Physics of the Earth and Planetary Interiors 171 (2008) 7–18

Contents lists available at ScienceDirect

Physics of the Earth and Planetary Interiors

journal homepage: www.elsevier.com/locate/pepi

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

Paul J. Tackley*

Institute of Geophysics, ETH Hoenggerberg HPP L13, Department of Earth Sciences, ETH Zurich 8093, Switzerland

ARTICLE INFO

Article history:
Received 30 October 2007
Received in revised form 15 July 2008
Accepted 4 August 2008

Keywords:
Mantle convection
Yin-yang grid
Multigrid
Spherical shell

ABSTRACT

Here it is documented how an existing code for modelling mantle convection in a cartesian domain, Stag3D, has been converted to model a 3D spherical shell by using the recently introduced yin-yang grid. StagYY is thus the latest evolution of a code that has been in continuous use and development for about 15 years so incorporates much physics and several features including compressibility, phase transitions, compositional variations, non-linear rheology, parallelisation, tracers to track composition, partial melting and melt migration, and the ability to also model spherical patches, cartesian boxes, and various 2D geometries by changing one input switch. StagYY uses a multigrid solver to obtain a velocity–pressure solution at each timestep on a staggered grid, a finite-volume scheme for advection of temperature and tracers to track composition. Convergence of multigrid solvers in the presence of realistically large viscosity variations has always been a problem; here a new pressure interpolation scheme is presented that can dramatically improve the robustness of the iterations to large viscosity variations, with up to 19 orders of magnitude variation in presented tests. Benchmark tests show that StagYY produces results that are consistent with those produced by other codes. Performance tests show reasonable scaling on a parallel Beowulf cluster up to 64 CPUs, with up to 1.2 billion unknowns solved for in a few minutes. StagYY is designed to be a stand-alone application with no libraries required and if MPI is installed it can be run in parallel. Technical issues and goals for the future are discussed.

1. Introduction

Mantle convection calculations in 3D spherical shell geometry have often used a spectral method (e.g., Bercovici et al., 1989a; Glatzmaier, 1988; Harder and Christensen, 1996; Machetel et al., 1995; Monnereau and Quere, 2001; Tackley et al., 1994; Young, 1974; Zhang and Christensen, 1993; Zhang and Yuen, 1996). Due, however, to limitations in the lateral viscosity contrasts that can be handled with a spectral method (Balachandar et al., 1996; Christensen and Harder, 1991; Zhang and Christensen, 1993), recent years have seen a surge of interest in developing grid-based methods using finite-element, finite-difference or finite-volume approximations. While a simple (longitude, latitude) grid has been used (Ratcliff et al., 1995; Yoshida et al., 1999), it has the disadvantage that the grid lines converge at the poles, causing uneven resolution, limiting the timestep and possibly causing convergence problems for iterative solvers. Various grids have been proposed to overcome this "pole problem" and give a more uniform grid spacing or element size. The first such grid to be used for mantle convection was the isocahedral grid (Baumgardner, 1985, 1988), in which the azimuthal discretization uses triangles suitable for a finite-element solver. A different finite-element discretization using several non-orthogonal rhombahedral patches was implemented by Zhong et al. (2000). The "cubed sphere" grid (Ronchi et al., 1996), on the other hand, is suitable for either finite-difference or finite-element discretizations, and consists of dividing the sphere into six patches by projecting a cube onto a sphere. In its simplest form as implemented for mantle convection by Choblet (2005), Hernlund and Tackley (2003) the grid lines are non-orthogonal, requiring many additional terms in the resulting finite-difference equations. A modified, nearly orthogonal version has however been found by Harder and Hansen (2005), Stemmer et al. (2006).

The so-called "yin-yang" grid proposed by Kageyama and Sato (2004) and first applied to the mantle convection problem by Yoshida and Kageyama (2004, 2006) has the advantage that it is

* Fax: +41 44 633 1065.
E-mail address: ptackley@ethz.ch.

