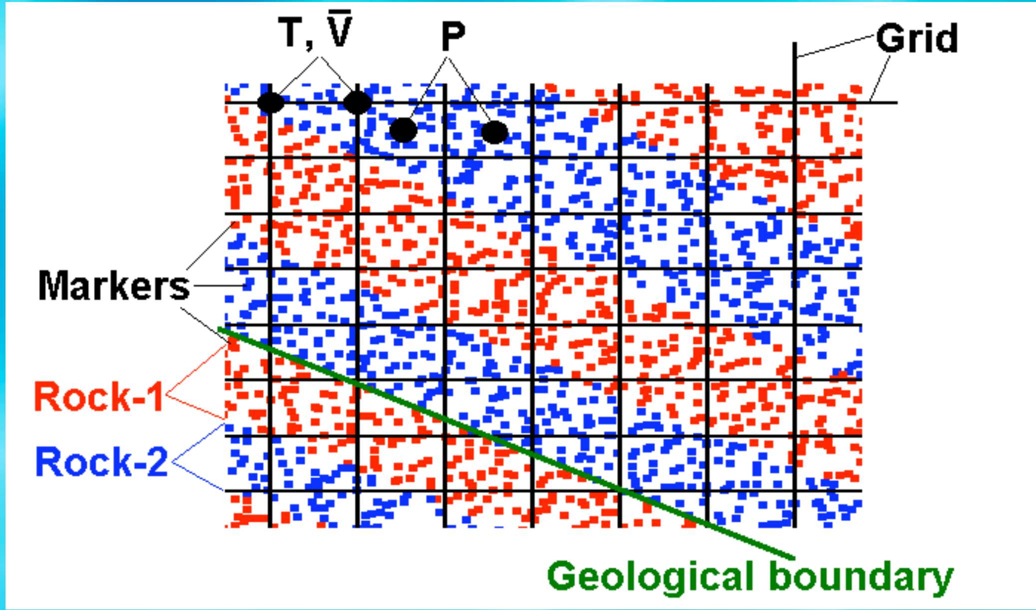
- Eliminates differing solutions in overlap
- Jagged boundaries of subgrids

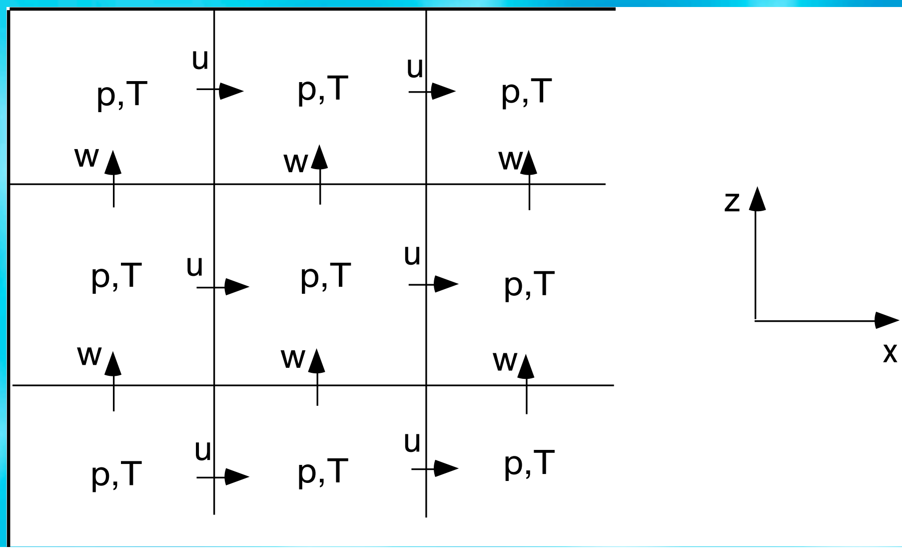
Staggered grid primitive variables



Compositional treatment uses tracers

- Track composition on Lagrangian tracers
- (Eulerian grid, as before)





Truncated anelastic equations

Conservation of mass:

$$\nabla \cdot (\rho \underline{v}) = 0 \quad , \tag{1}$$

momentum

$$\underline{\nabla} \cdot \underline{\underline{\sigma}} - \underline{\nabla} p = Ra \,.\, \hat{\underline{r}} . \rho(C, r, T) / \Delta\rho_{thermal} \tag{2}$$

and energy

$$\rho C_p \frac{DT}{Dt} = -Di_s \alpha \rho T v_r + \underline{\nabla} \bullet (k \nabla T) + \rho H + \frac{Di_s}{Ra} \underline{\underline{\sigma}} : \underline{\underline{\dot{\varepsilon}}} \tag{3}$$

In cases where bulk chemistry is treated the following must also be satisfied:

$$\frac{DC}{Dt} = 0$$

Spherical stress divergences

$$\left(\nabla \bullet \underline{\underline{\sigma}}\right)_r = -\frac{\partial p}{\partial r} + \frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2 \tau_{rr}\right) + \frac{1}{r\sin\theta}\frac{\partial}{\partial\theta}\left(\tau_{r\theta}\sin\theta\right) + \frac{1}{r\sin\theta}\frac{\partial \tau_{r\phi}}{\partial\phi} - \frac{\tau_{\theta\theta} + \tau_{\phi\phi}}{r} \tag{5}$$

$$\left(\nabla \bullet \underline{\underline{\sigma}}\right)_\theta = -\frac{1}{r}\frac{\partial p}{\partial \theta} + \frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2 \tau_{r\theta}\right) + \frac{1}{r\sin\theta}\frac{\partial}{\partial\theta}\left(\tau_{\theta\theta}\sin\theta\right) + \frac{1}{r\sin\theta}\frac{\partial \tau_{\phi\theta}}{\partial\phi} + \frac{1}{r}\left(\tau_{r\theta} - \tau_{\phi\phi}\cot\theta\right) \tag{6}$$

$$\left(\nabla \bullet \underline{\underline{\sigma}}\right)_\phi = -\frac{1}{r\sin\theta}\frac{\partial p}{\partial \phi} + \frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2 \tau_{r\phi}\right) + \frac{1}{r\sin\theta}\frac{\partial}{\partial\theta}\left(\tau_{\theta\phi}\sin\theta\right) + \frac{1}{r\sin\theta}\frac{\partial \tau_{\phi\phi}}{\partial\phi} + \frac{1}{r}\left(\tau_{r\phi} + \tau_{\theta\phi}\,\mathrm{co}\right. \tag{7}$$

Iteration procedure (velocity/ pressure)

- Pointwise (~like Patankar's SIMPLER)
  - Update x-velocities
  - Update y-velocities
  - Update z-velocities
  - Update pressure to reduce div.v
- Cellwise ('pressure coupled')
  - Solve pressure + 6 surrounding v components simultaneously
  - Converges better but slower
  - Not yet implemented in new version

A subtlety occurs in the treatment of normal strain rates (hence normal stresses) when density is spatially varying, i.e., for compressible cases. The divergence of velocity is then non-zero and the expressions for normal strain rate contain a $-\frac{1}{3}\nabla\cdot\underline{v}$. If this is calculated literally from the velocities, then instabilities can occur in an iterative solution procedure, because $\nabla\cdot\underline{v}$ can be incorrectly very high or low during early iterations. Thus, it is better to recognise that:

$$\nabla\cdot(\rho\underline{v}) = 0 = \rho\nabla\cdot\underline{v} + \underline{v}\cdot\nabla\rho \qquad \Rightarrow \qquad \nabla\cdot\underline{v} = \frac{-\underline{v}\cdot\nabla\rho}{\rho} \tag{8}$$

and use $\underline{v}\cdot\nabla\rho/\rho$ in the strain rate expressions instead of $\nabla\cdot\underline{v}$, because velocities is more reliable than gradients of velocity. This simply appears as an extra term in the calculation of the finite-difference stencils.

Iterations: details

Velocity correction

$$\delta v^{\theta}_{i-.5jk} = -\alpha_m R^{\theta mom}_{i-.5jk} \Bigg/ \left(\frac{\partial R^{\theta mom}_{i-.5jk}}{\partial v^{\theta}_{i-.5jk}} \right)$$

Pressure correction
(to reduce divergence)

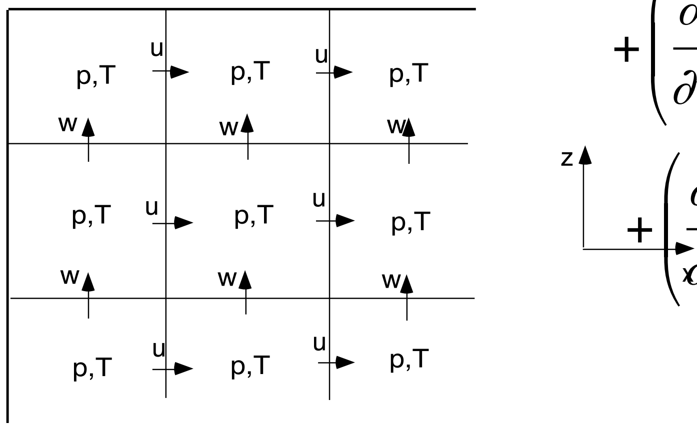
$$\delta P_{ijk} = -\alpha_c \, R^{cont}_{ijk} \Bigg/ \left(\frac{\partial R^{cont}_{ijk}}{\partial P_{ijk}} \right)$$

Velocity update
for pressure
correction

$$\delta v^{\theta}_{i-.5jk} = \left(\delta P_{i-1jk} \left(\frac{\partial R^{\theta mom}_{i-.5jk}}{\partial P_{i-1jk}} \right) + \delta P_{ijk} \left(\frac{\partial R^{\theta mom}_{i-.5jk}}{\partial P_{ijk}} \right) \right) \Bigg/ \left(\frac{\partial R^{\theta mom}_{i-.5jk}}{\partial v^{\theta}_{i-.5jk}} \right)$$

Calculation of dR/dP

$$
\begin{aligned}
\left(\frac{\partial R^{cont}_{ijk}}{\partial P_{ijk}}\right) \approx\; & \left(\frac{\partial R^{cont}_{ijk}}{\partial v^{\theta}_{i+.5jk}}\right)\left(\frac{\partial R^{\theta mom}_{i+.5jk}}{\partial P_{ijk}}\right)\Bigg/\left(\frac{\partial R^{\theta mom}_{i+.5jk}}{\partial v^{\theta}_{i+.5jk}}\right) + \left(\frac{\partial R^{cont}_{ijk}}{\partial v^{\theta}_{i-.5jk}}\right)\left(\frac{\partial R^{\theta mom}_{i-.5jk}}{\partial P_{ijk}}\right)\Bigg/\left(\frac{\partial R^{\theta mom}_{i-.5jk}}{\partial v^{\theta}_{i-.5jk}}\right) \\
& + \left(\frac{\partial R^{cont}_{ijk}}{\partial v^{\phi}_{ij+.5k}}\right)\left(\frac{\partial R^{\phi mom}_{ij+.5k}}{\partial P_{ijk}}\right)\Bigg/\left(\frac{\partial R^{\phi mom}_{ij+.5k}}{\partial v^{\phi}_{ij+.5k}}\right) + \left(\frac{\partial R^{cont}_{ijk}}{\partial v^{\phi}_{ij-.5k}}\right)\left(\frac{\partial R^{\phi mom}_{ij-.5k}}{\partial P_{ijk}}\right)\Bigg/\left(\frac{\partial R^{\phi mom}_{ij-.5k}}{\partial v^{\phi}_{ij-.5k}}\right) \\
& + \left(\frac{\partial R^{cont}_{ijk}}{\partial v^{r}_{ijk+.5}}\right)\left(\frac{\partial R^{rmom}_{ijk+.5}}{\partial P_{ijk}}\right)\Bigg/\left(\frac{\partial R^{rmom}_{ijk+.5}}{\partial v^{r}_{ijk+.5}}\right) + \left(\frac{\partial R^{cont}_{ijk}}{\partial v^{r}_{ijk-.5}}\right)\left(\frac{\partial R^{rmom}_{ijk-.5}}{\partial P_{ijk}}\right)\Bigg/\left(\frac{\partial R^{rmom}_{ijk-.5}}{\partial v^{r}_{ijk-.5}}\right)
\end{aligned}
$$



A quick examination of $(\partial R^{cont}/\partial P)$ reveals that it scales as 1/viscosity, as follows. If h represents grid spacing, then $(\partial R^{cont}/\partial v) \approx 1/h$, $(\partial R^{mom}/\partial P) \approx 1/h$, and $(\partial R^{mom}/\partial v) \approx \eta/h^2$. Thus, the pressure correction in a cell can be approximated as $-\eta\nabla\cdot(\rho\underline{v})$, which was what was used in the original cartesian version of this code (e.g., (Tackley, 1996))

‘pseudo-compressibility’ also gives 1/viscosity factor (Kameyama)

Multigrid solvers

- Gauss-Seidel or Jacobi iterations effectively smooth short-wavelength error (residual) but long-wavelengths take a long time
- Therefore smooth the residual on grids with 2* the spacing, then 4* spacing, 8* spacing etc.
- Ideally leads to convergence in fixed #iter regardless of grid size
- Problem: if viscosity varies rapidly, not correctly represented at coarse levels => slow or non-convergence

Multigrid cycles

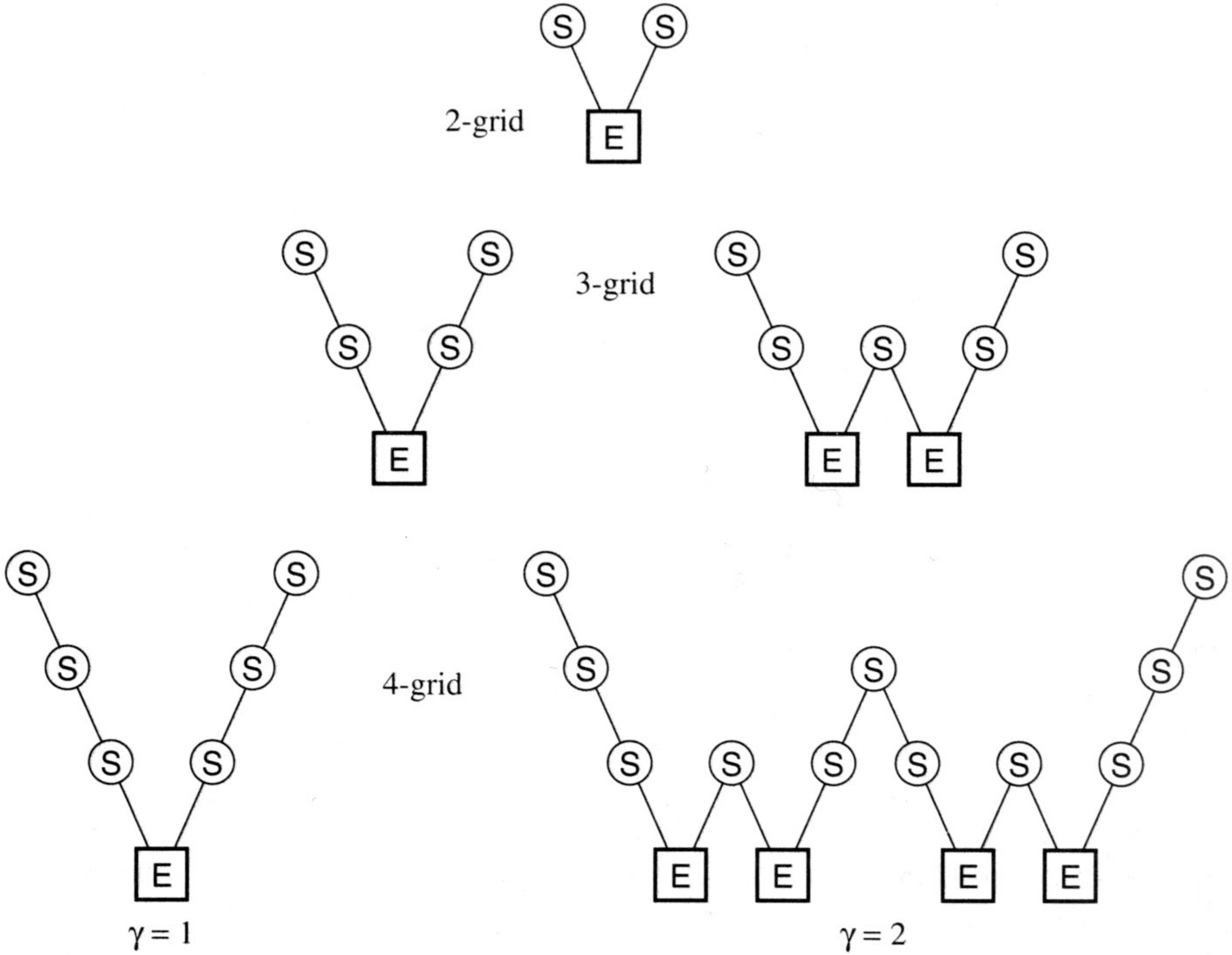


Figure 19.6.1. Structure of multigrid cycles. S denotes smoothing, while E denotes exact solution on the coarsest grid. Each descending line \ denotes restriction ($\mathcal{R}$) and each ascending line / denotes prolongation ($\mathcal{P}$). The finest grid is at the top level of each diagram. For the V-cycles ($\gamma = 1$) the E step is replaced by one 2-grid iteration each time the number of grid levels is increased by one. For the W-cycles ($\gamma = 2$), each E step gets replaced by two 2-grid iterations.