0031-9201/$ – see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.pepi.2008.08.005

naturally orthogonal. This grid consists of two (longitude, latitude) grids each stretching ±45° in latitude by 270° in longitude that are combined like the two patches of a tennis ball or baseball to make a sphere. The main advantage of this grid over alternatives is that the grid lines remain orthogonal and therefore a simple discretization such as finite-differences can be used, simplifying the implementation and, for example, allowing one to take advantage of various advection techniques that have been developed in the atmospheric community on orthogonal grids. The grid spacing various by cosine(45°), i.e., $\sqrt{2}$. A disadvantage is that the two subgrids do not mesh neatly, unlike the cubed sphere or the isocahedral grid, requiring a more complex interpolation at the edge. Additionally, the two subgrids overlap—by 6% if the simple rectangular subgrids are used (Kageyama and Sato, 2004), which gives the possibility of developing somewhat different solutions in the overlapping areas. This can, however, be minimized by using a minimum overlap version of the grid, two different version of which are shown in Figs. 5 and 6 of Kageyama and Sato (2004).

In this paper, an implementation of the yin-yang grid for mantle convection is described that advances on that of Yoshida and Kageyama (2004) in several ways. One way is to use a "minimum overlap" version of the yin-yang grid, namely the version shown in Fig. 5 of Kageyama and Sato (2004), in which any cells at the corners of each subgrid that are completely contained within the other subgrid are removed to produce what they refer to as a "baseball-like" border curve. Other improvements are the implementation of more complex physics, including the compressible anelastic equations, phase transitions, compositional variations, and non-linear rheology, and technical advances including parallelisation, tracers to track composition, and a pressure interpolation scheme that gives enhanced robustness to large viscosity variations. The code is a development of Stag3D, initially developed in 1992 (Tackley, 1993) and steadily developed since then (e.g., Tackley, 1996, 1998b, 2002; Tackley and Xie, 2003), and thus inherits most of the features of that code. The new version of the code reported here retains the ability to model a 3D or 2D cartesian domain, and in addition to a full "yin-yang" spherical shell can also model a regional spherical block, or the two-dimensional spherical geometries of axisymmetric or spherical annulus (Hernlund and Tackley, 2008). Results in these alternative geometries are illustrated in Fig. 1a–c, and have the same parameters as the case in Fig. 1g. The code is written in Fortran 95 and takes advantages of features such as dynamic array allocation and derived variable types.

2. Physical model

2.1. Approximations and equations

As usual for the solid Earth, the infinite Prandtl number approximation is made. Additionally, compressibility is included by assuming the anelastic approximation, in which the continuity equation uses a reference state density rather than the full pressure-dependent density, in order to avoid acoustic waves (e.g., Schubert et al., 2000). In the present version, the "truncated" version of the anelastic approximation is used, which means that the effect of dynamic pressure on temperature is neglected. A quick calculation suggests that this term is unimportant but this should be rigorously tested in the future. Implementing the full anelastic equations would not add significant complexity and the author has worked with them in the past (Tackley et al., 1993, 1994), but in this application using the truncated version allows dynamic pressure to be used as a variable rather than total pressure, allowing the use of 32-bit rather than 64-bit precision for the pressure field. That is, as dynamic pressure is small compared to total pressure in the deep mantle, treating pressure variations accurately enough to enforce the continuity equation would require 64-bit precision if using total pressure.

These assumptions lead to the following set of equations, non-dimensionalised to thermal diffusion scales. Conservation of mass:

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

momentum

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

and energy

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

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

$$\frac{DC}{Dt} = 0 \tag{4}$$

The (non-dimensional) variables are total temperature T, composition C, velocity $\underline{v}$ and pressure p. $\underline{\underline{\sigma}}$ is the deviatoric stress tensor and $\underline{\underline{\dot{\varepsilon}}}$ is the strain rate tensor. The governing parameters are Rayleigh number Ra, internal heating rate H, and surface dissipation number $\mathrm{Di_s}$. Material properties are density ρ, thermal expansivity α, thermal conductivity k, and specific heat capacity C_p. $\Delta\rho_{\text{thermal}}$ is the fractional density variation with temperature ($=\alpha_{\text{dimensional}}\,\Delta T_{\text{dimensional}}$) and r is the radius. Viscosity η can vary with temperature, depth, strain rate or stress, composition, phase, melt fraction, etc.

2.2. Variation of physical properties

The above equations are written and implemented in a general way such that different dependencies of physical properties on temperature, pressure and composition can be used. In the classical anelastic approximation, density, expansivity, diffusivity and heat capacity are functions of depth (radius) only, whereas in the Boussinesq approximation they are 1, except in the buoyancy term in the momentum equation. To progress to more realistic models, in StagYY these can alternatively be arbitrary functions of temperature, depth, and composition. For example phase changes can cause density jumps of up to 10% so in the continuity equation it can be chosen to use the total density instead of a reference density—as long as the density being used is not dependent on dynamic pressure there is no problem with sound waves.

Phase changes are implicitly included in the density (thereby contributing to buoyancy) and their latent heat is included in the energy equation by using "effective" values of heat capacity and thermal expansivity (Christensen and Yuen, 1985). Density, expansivity, conductivity and viscosity are implemented as functions, so that different forms can be easily substituted without changing the main code.

The "standard" method of calculating the temperature and pressure dependence of these in the context of a multi-phase, two-component system has been described in a number of papers over the years (e.g., Nakagawa and Tackley, 2005; Tackley, 1996; Xie and Tackley, 2004), so the reader is referred to these for details. Recently, physical properties calculated from self-consistent minimization of free energy to determine stable minerals, coupled to a thermodynamic database, have been implemented using the PERPLEX package (Connolly, 2005), along the lines of Gerya et al. (2006).

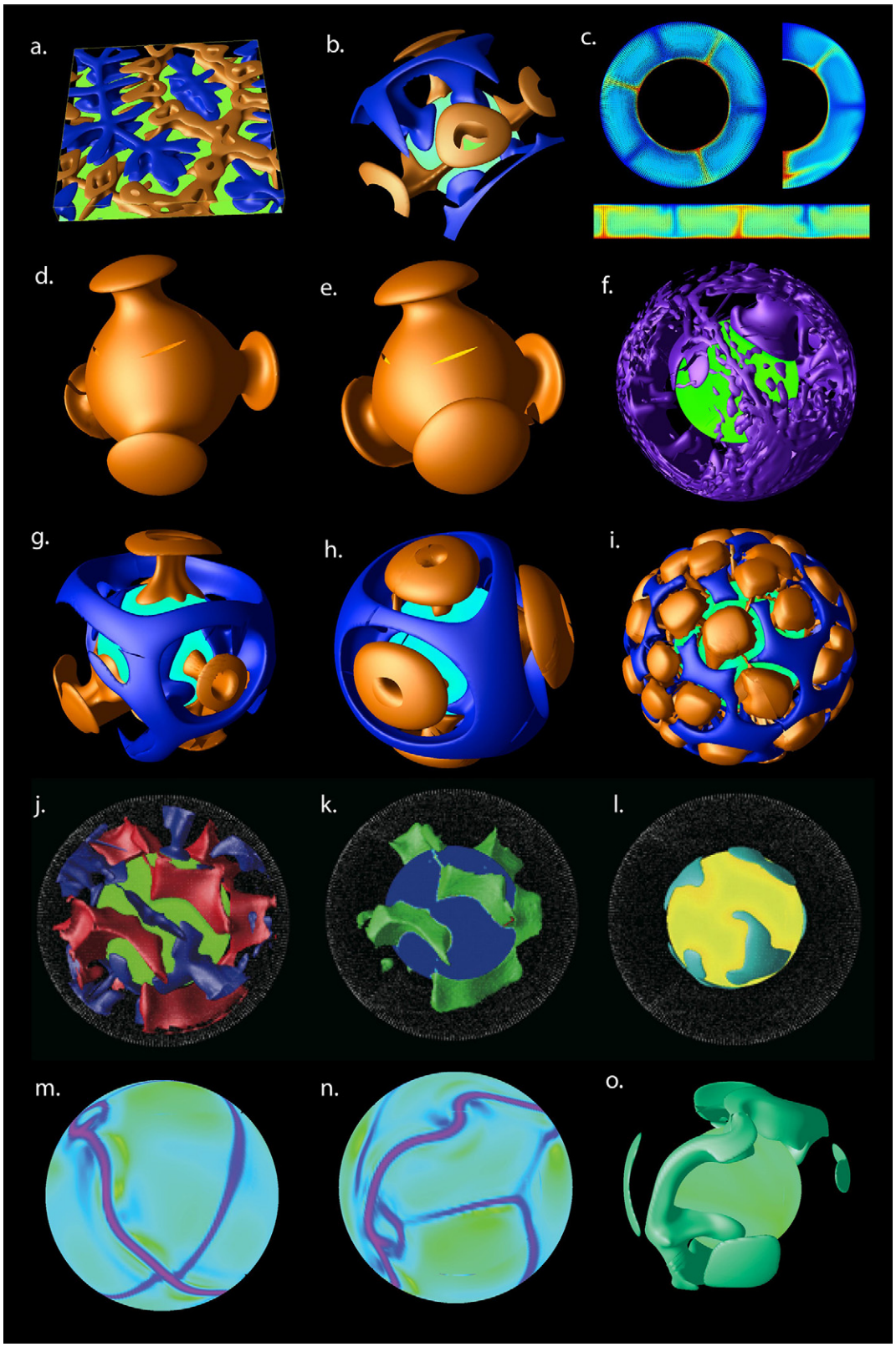

Fig. 1. Various results obtained with StagYY. The top row illustrates alternative geometries that can be modelled by changing one input switch, all for basal heated convection at $Ra = 10^5$: (a) cartesian, (b) spherical patch, and (c) 2D spherical annulus, spherical axisymmetric, or cartesian. (d) Isoviscous or (e) viscosity contrast 20 tetrahedral benchmark cases with $Ra = 7000$; isosurface of $T = 0.4$ is shown. (f) Compressible convection with an endothermic phase change at 670 km depth and parameters as in Tackley et al. (1993). (g–i) Basally heated convection at $Ra = 10^5$: isoviscous, viscosity contrast 10^3 or 10^6, respectively. (j) Residual temperature isosurfaces (k) composition isosurfaces and (l) post-perovskite for compressible thermo-chemical multi-phase convection discussed in (Nakagawa and Tackley, 2008), (m and n) viscosity in the outer layer and (o) temperature isosurface for internally heated convection with visco-plastic temperature-dependent viscosity, showing self-consistent generation of tectonic plates with parameters similar to Tackley (2000a,b) and van Heck and Tackley (in press).

Details of this treatment will be elaborated in a future publication.

2.3. Geometry and boundaries

While the focus here is on treating a full 3D spherical shell using the yin-yang grid, a single spherical region or arbitrary size can also be modelled, and 3D cartesian geometry is maintained as an option. Various related 2D geometries can also be modelled, including cartesian 2D, spherical axisymmetric, and spherical annulus (see Hernlund and Tackley (2008) for details on this). Top and bottom boundary condition are typically isothermal or insulating, and free slip or rigid. Side boundaries can be periodic, reflecting, or permeable, or are interpolated from the other block in the case of the yin-yang grid.