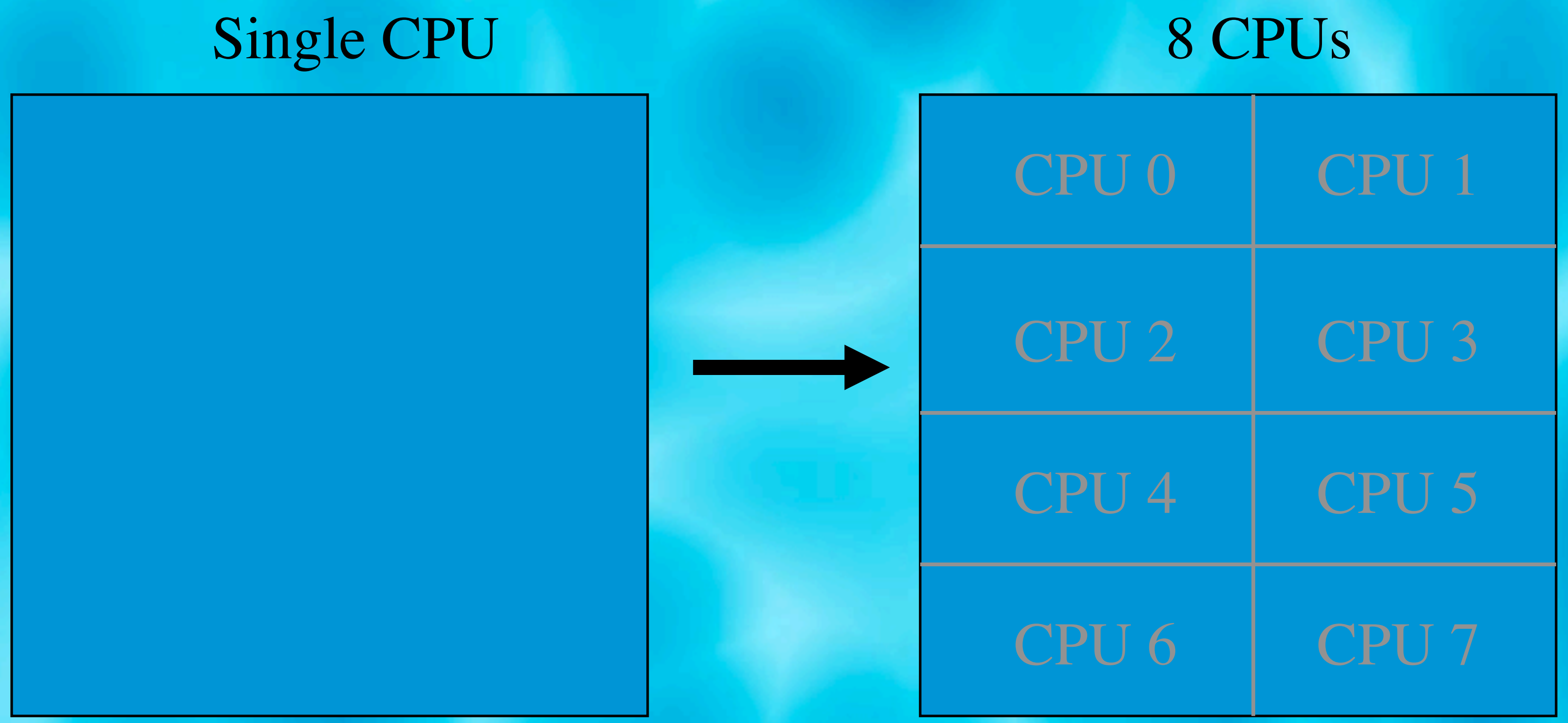
Multigrid viscous flow solvers are well established in the community

- Finite-difference const visc (potentials)
  - Sotin & Parmentier 1994: Cartesian
- Finite volume/difference, primitive variable, variable viscosity
  - Tackley 1993 (compressible)
  - Trompert&Hansen 1996: implicit T, improved viscosity restriction
  - Auth+Harder 1999: 2D, FAS, SCGS smoother
  - Albers 2000: FAS, mesh refinement
  - Hernlund+Tackley 2003: Cubed sphere (constant viscosity)
  - Kameyama 2004: Cartesian, Earth Simulator
  - Choblet 2004: Cubed sphere
  - Tackley 2006: Yin-yang sphere
- Finite-element, variable viscosity
  - TERRA (1980s-): Spherical, isocahedral
  - CITCOM (~1993): Cartesian, rectangular
  - CITCOM-S (1997?): Spherical, 8-sided elements

Parallelization

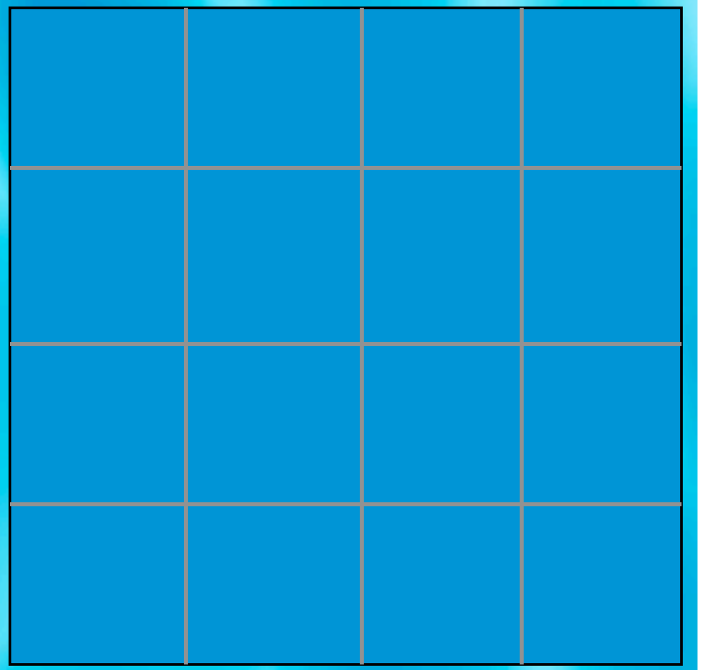
- Cartesian or single spherical block
  - Straighforward 3D domain decomposition, simple communication patterns, 100s CPUs
  - Care needed on coarse grids
- Yin-Yang sphere
  - 2 blocks on different node(s)
  - Each block divided in 4 while maintaining simple communication
  - Then decompose in radius
  - Current version up to 64 cpus.

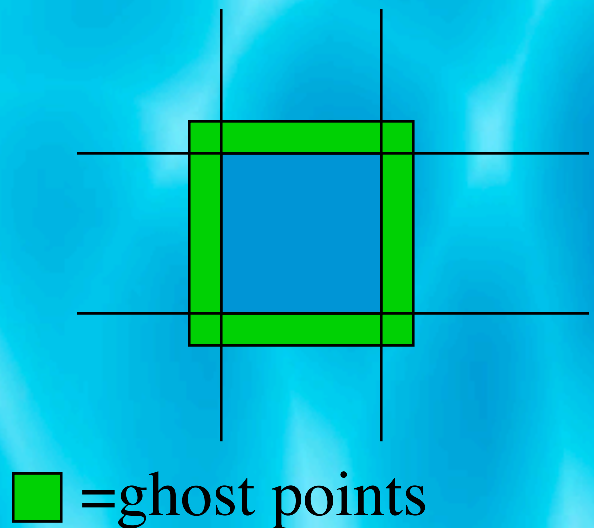
Domain decomposition



Boundaries

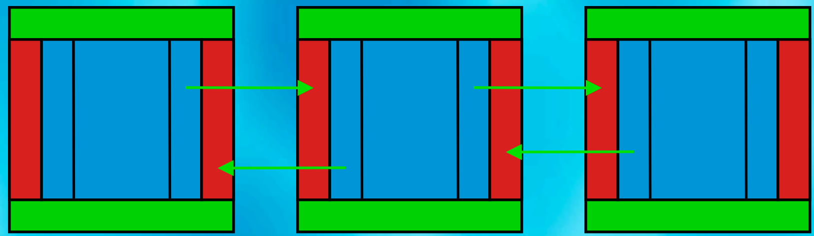
- When updating points at edge of subdomain, need values on neighboring subdomains
- Hold copies of these locally using "ghost points"
- This minimizes #of messages, because they can be updated all at once instead of individually



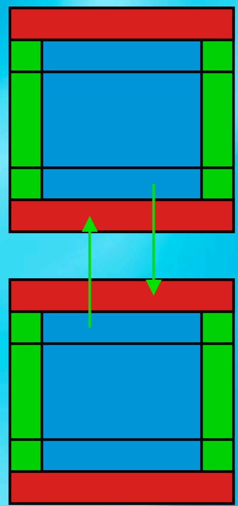


Boundary communication

Step 1: x-faces



Step 2: y-faces (including corner values from step 1)

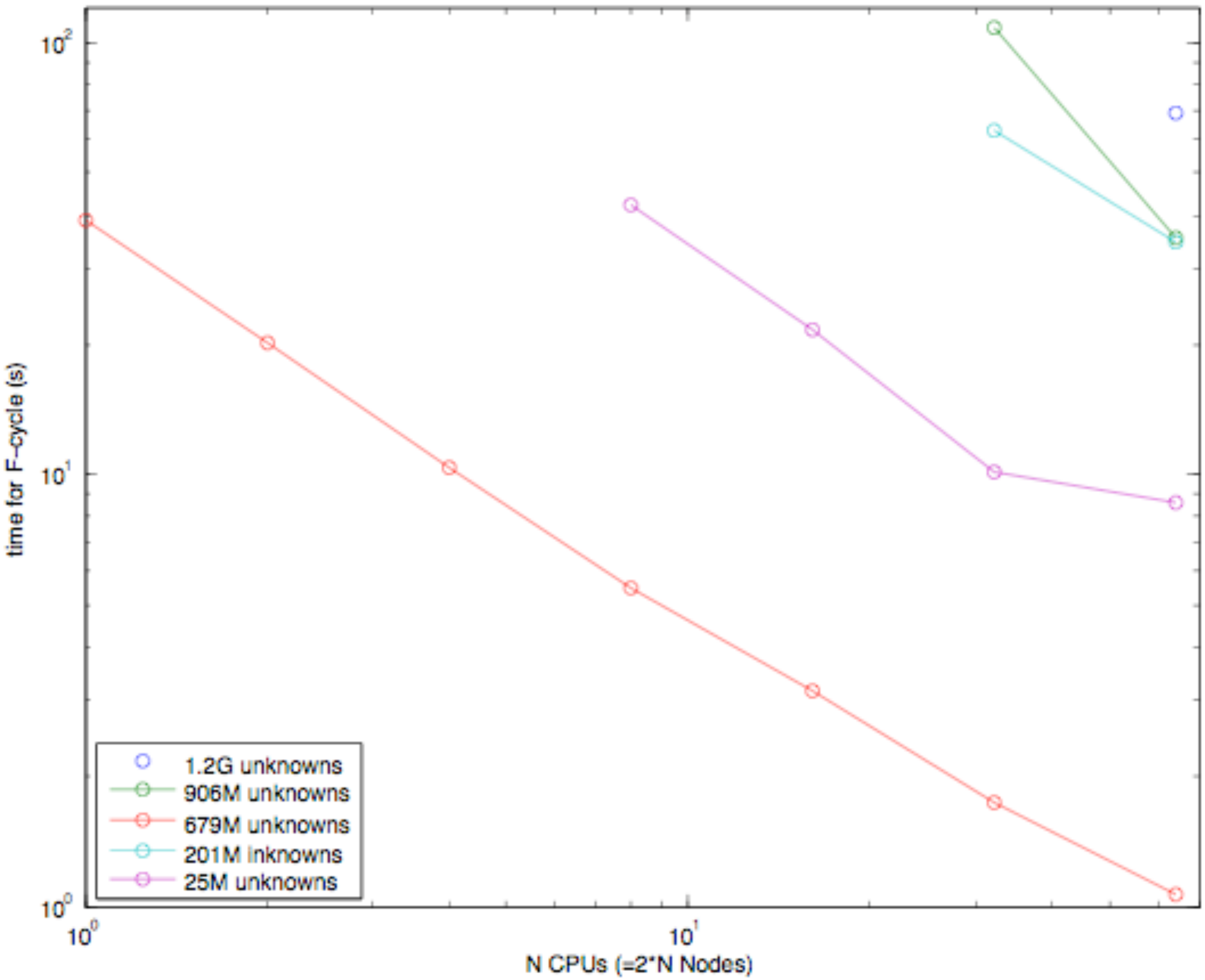


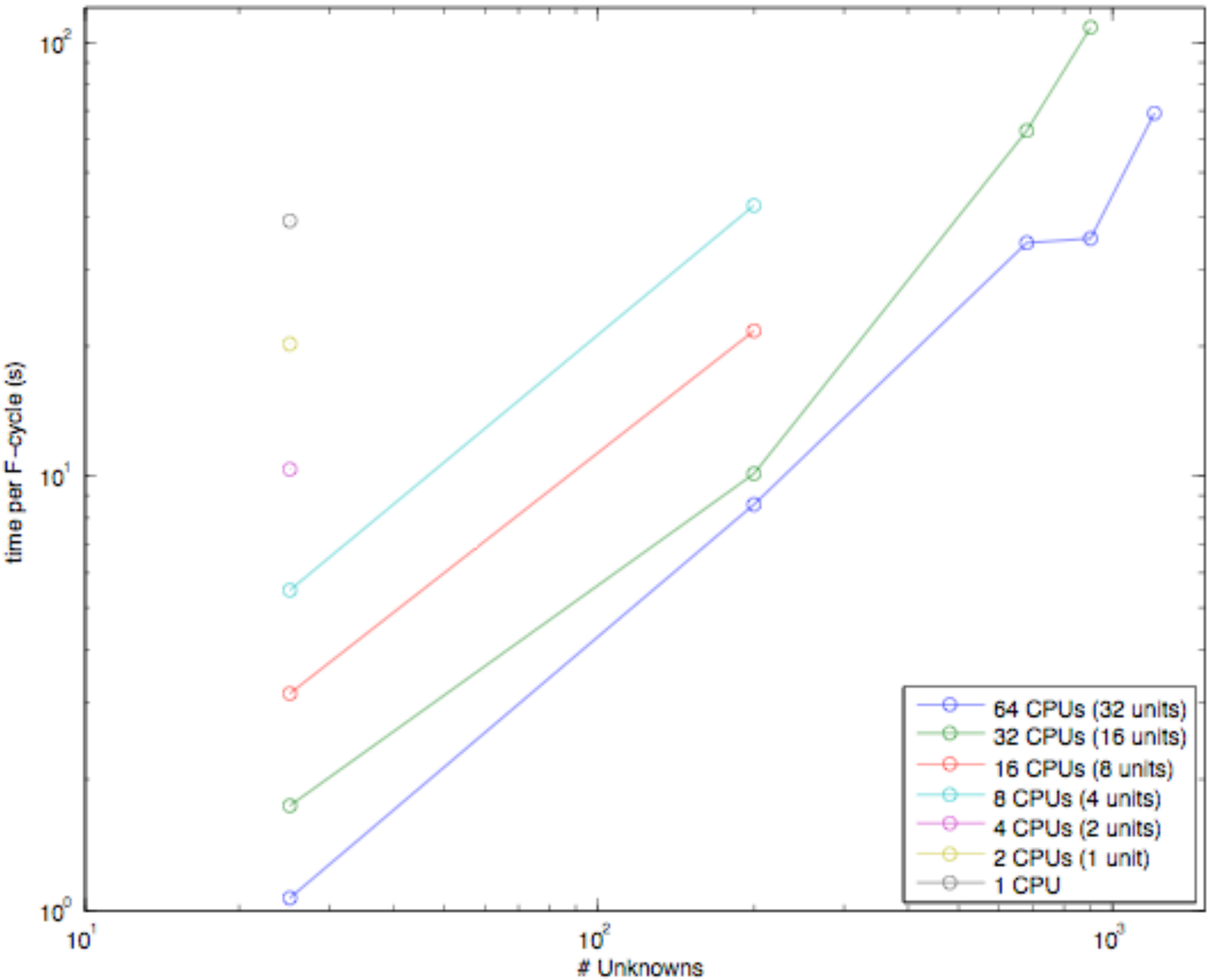
[Step 3: z-faces (including corner values from steps 1 & 2)]

Doing the 3 directions sequentially avoids the need for additional messages to do edges & corners (=>in 3D, 6 messages instead of 26)

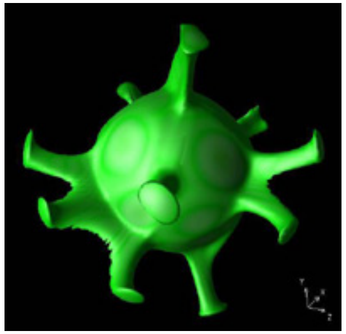
StagYY Performance

Up to 1.2 billion unknowns on only 32 nodes (64 cpus)

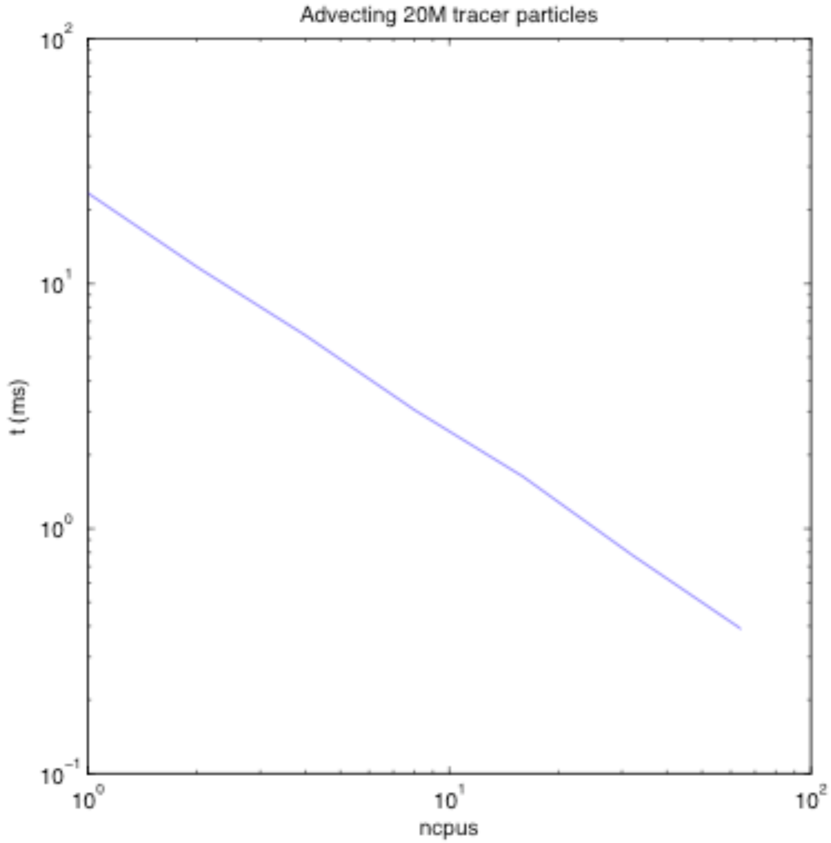


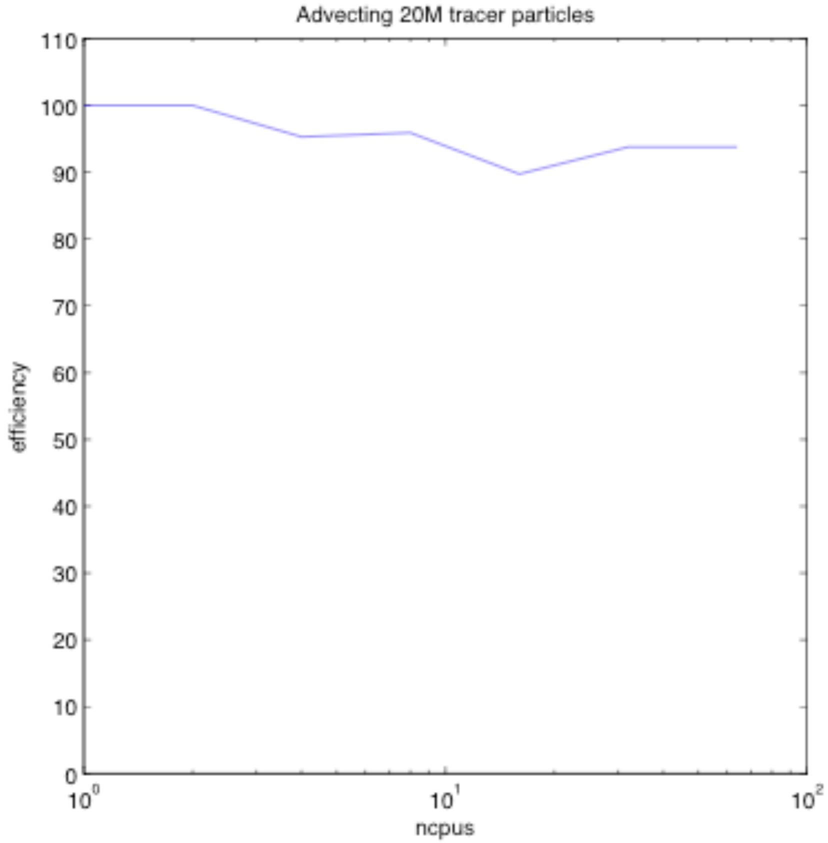


time per F–cycle (s)
Unknowns
10^0
10^1
10^2
10^3
64 CPUs (32 units)
32 CPUs (16 units)
16 CPUs (8 units)
8 CPUs (4 units)
4 CPUs (2 units)
2 CPUs (1 unit)
1 CPU



Advecting 20M tracers



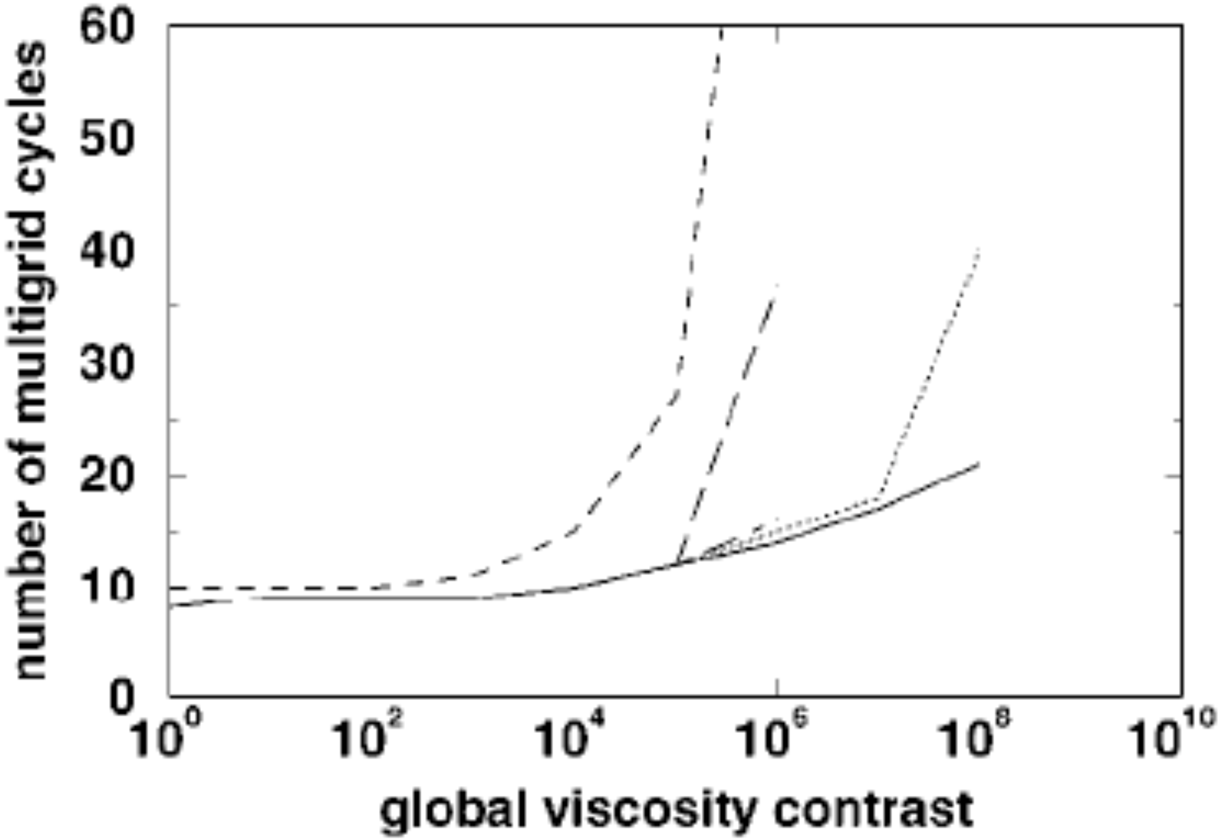


- Excellent efficiency

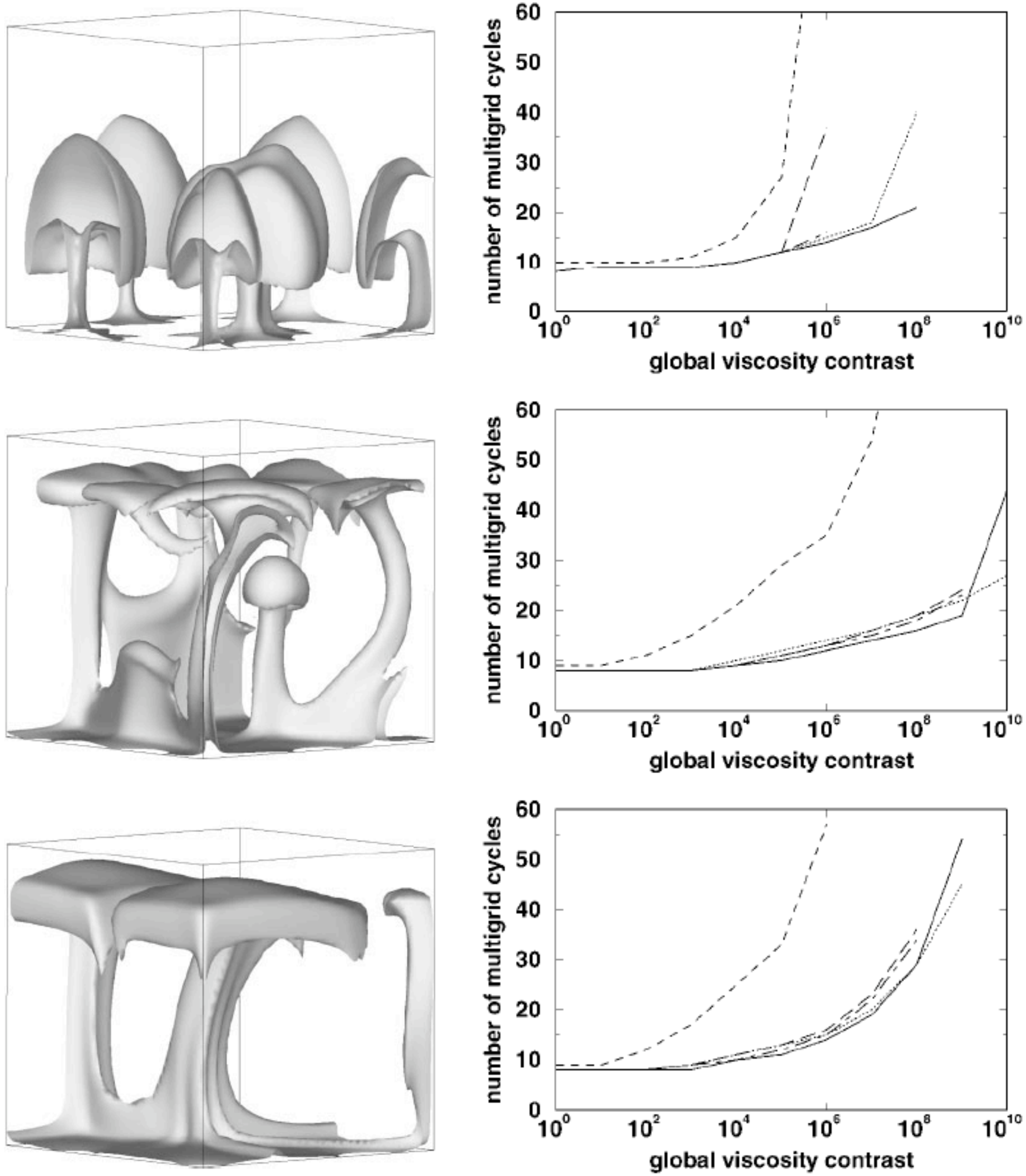
How about other aspects of performance?

- The main problem facing these codes is lack of robustness to large viscosity variations (e.g., orders of magnitude per grid point)
- Accurate treatment of non-diffusive chemical variations is also a major challenge

Problem: Not robust with large viscosity variations!



From Albers 2000
V=dashed
F=long-dashed
W=dot-dashed
Mod-V (dotted)
Mod-W (solid)



From Albers
V=dashed
F=long-dashed
W=dot-dashed
Mod-V (dotted)
Mod-W (solid)

- Convergence depends on 3D structure
- Additional coarse iterations greatly helps!

The solution: Matrix-dependent pressure prolongation

The pressure correction is ~proportional to viscosity
If fine-grid cell has much lower viscosity than coarse-grid cell, correction is much too large => divergence!

Tried weighting prolongation according to viscosity:
can help, but sometimes gets worse

Instead weight using $\left(\frac{dR}{dP}\right)_{ijk} = \frac{d(\nabla \cdot \vec{v})_{ijk}}{dP_{ijk}}$

Prolongation & restriction on staggered grid

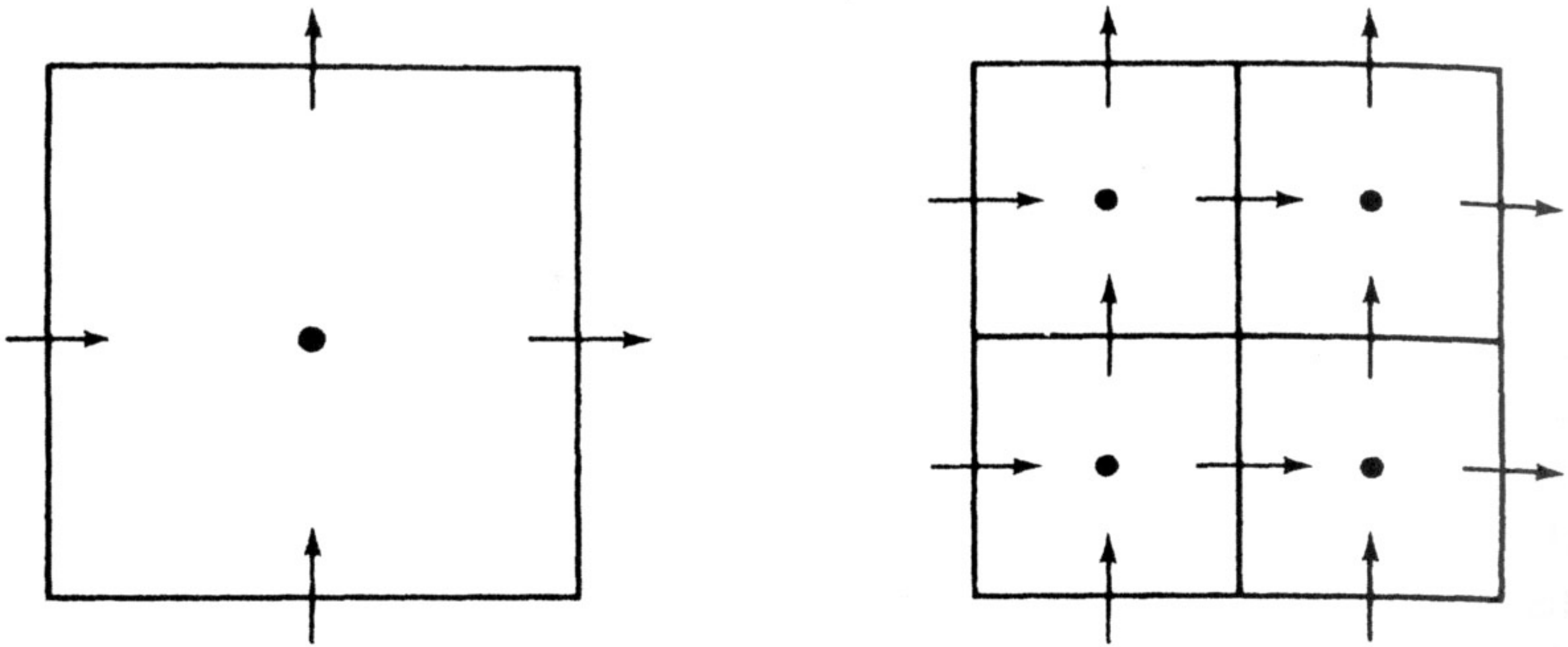


Figure 9.7.3 Coarse and fine grid cells.

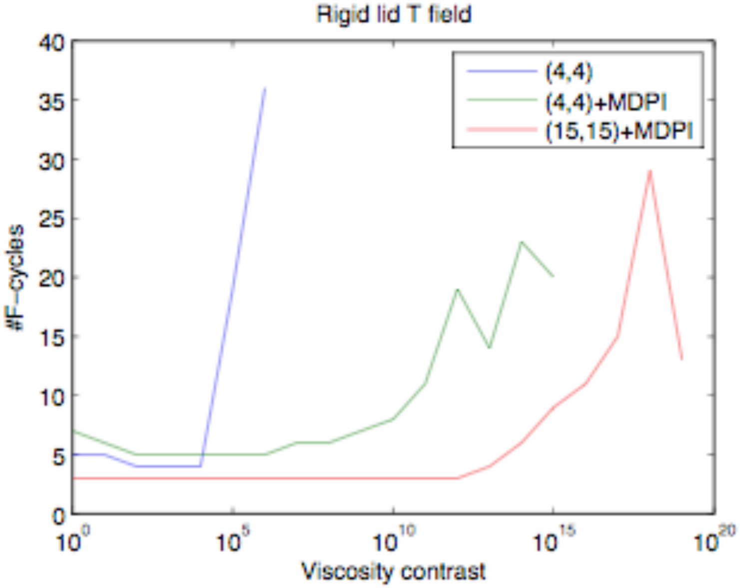
Matrix-dependent pressure prolongation scheme

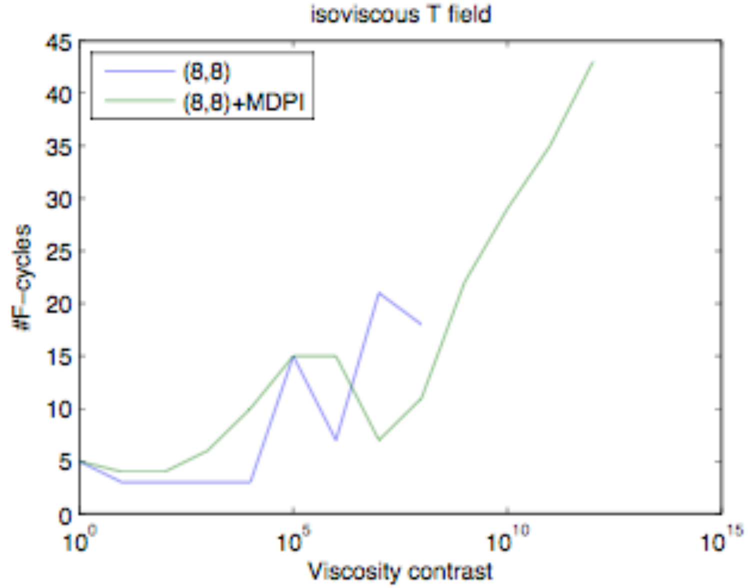
$$\delta P_{fine} = C\,\delta P_{coarse} \Big/ \left(\frac{dR_{cont}}{dP}\right)_{fine}$$

$$\frac{1}{8}\sum \delta P_{fine} = \delta P_{course}$$

$$C = 8\left(\sum 1 \Big/ \left(\frac{dR_{cont}}{dP}\right)_{fine}\right)^{-1}$$

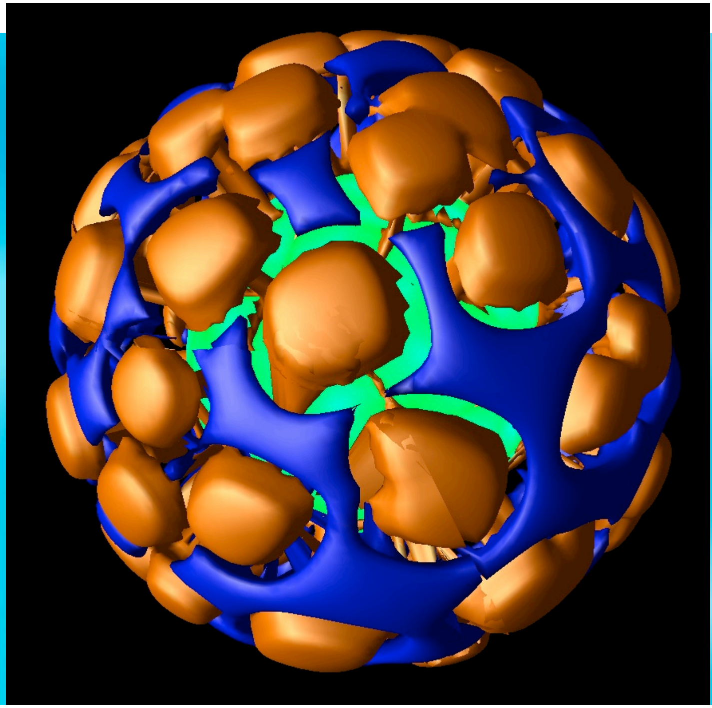
Robust for any viscosity field (so far)