3. Numerical implementation

3.1. Grid

As is usual (e.g., Brandt (1982), Patankar (1980) and used for mantle convection since Ogawa et al. (1991)), velocity components and pressure are defined on a staggered grid on which the pressure is defined in the center of each cell and velocity components are defined at cell boundaries perpendicular to their direction. This has the advantages that all derivatives in the momentum and continuity equations involve adjacent points and are second-order accurate in the case of even grid spacing. Additionally, checkerboard pressure fields are avoided.

As mentioned earlier, StagYY uses the minimum overlap yin-yang grid shown in Fig. 5 of Kageyama and Sato (2004), in which any cells at the corners of each subgrid that are completely contained within the other subgrid are removed to produce what they refer to as a "baseball-like" border curve. Any loops that loop over all grid cells must take into account the removed cells in the corners. Each subgrid acts as the boundary condition to the other subgrid, with "ghost points" holding velocity and pressure values that are linearly interpolated from the interiors of the other subgrid. Even for a regular (fully overlapping) yin-yang grid, the locations of these ghost points are different at different grid levels due to the different grid spacing: for coarser grids they are further inside the other subgrid. For the minimum overlap grid it is more complicated to determine the locations of the ghost points, and the fact that they are different at each grid level does not change. For this minimum overlap grid, the boundary is also different in detail at different grid levels, but this does not matter because the solution is still defined everywhere—there are no "gaps". For computational efficiency, the interpolation weights and the points to be used are pre-calculated once and stored.

3.2. Finite-difference discretization

The equations are expressed in spherical polar coordinates. The form of the stress and strain rate tensors and their divergences in spherical polar coordinates is well known (e.g., Schubert et al., 2000) so the full set is not repeated here. It is, however, noted that they can be written in different ways, which although mathematically identical lead to a different finite-difference expansion hence a slightly different numerical solution. The form of stress divergences used here is slightly rearranged from that given on page 281 of Schubert et al. (2000) in order to best reflect the physical meaning of the different terms, as shown in the following equations.