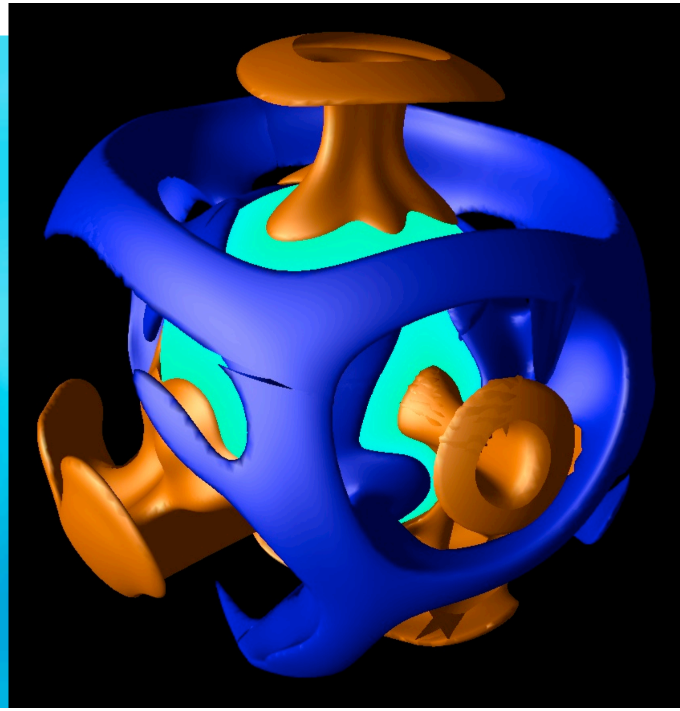




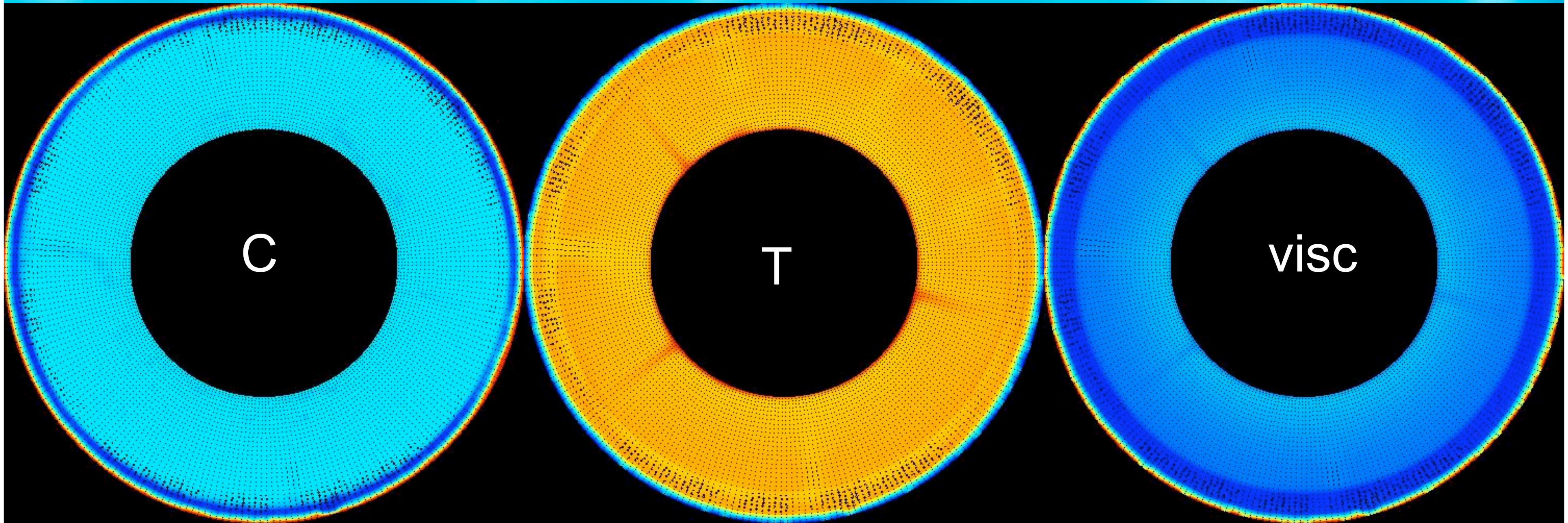
54000* between adjacent points

196000* between adjacent points





ROBUST to **large viscosity variations**

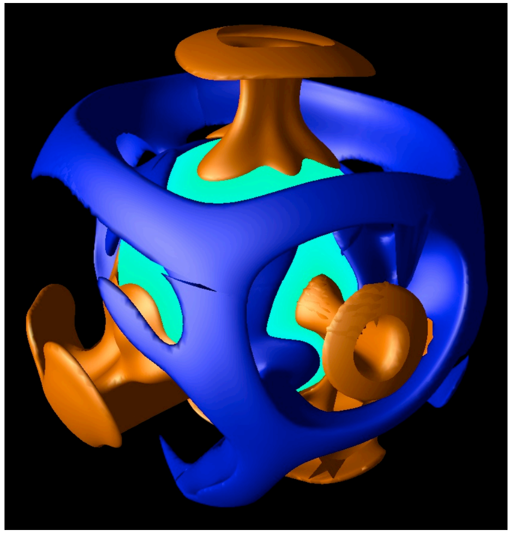


- Case above has 13+ orders of magnitude total, 6 orders between adjacent cells

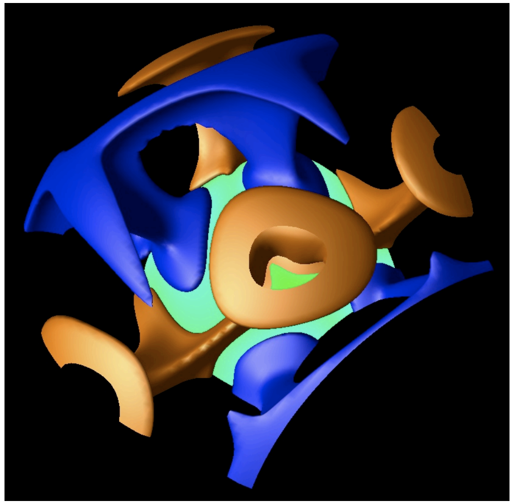
Geometries modelled

Change with single switch

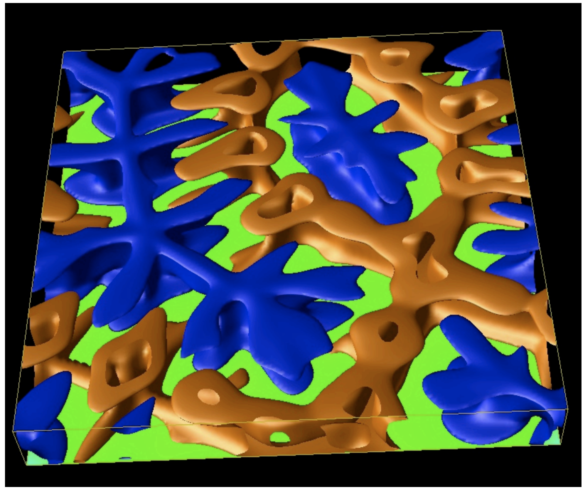
full sphere



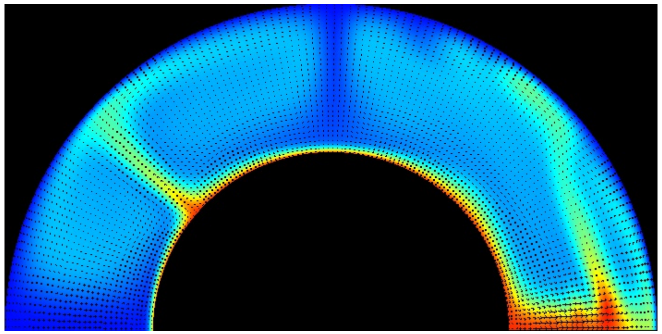
regional spherical



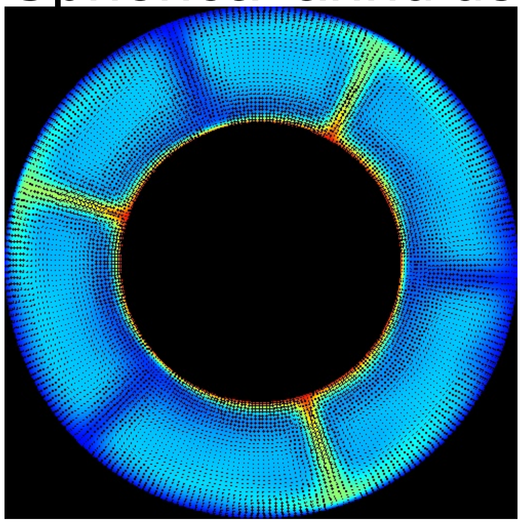
Cartesian -3D



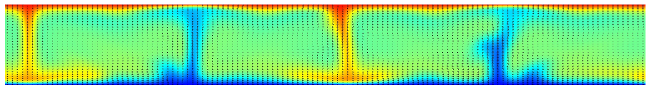
Spherical axisymmetric



Spherical annulus



-2D



2D Spherical Annulus geometry (Hernlund & Tackley, 2008)

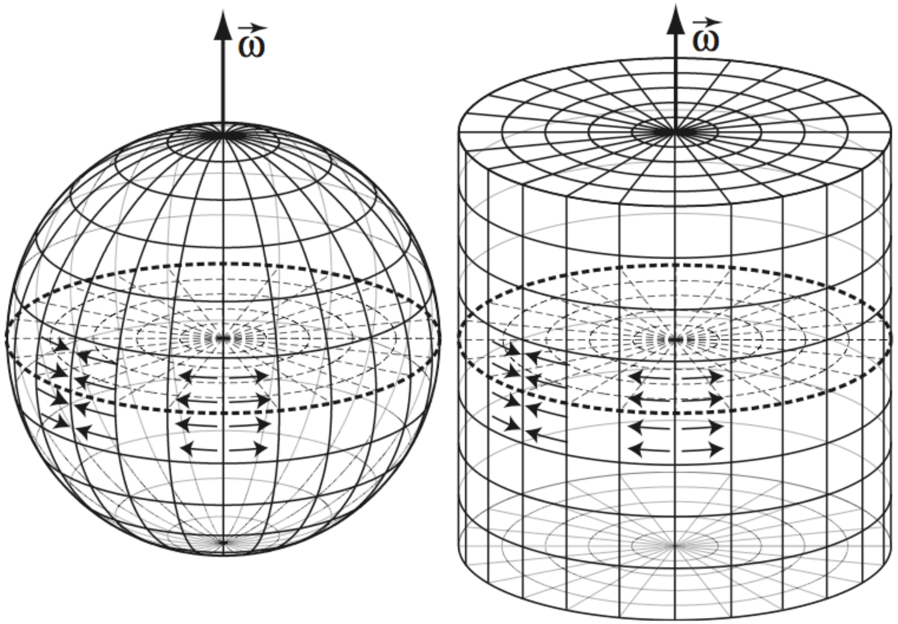


Figure 1: Comparison between the 2D approximations for rigid body translations on the surface of a sphere (left) and a cylinder (right), with the circular 2D slice of interest indicated by dashed lines. Arrows are shown to indicate a divergent motion such as that along a mid-ocean ridge as well as convergent motion such as a subduction zone setting. In both cases, the angular velocity vector $\vec{\omega}$ describing the 2D lateral motions in the slice is directed along the axes of the coordinate systems if they are taken to be oriented perpendicular to the slice. The primary difference between the two descriptions is that motions on a sphere are projected onto a surface with two degrees of curvature, while a cylinder has only one degree of curvature.

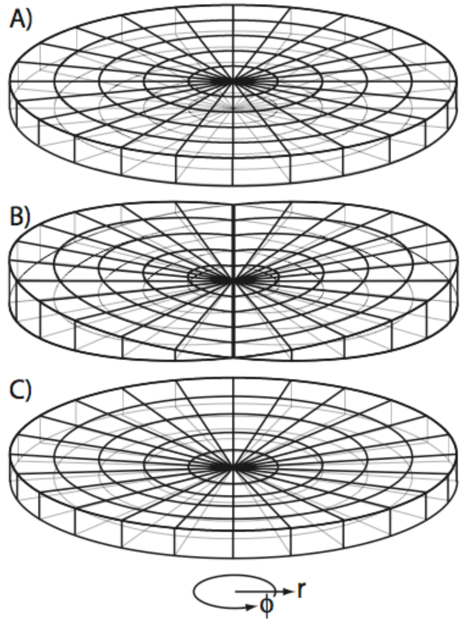


Figure 2: Illustration of what is meant by the "virtual" thickness J/r of a 2D circular slice through a 3D grid. In the constant thickness case (A), representative of a cylindrical model with effective Jacobian $J = r$, the virtual thickness is constant everywhere. For a variable thickness in the angular direction (B), representative of a spherical axi-symmetric grid with effective Jacobian $J = r^2 \sin\phi$, the virtual thickess depends on the angular location in the grid and the radius. In the variable radial thickness case (C) with effective Jacobian $J = r^2$, the virtual thickness increases with distance from the center of the grid without any angular dependence.

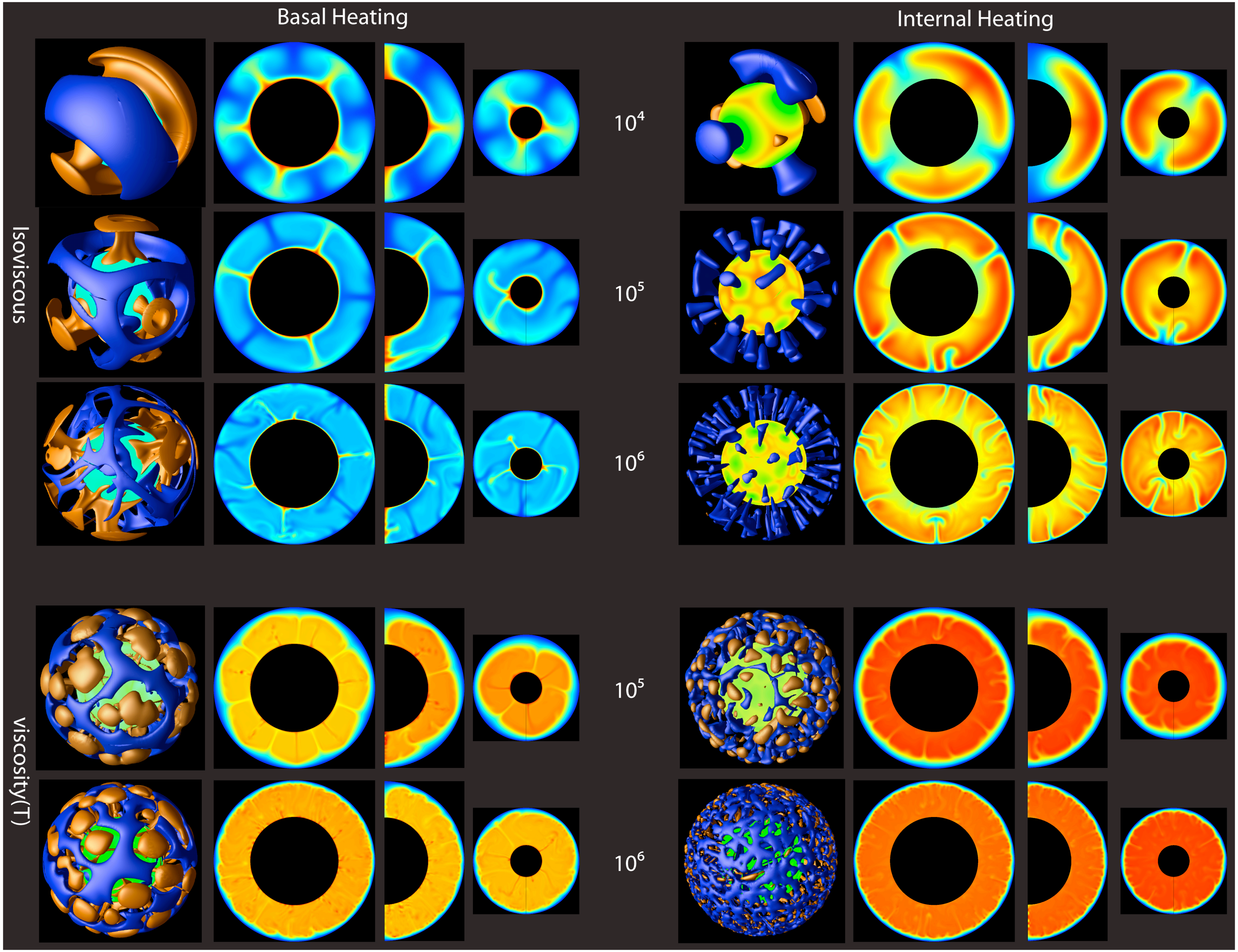

Basal Heating
Internal Heating
Isoviscous
viscosity(T)
10^4
10^5
10^6
10^5
10^6

Table 1. Basal heated, isoviscous convection

Ra	Geometry	<Nu>	$\Delta Nu_{peak-peak}$	$<V_{rms}>$	$\Delta(V_{rms})_{pk-pk}$
10^4	3D	3.85	steady	42.3	0
	annulus	4.18	steady	37.7	0
	axisymmetric	4.01	steady	41.0	0
	cylindrical	3.99	steady	35.6	0
10^5	3D	7.27	0.5	160	11
	annulus	7.39	0.3	160	14
	axisymmetric	7.26	3.2	159	100
	cylindrical	6.2	2.1	165	90
10^6	3D	15.9	1.3	625	80
	annulus	14.4	3.4	640	275
	axisymmetric	13.7	6.0	520	500
	cylindrical	14.4	5.5	613	460

Table 2. Basal heated, temperature-dependent viscosity convection

$Ra_{1/2}$	Geometry	<Nu>	$\Delta Nu^*_{peak-peak}$	$<V_{rms}>$	$\Delta(V_{rms})_{pk-pk}$
10^5	3D	6.30	0	405	5
	annulus	5.71	0.1	463	90
	axisymmetric	5.07	0.2	450	210
	cylindrical	4.97	0.1	495	200
10^6	3D	9.7	0	1804	100
	annulus	10.1	0.1	1390	780
	axisymmetric	10.45	0.1	1370	1040
	cylindrical	10.4	0.1	1850	1200

Table 3. Internally heated, isoviscous convection

Ra	Geometry	<T>	$\langle V_{rms} \rangle$	$\Delta(V_{rms})_{pk\text{-}pk}$
10^4	3D	0.311	23.3	0
	annulus	0.308	23.5	0
	axisymmetric	0.330	25.8	0
	cylindrical	0.319	22.8	0
10^5	3D	0.322	60.5	7
	annulus	0.349	78.5	36
	axisymmetric	0.357	87.0	65
	cylindrical	0.384	77.0	75
10^6	3D	0.337	180	10
	annulus	0.350	265	160
	axisymmetric	0.349	270	225
	cylindrical	0.380	268	350

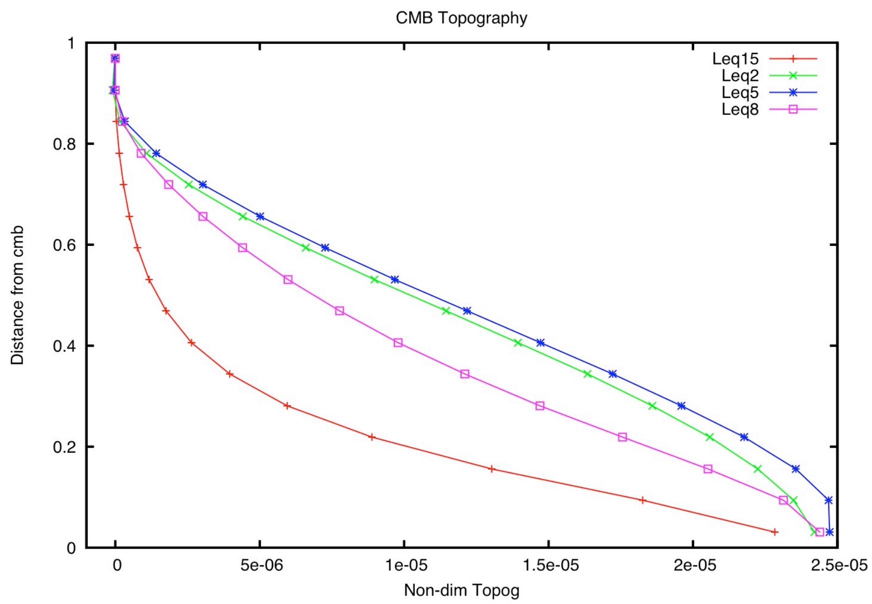
Table 4. Internally heated, temperature-dependent viscosity convection

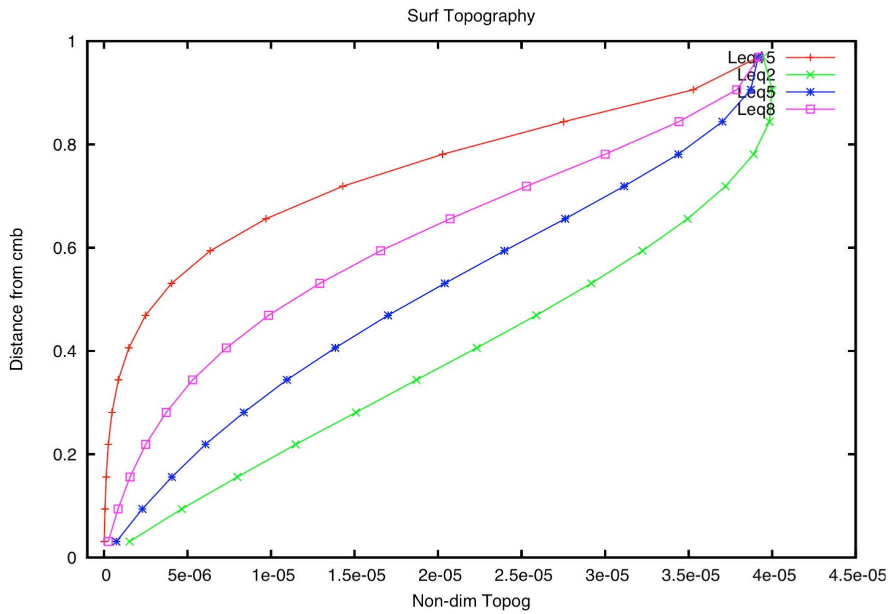
$Ra_{1/2}$	Geometry	<T>	$\langle V_{rms} \rangle$	$\Delta(V_{rms})_{pk\text{-}pk}$
10^5	3D	0.587	93	4
	annulus	0.611	135	70
	axisymmetric	0.610	142	95
	cylindrical	0.623	148	90
10^6	3D	0.665	565	65
	annulus	0.667	575	300
	axisymmetric	0.666	560	390
	cylindrical	0.690	650	430

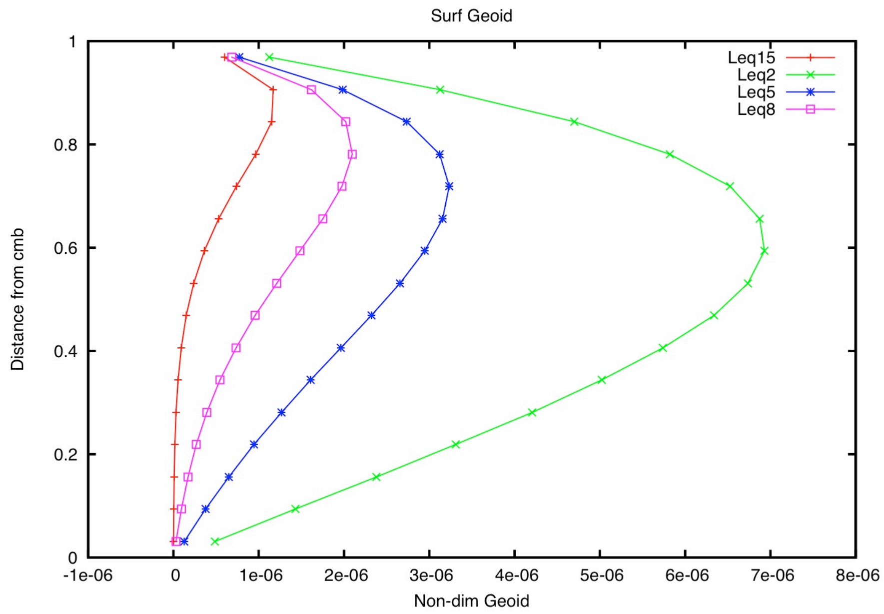
‘Advanced’ features

- Geoid
- Self-consistent mineralogy

Geoid & dynamic topography (me, Nakagawa & Stegman)



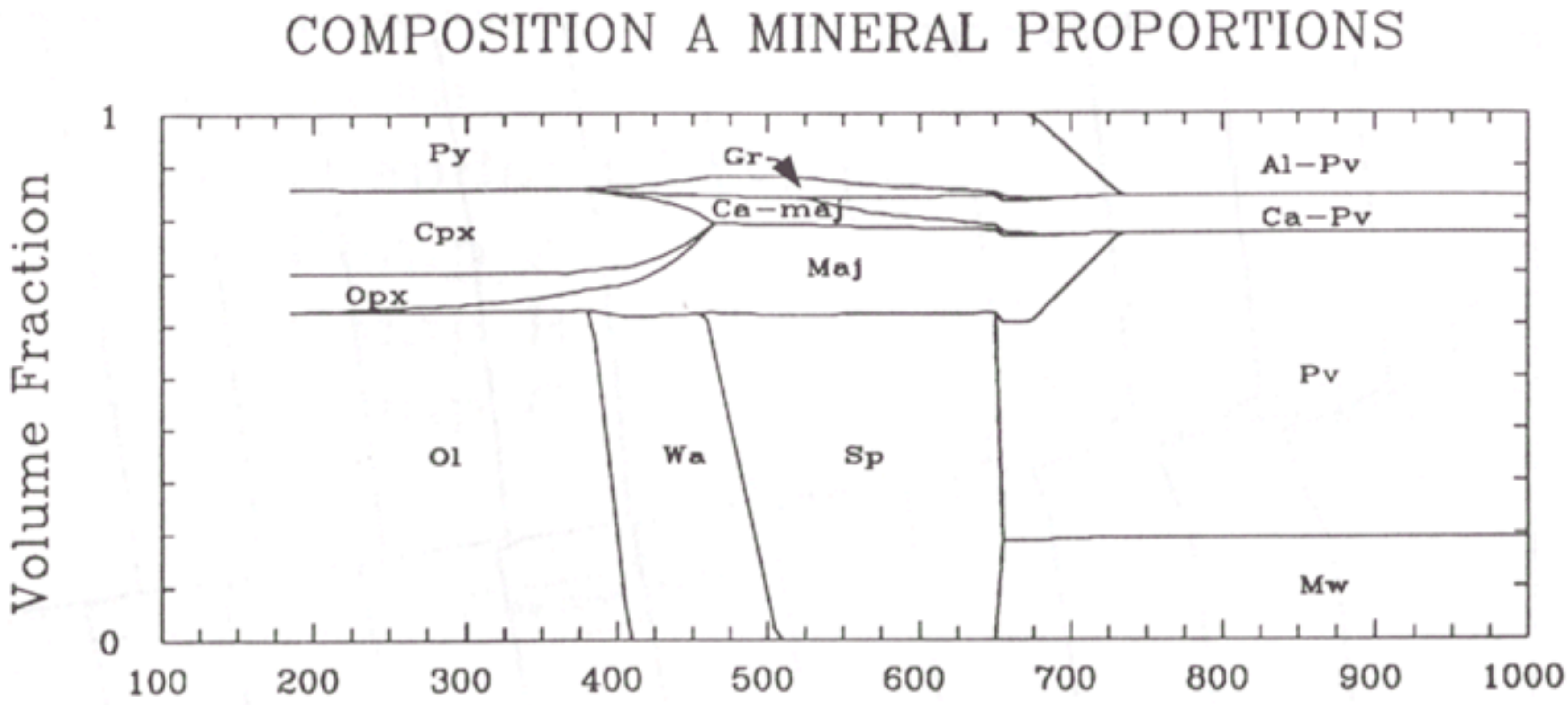




Self-consistent phase changes / mineralogy (with J. Connolly & F. Deschamps

- Mantle rocks have complicated phase diagrams that are only crudely approximated in typical convection calculations
- Phase assemblage depends on composition, temperature, pressure
- => Calculate phase assemblage and resulting physical properties by minimization of free energy using PERPLEX by J. Connolly
- Integrate into large-scale dynamical simulations of thermo-chemical convection of planets

Mineralogy: complex sequence of composition-dependent phase changes



- From Ita and Stixrude

Calculated phase relationships

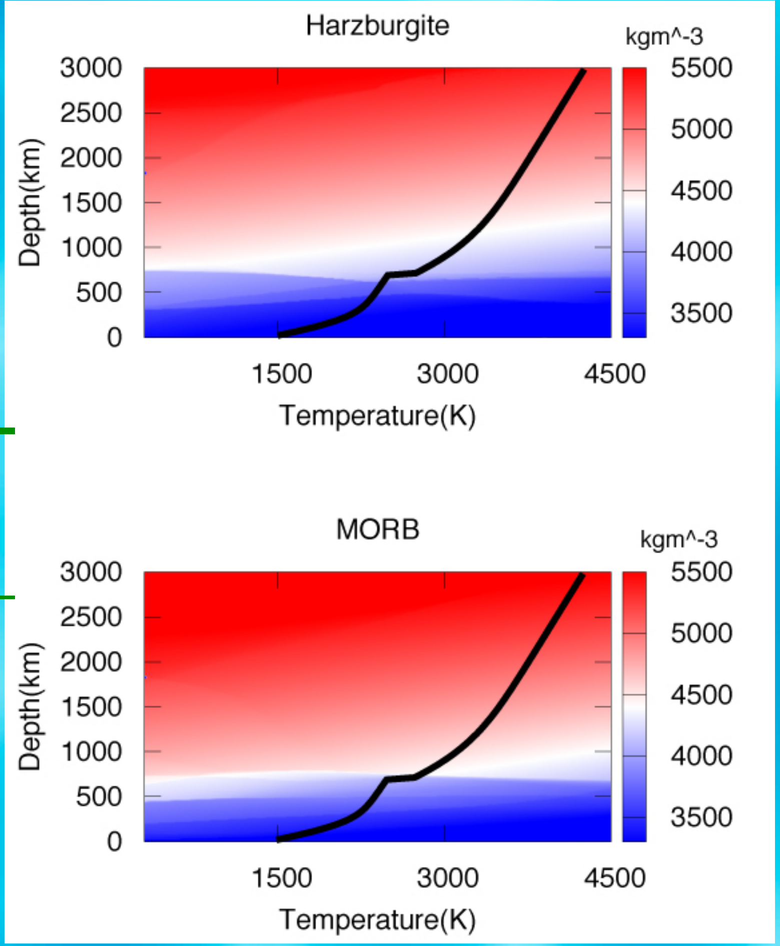
Determined by Free Energy minimization technique: PERPLEX [Connolly, 2005]

$$G(T,P) = \sum_i n_i(T,P)\mu_i(T,P)$$

Data for components for two materials from [Stixrude and Lithgow-Bertelloni, 2005]

Component	Harzburgite (mol%)	MORB (mol%)
SiO_2	36.04	41.75
MgO	57.14	22.42
FeO	5.41	6.00
CaO	0.44	13.59
Al_2O_3	0.96	16.24

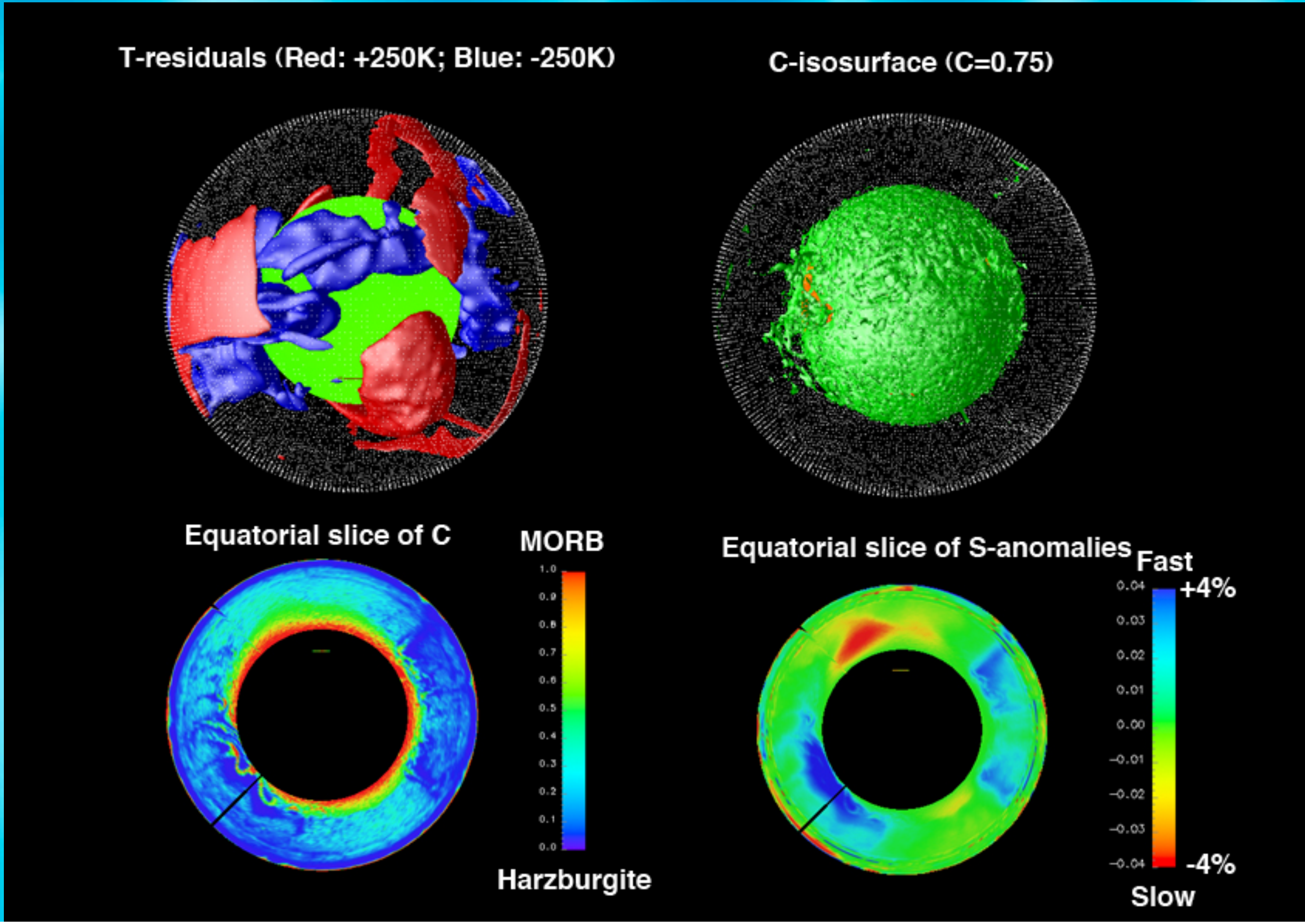
Physical properties (density)