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

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

$$(\nabla \bullet \underline{\underline{\sigma}})_\phi = -\frac{1}{r\sin\theta}\frac{\partial p}{\partial\phi} + \frac{1}{r^2}\frac{\partial}{\partial r}(r^2\tau_{r\phi}) + \frac{1}{r\,\sin\theta}\frac{\partial}{\partial\theta}(\tau_{\theta\phi}\sin\theta) + \frac{1}{r\sin\theta}\frac{\partial\tau_{\phi\phi}}{\partial\phi} + \frac{1}{r}(\tau_{r\phi} + \tau_{\theta\phi}\,\cot\,\theta) \quad (7)$$

These stress divergences are expanded in terms of velocities then a straightforward finite-difference expansion is applied. The finite-difference stencil for the equation at each of the staggered velocity or pressure points is pre-calculated and the stencil weights are stored. Thus, calculating the residue at that point simply requires multiplying nearby velocity and pressure values by the appropriate weights and summing. This has the advantage that once the stencil weights are calculated, the solution routines are the same for spherical or cartesian geometry.

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 $-1/3\nabla\cdot\underline{v}$. 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:

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

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

Viscosity is initially calculated at cell centers where temperature is defined. Calculation of normal stresses requires it at that location, but viscosity must be interpolated to the centers of cell edges for the shear stress terms. The appropriate type of interpolation (e.g., arithmetic, harmonic, geometric or more complex (Ogawa et al., 1991)) depends on the conceptualization of how the viscosity varies physically (e.g., stepwise, linear, geometric). Recently, Deubelbeiss and Kaus (2008) compared numerical results with analytic solutions to determine the accuracy obtained with different interpolation methods, and found that harmonic interpolation gave the most accurate results, followed by geometric, with linear (arithmetic) being much worse. Similar findings were also becoming apparent from the subduction benchmark (Schmeling et al., 2008). Thus, linear (arithmetic) interpolation should certainly be avoided. In the present code, geometric interpolation is generally used because it gives much better convergence than with linear interpolation, and slightly better than with harmonic interpolation.

3.3. Velocity–pressure iterations

The velocity–pressure solution that satisfies the continuity and momentum equations is obtained by iterations, which are

encapsulated into a multigrid cycle as described later. The energy equation is treated explicitly as also discussed later. Here the basic velocity–pressure iteration scheme is described, which has some similarities and differences to the well-known SIMPLER scheme (Patankar, 1980).

1. Improve velocity field according to the momentum equations
   a. Radial velocity field according to the radial momentum equations.
   b. Theta velocity field according to the theta momentum equations.
   c. Phi velocity field according to the phi momentum equations.
2. Update pressure field to reduce divergence in the continuity equation.
   a. Calculate pressure correction.
   b. Calculate velocity correction caused by pressure correction.

In each velocity update (e.g., step 1a), the residue is calculated over the entire grid, then the correction is calculated and finally added to the solution, i.e., Jacobi iterations. This has the advantage that the procedure can be parallelised without affecting the result, whereas if Gauss–Seidel iterations are used then a different result will be obtained when the problem is split between different numbers of CPUs. The disadvantage is that convergence is slower. However, this can be overcome by using "red–black" iterations, in which two subsweeps are performed on alternating grid points, like the colouring of squares on a checkerboard. Red–black iterations give better convergence than point-by-point iterations of either Gauss–Seidel or Jacobi varieties (Press et al., 1992).

The correction to each velocity component depends on the stencil weight and a relaxation parameter α_m, for example for the theta-velocity:

$$\delta v^\theta_{i-.5jk} = -\alpha_\mathrm{m} R^{\theta\ \mathrm{mom}}_{i-.5jk} \Big/ \left(\frac{\partial R^{\theta\,\mathrm{mom}}_{i-.5jk}}{\partial v^\theta_{i-.5jk}}\right) \tag{9}$$

where $v^\theta_{i-.5jk}$ is the theta-velocity at point $(i-0.5,j,k)$ and $R^{\theta\,\mathrm{mom}}_{i-.5jk}$ is the residue (error) of the theta-momentum equation at the same point. The 0.5 in the indices arises because (i,j,k) is the center of a cell, at which pressure is defined; the velocity components are defined at staggered points half a grid spacing from this cell center. For multigrid purposes, α should be less than unity in order to obtain optimal smoothing (Brandt, 1982; Wesseling, 1992), so is typically set to 0.7. In the absence of multigrid, over-relaxation is optimal, i.e., $\alpha > 1$.