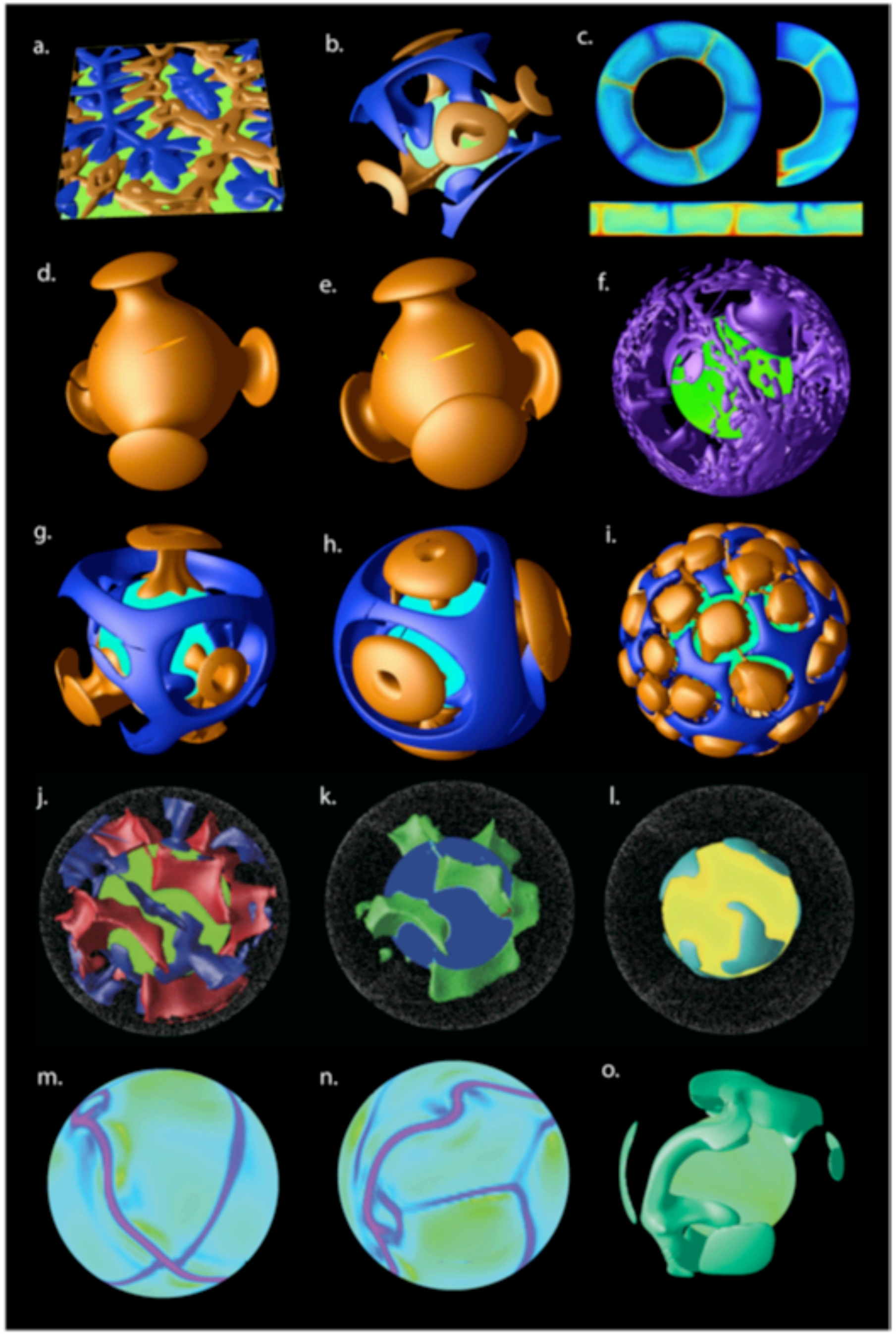
Solid line: Solidus

Numerical example: Thermo-chemical with PERPLEX properties

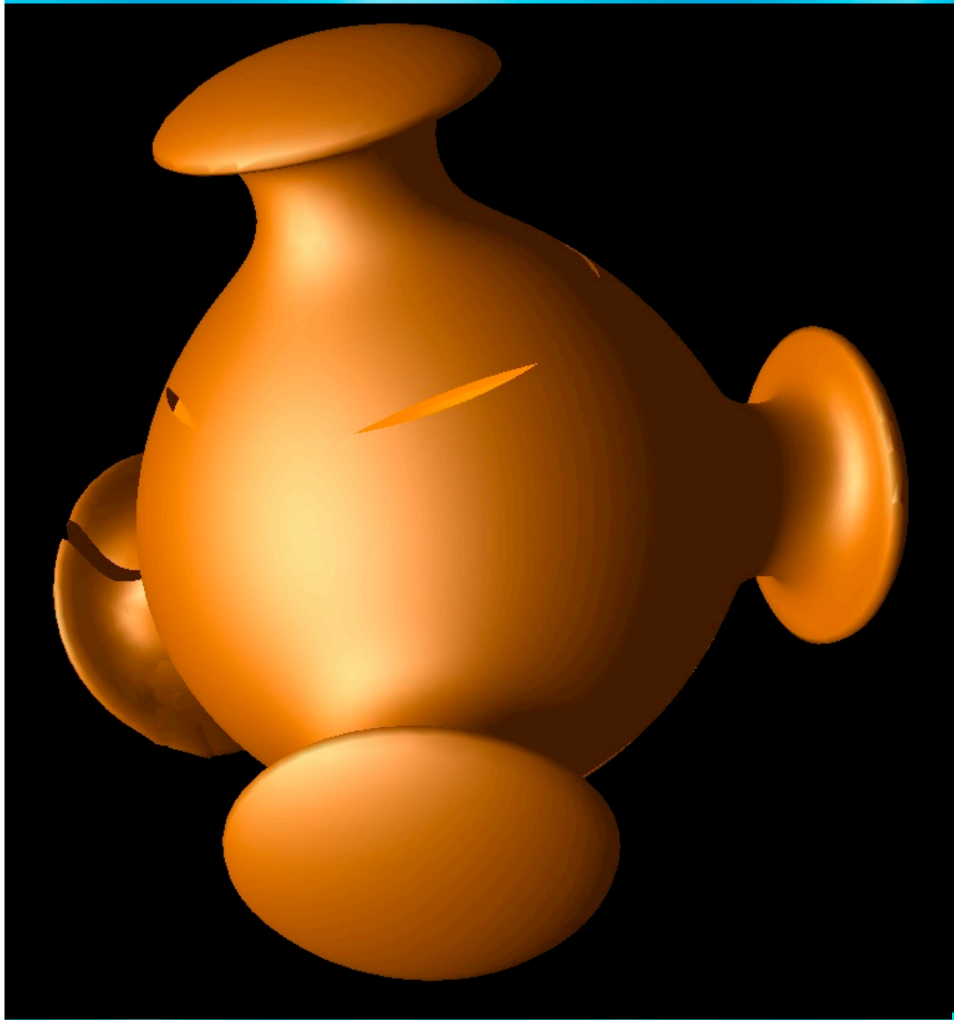
Time = 4.5Gyrs after initial state

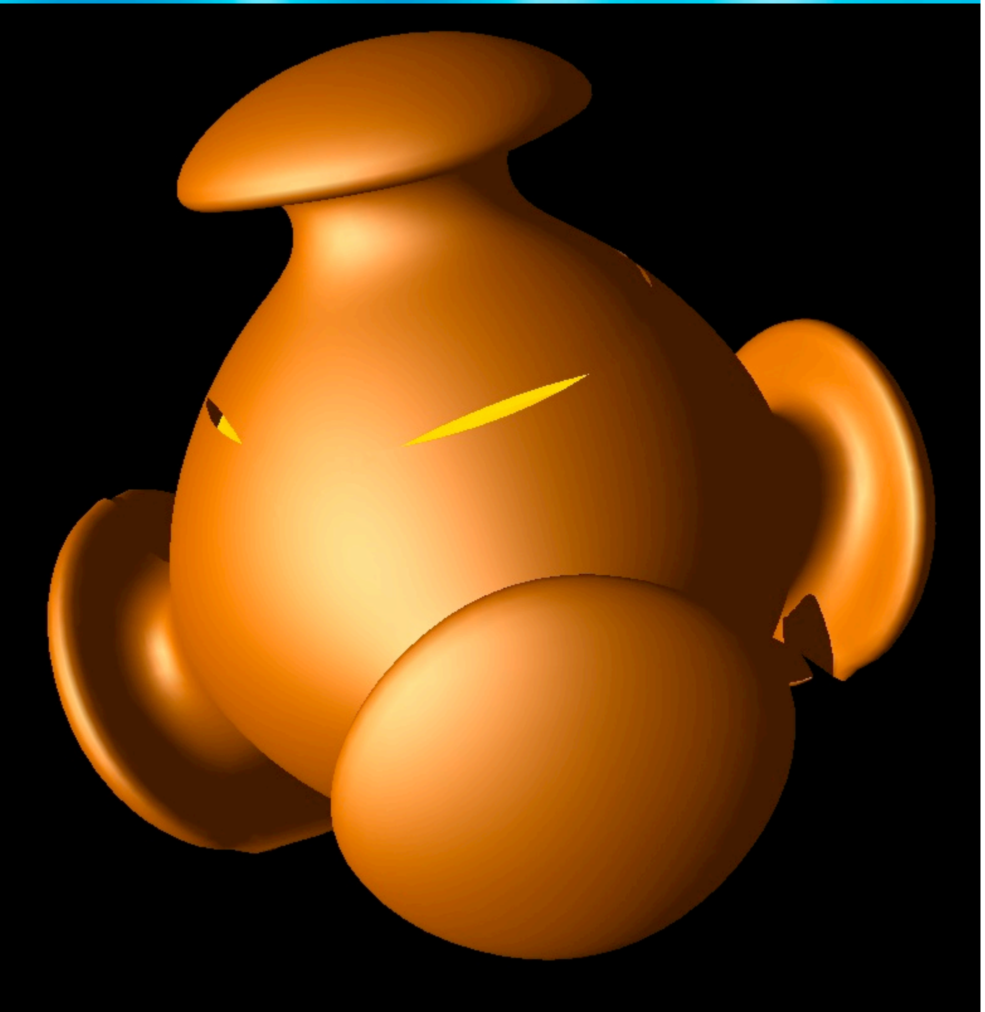


Examples of applications

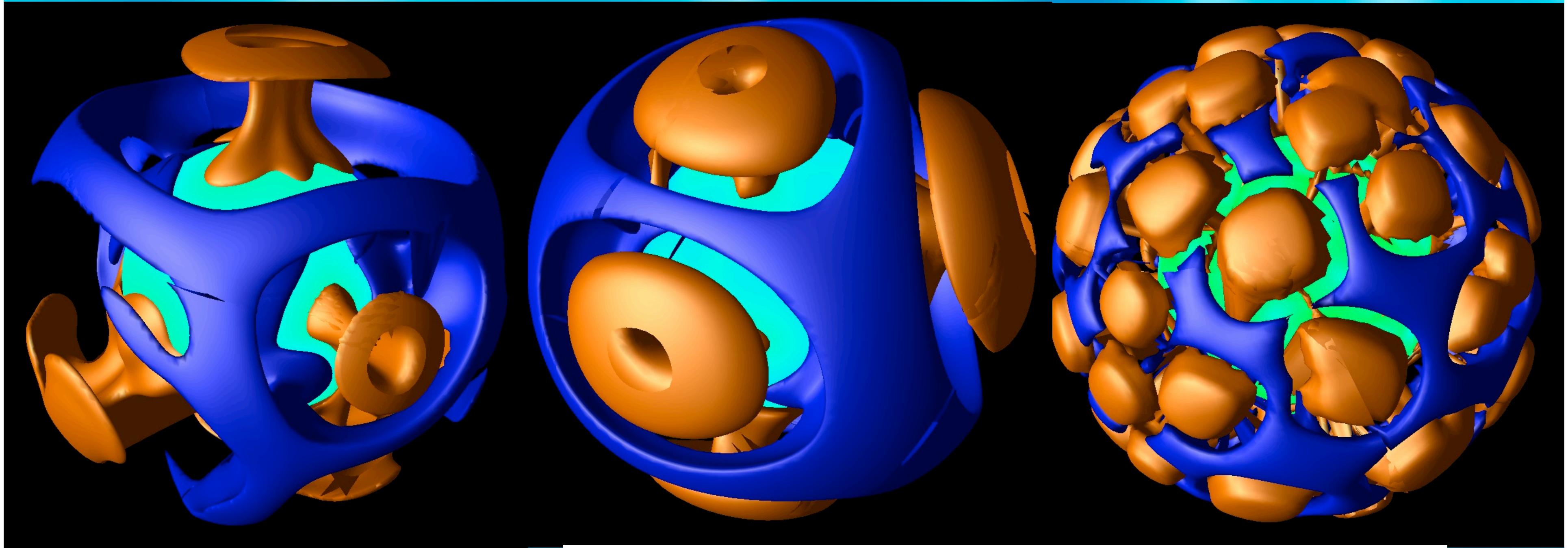


The usual benchmark tests

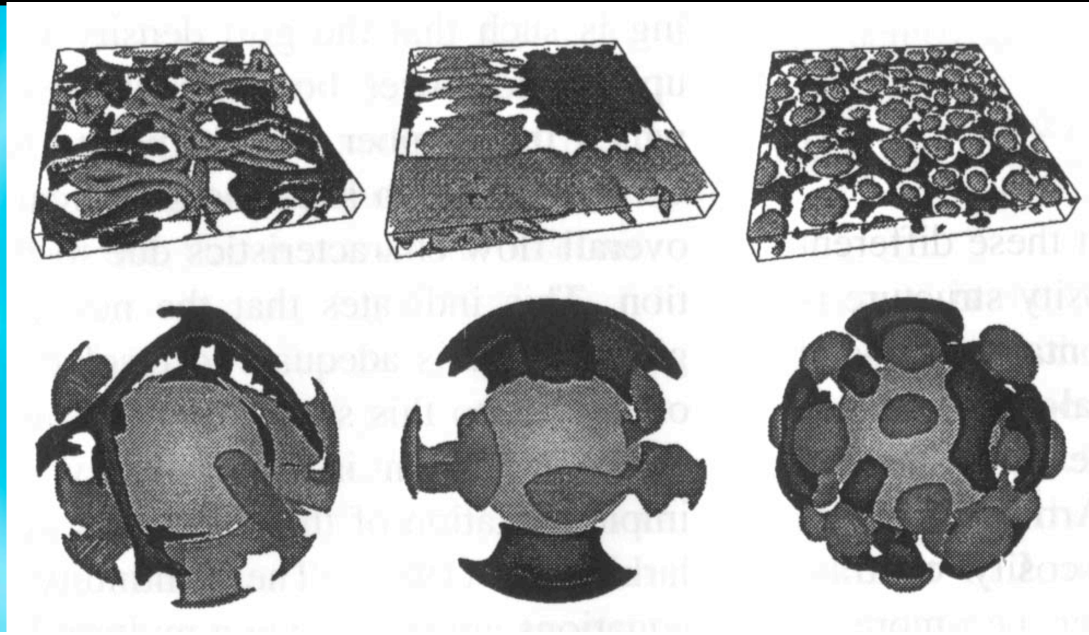


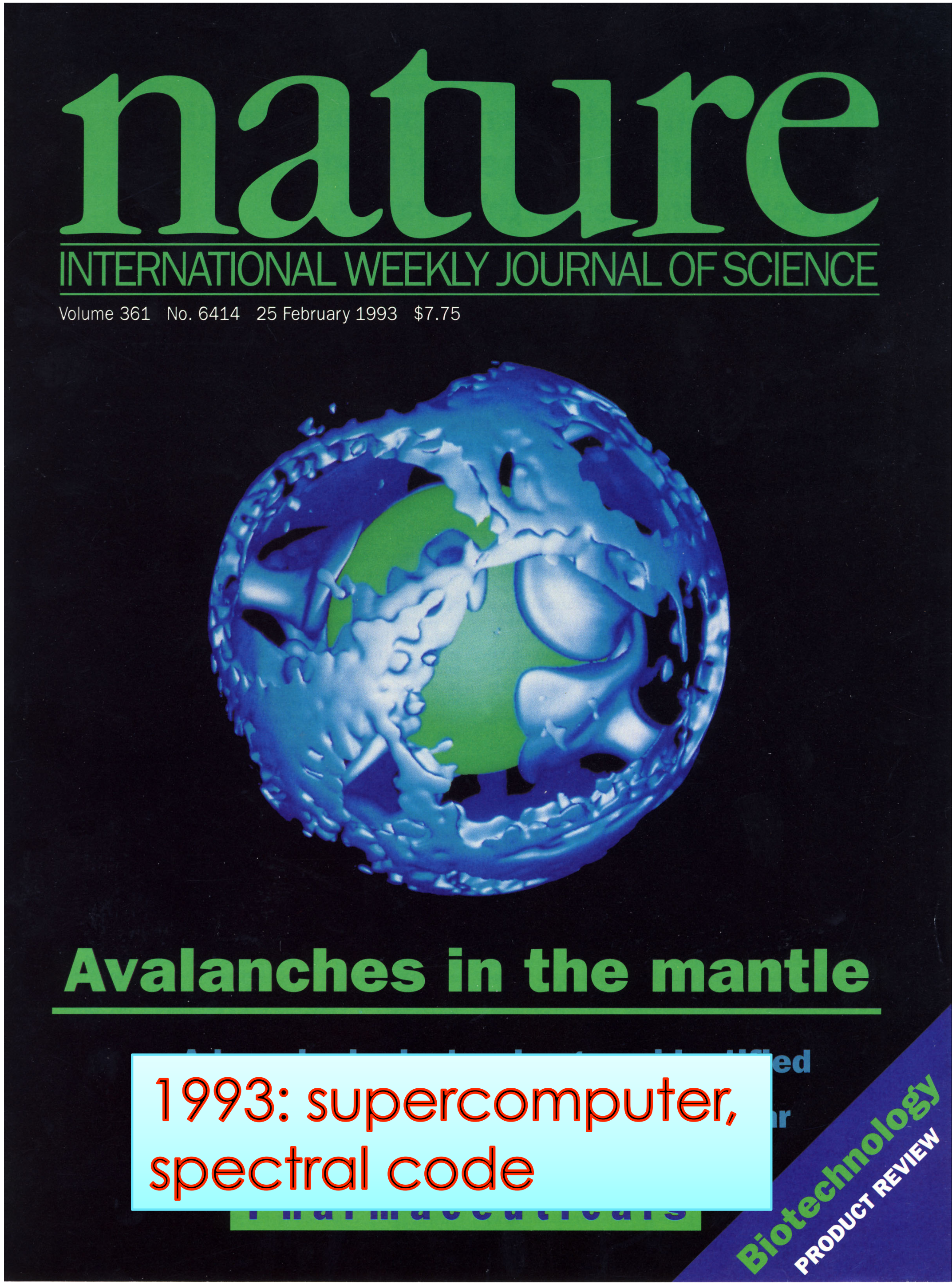


Transitions mobile->sluggish->stagnant lid



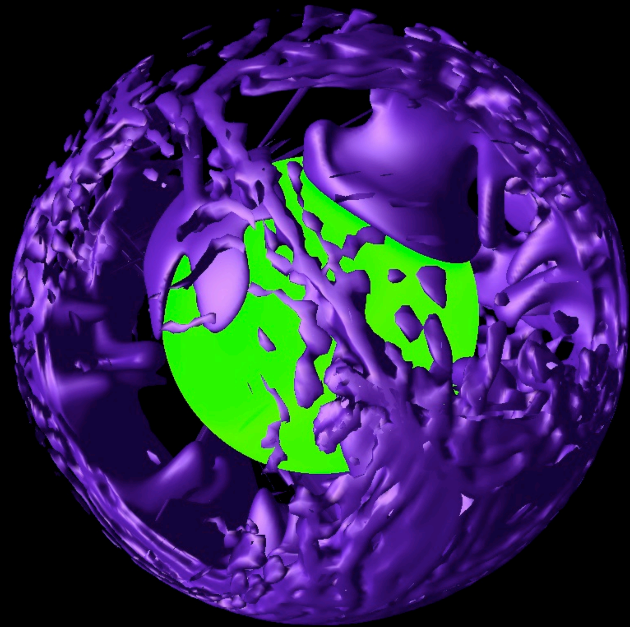
like Ratcliff et al 1996





1993: supercomputer, spectral code

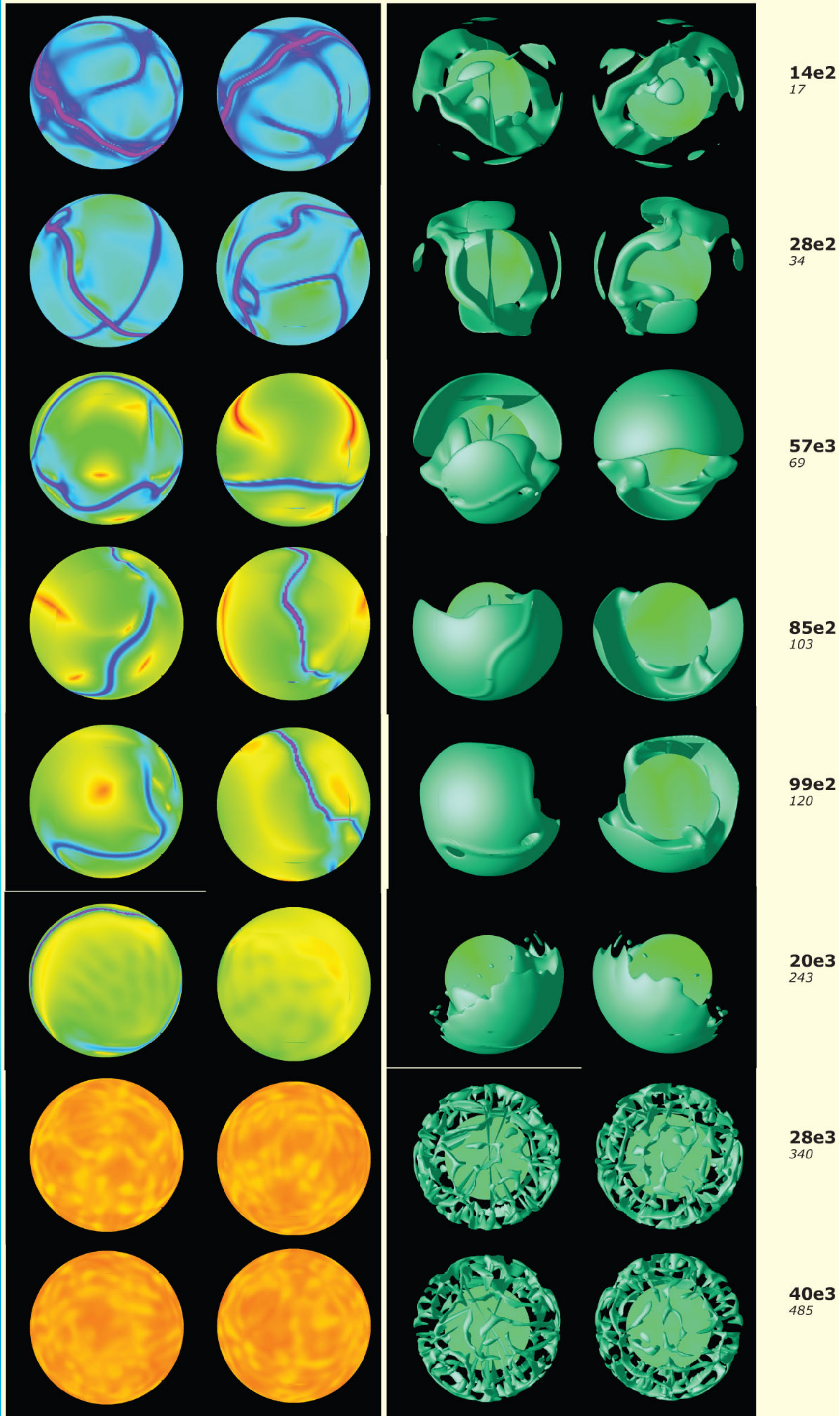
15 years of progress



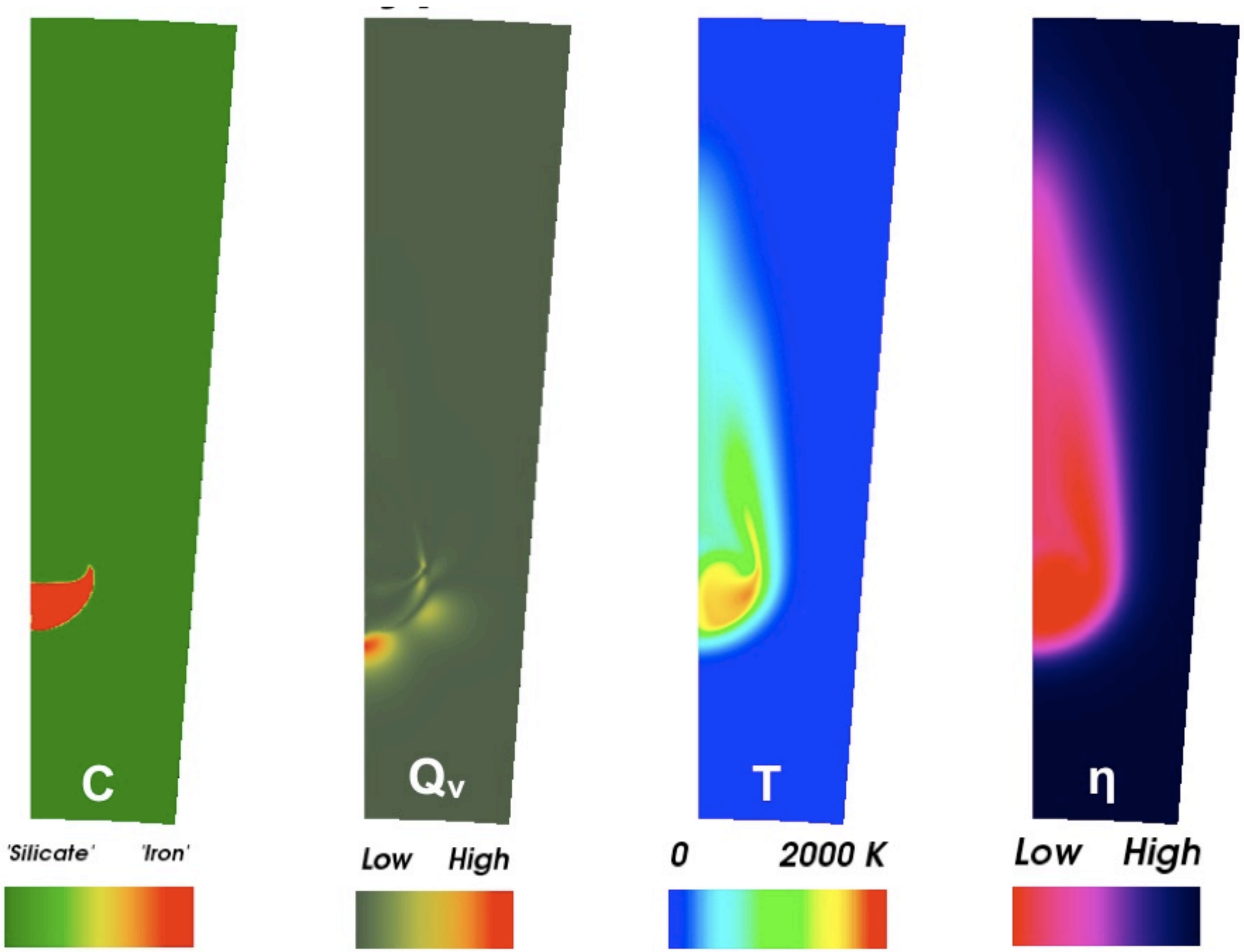
2008: laptop, multigrid code

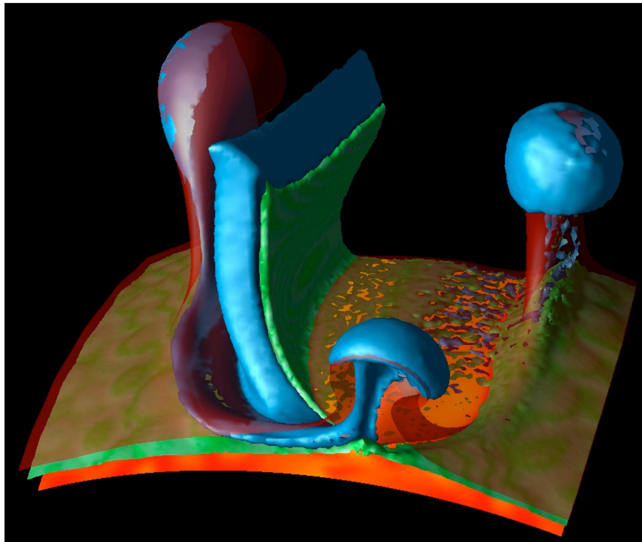
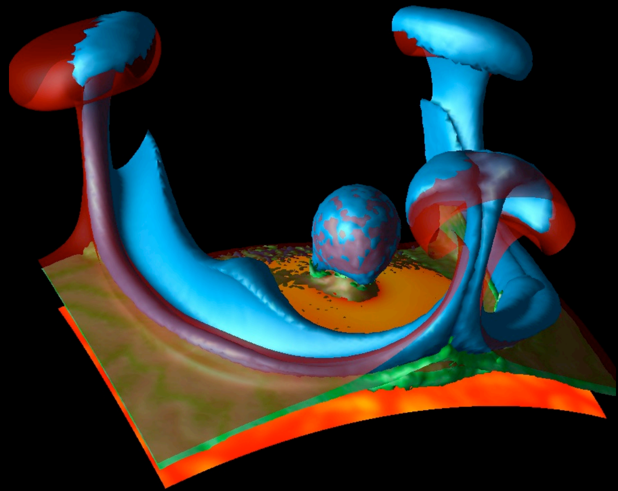
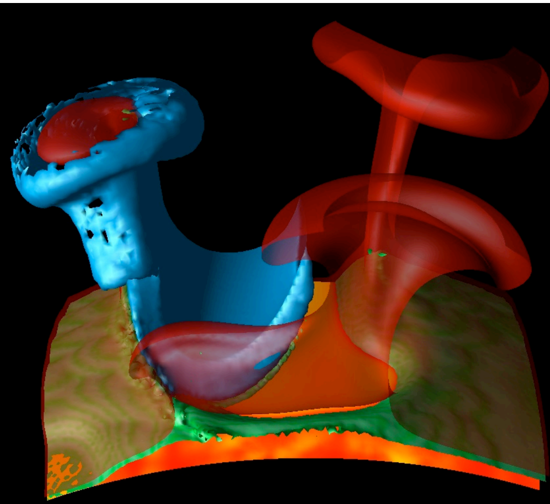
Generation of plate tectonics

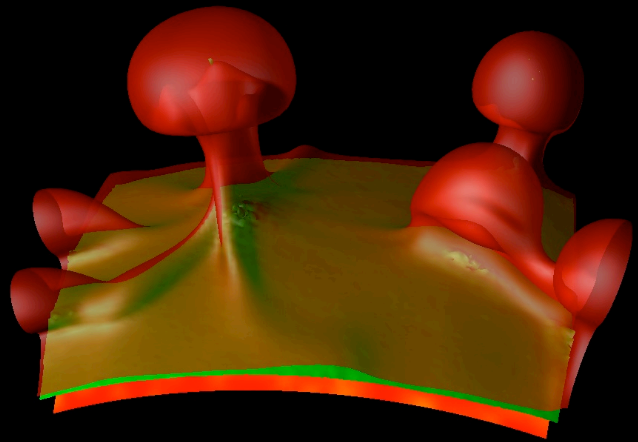
Hein van Heck & me,
GRL 2008



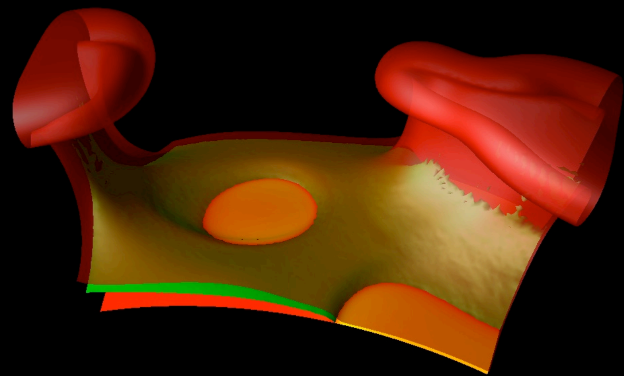
Henri Samuel: Core formation (G3, 2008)

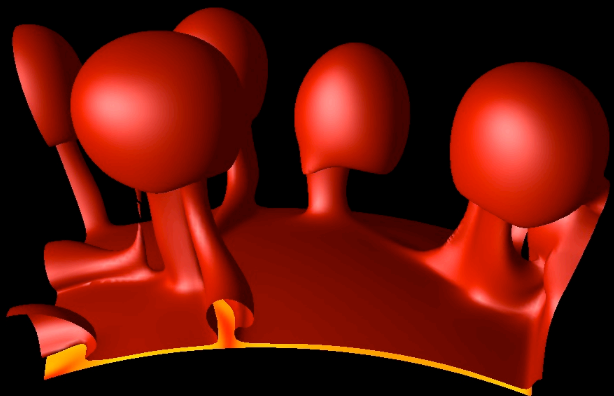
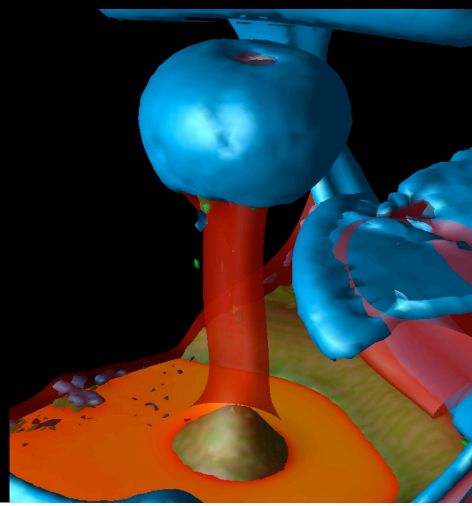


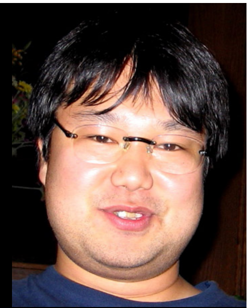


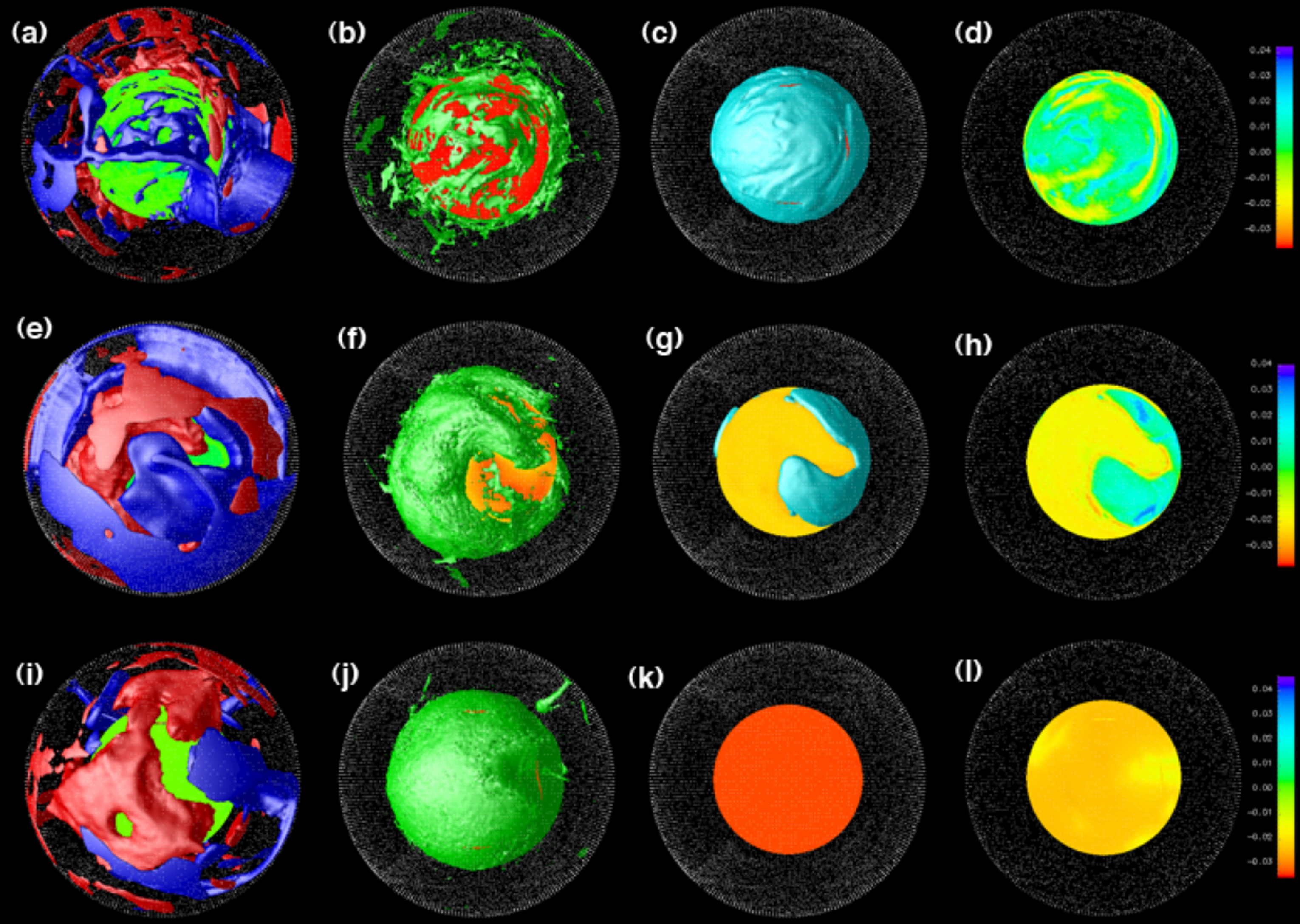
Slab-CMB
interaction
(me)



Earth evolution (Nakagawa & me)





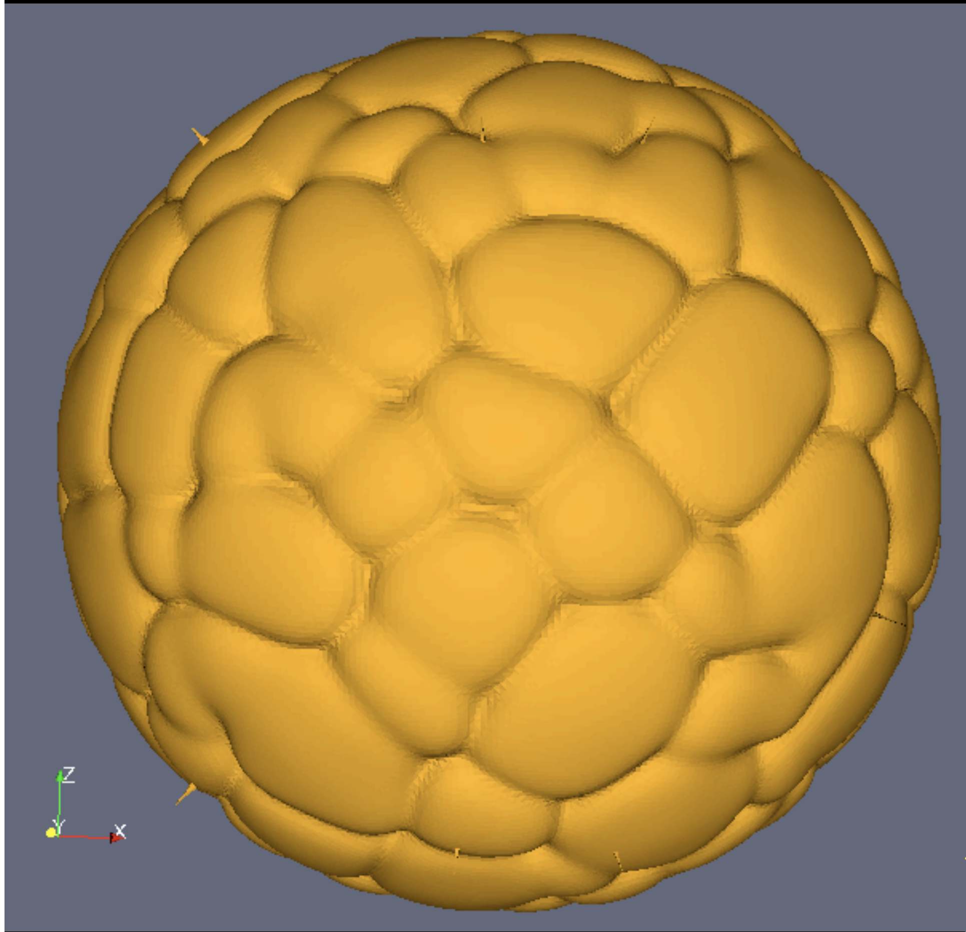
0.0%

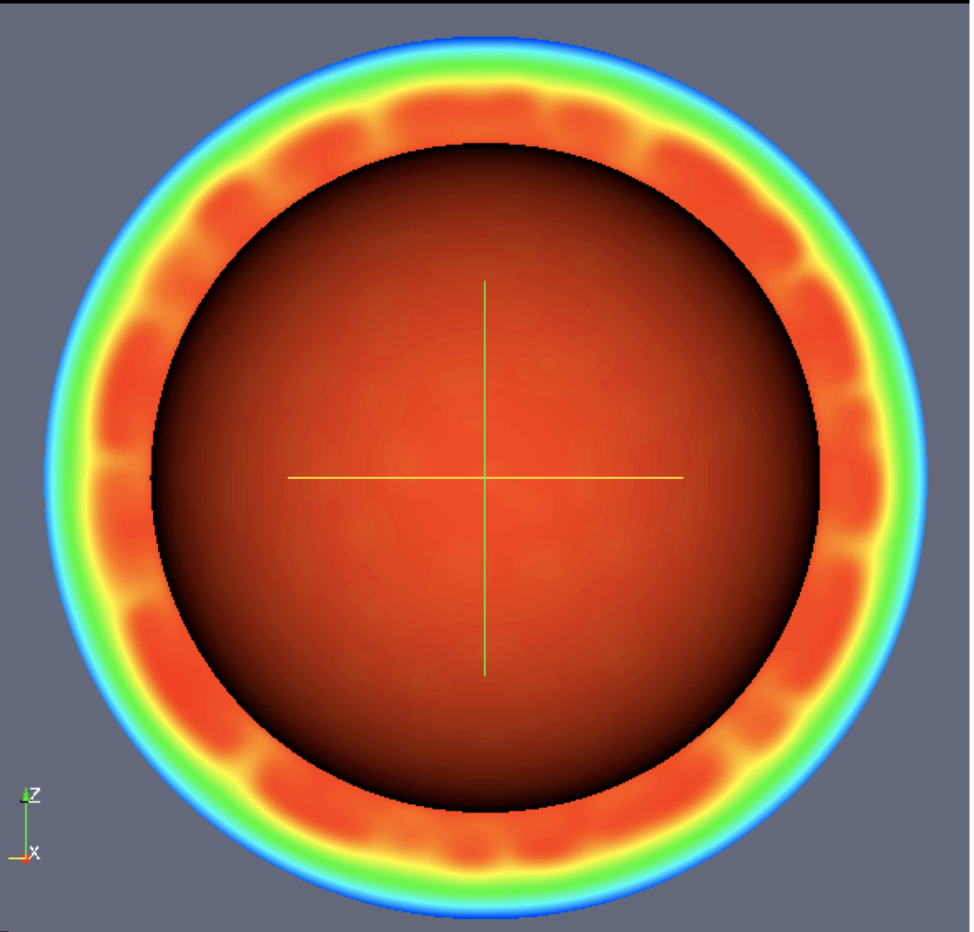
1.8%

3.6%

Temp. Comp. PPV S-anomalies

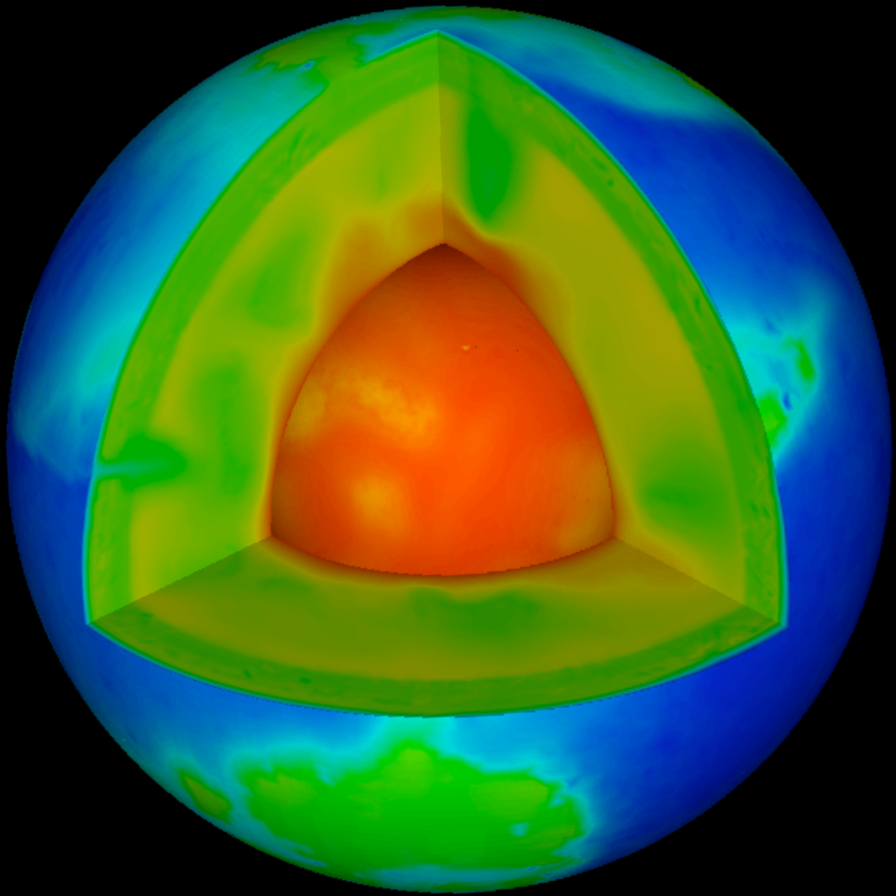
Mercury



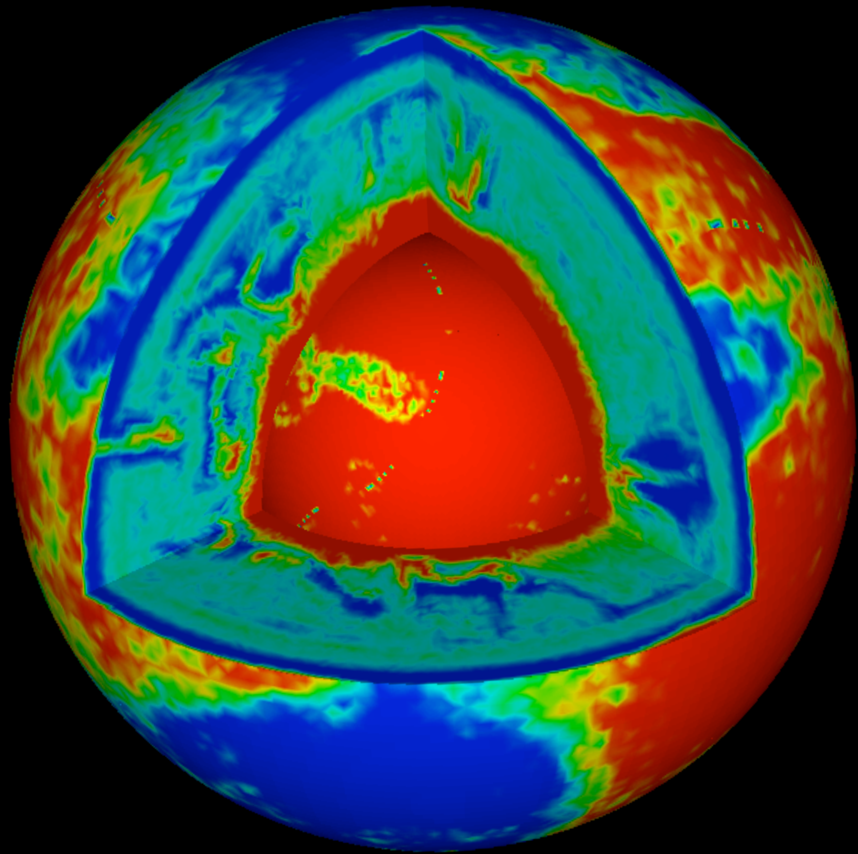


Venus

M. Armann & me



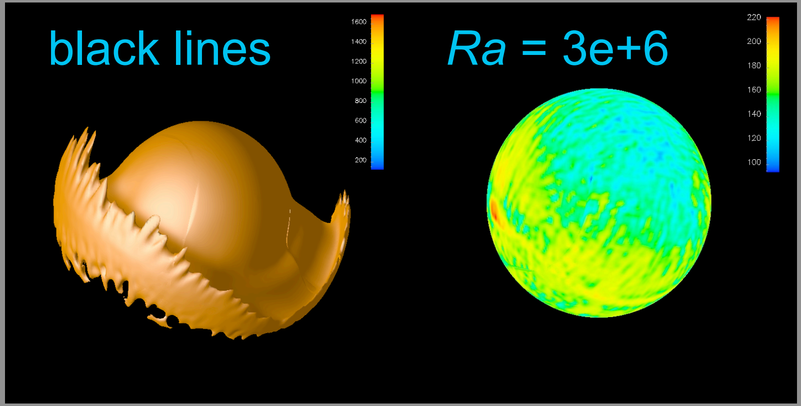
Temperature

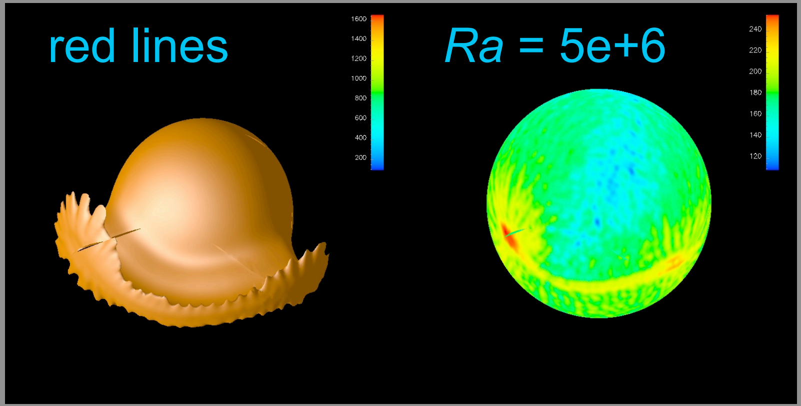


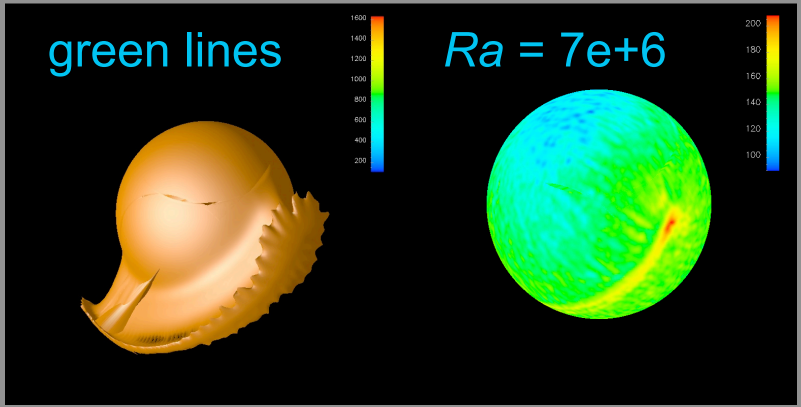
Composition

- episodic “subduction”
- Thin crust
- Layer above CMB

Temperature *Crust. thickness*



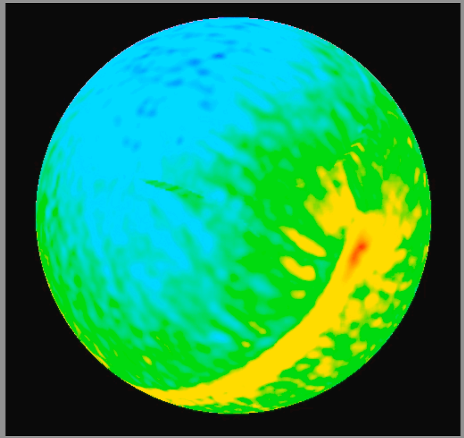


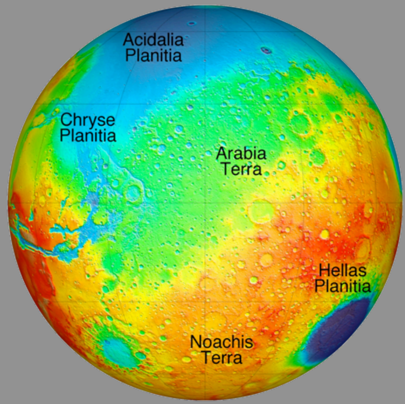


blue lines: *Ra* = 3e+6, no melting

Mars
- after 1 billion years

Tobias Keller & me





Summary of StagYY

- Many geometries including spherical shell using the yin-yang grid
- Efficient & scalable multigrid solver, tracers for composition
- Large viscosity contrasts due to MDPI
- Compressible truncated anelastic
- Self-consistent mineralogy
- Melting, melt migration, crustal formation
- Self-gravitational geoid
- Parameterized core cooling
- Self-contained – no libraries except MPI

Future extensions

- Local grid refinement (adaptive?)
- Visco-elasticity


THE END!
X
Z