The pressure correction (step 2a) is calculated using a coefficient that describes how much changing the pressure at a point changes the continuity residue (i.e., divergence of density times velocity) at that point:

$$\delta P_{ijk} = -\alpha_c R^{\mathrm{cont}}_{ijk} \Big/ \left(\frac{\partial R^{\mathrm{cont}}_{ijk}}{\partial P_{ijk}}\right) \tag{10}$$

where the symbols have similar meanings to those in Eq. (9) and α_c is a relaxation parameter generally taken to be 1.0. $(\partial R^{\mathrm{cont}}/\partial P)$ is not a stencil weight because the continuity equation does not include pressure, but it is related to the stencil weights of the momentum and continuity equations. An important question is how to calculate this, because in principle changing the pressure at one point affects velocities and pressures in the entire domain, requiring a global solution. It has been found, however, that the lowest order approximation is sufficient. This means that the effect of pressure on the six neighbouring velocity points is taken into account, but its effect on more distant velocity points is not considered, and neither is the effect of a change in the velocity at one point on the velocities at other points. Specifically, stencil weights of the momentum equations at the six surrounding velocity points give the amount by which velocities at those points are changed when P_{ijk} is changed, then combining these with the stencil weights for the continuity equation leads to the desired approximation to the derivative.

$$\begin{aligned}\left(\frac{\partial R^{\mathrm{cont}}_{ijk}}{\partial P_{ijk}}\right) \approx{}& \left(\frac{\partial R^{\mathrm{cont}}_{ijk}}{\partial v^\theta_{i+.5jk}}\right)\left(\frac{\partial R^{\theta\ \mathrm{mom}}_{i+.5jk}}{\partial P_{ijk}}\right) \Big/ \left(\frac{\partial R^{\theta\,\mathrm{mom}}_{i+.5jk}}{\partial v^\theta_{i+.5jk}}\right)\\ &+\left(\frac{\partial R^{\mathrm{cont}}_{ijk}}{\partial v^\theta_{i-.5jk}}\right)\left(\frac{\partial R^{\theta\ \mathrm{mom}}_{i-.5jk}}{\partial P_{ijk}}\right) \Big/ \left(\frac{\partial R^{\theta\,\mathrm{mom}}_{i-.5jk}}{\partial v^\theta_{i-.5jk}}\right)\\ &+\left(\frac{\partial R^{\mathrm{cont}}_{ijk}}{\partial v^\phi_{ij+.5k}}\right)\left(\frac{\partial R^{\phi\ \mathrm{mom}}_{ij+.5k}}{\partial P_{ijk}}\right) \Big/ \left(\frac{\partial R^{\phi\,\mathrm{mom}}_{ij+.5k}}{\partial v^\phi_{ij+.5k}}\right)\\ &+\left(\frac{\partial R^{\mathrm{cont}}_{ijk}}{\partial v^\phi_{ij-.5k}}\right)\left(\frac{\partial R^{\phi\ \mathrm{mom}}_{ij-.5k}}{\partial P_{ijk}}\right) \Big/ \left(\frac{\partial R^{\phi\,\mathrm{mom}}_{ij-.5k}}{\partial v^\phi_{ij-.5k}}\right)\\ &+\left(\frac{\partial R^{\mathrm{cont}}_{ijk}}{\partial v^r_{ijk+.5}}\right)\left(\frac{\partial R^{r\ \mathrm{mom}}_{ijk+.5}}{\partial P_{ijk}}\right) \Big/ \left(\frac{\partial R^{r\,\mathrm{mom}}_{ijk+.5}}{\partial v^r_{ijk+.5}}\right)\\ &+\left(\frac{\partial R^{\mathrm{cont}}_{ijk}}{\partial v^r_{ijk-.5}}\right)\left(\frac{\partial R^{r\ \mathrm{mom}}_{ijk-.5}}{\partial P_{ijk}}\right) \Big/ \left(\frac{\partial R^{r\,\mathrm{mom}}_{ijk-.5}}{\partial v^r_{ijk-.5}}\right)\end{aligned} \tag{11}$$

where the derivatives in parentheses are the stencil weights of the finite-difference approximations of the equations.

A quick examination of $(\delta R^{\mathrm{cont}}/\delta R)$ reveals that it scales as 1/viscosity, as follows. If h represents grid spacing, then $(\delta R^{\mathrm{cont}}/\delta v) \approx 1/h$, $(\delta R^{\mathrm{mom}}/\delta v) \approx 1/h$, and $(\delta R^{\mathrm{mom}}/\delta v) \approx \eta/h^2$. 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), although the latest version calculates the coefficient according to the above equation. Kameyama et al. (2005) also derived a pressure correction proportional to viscosity using a quite different train of logic based on pseudo-compressibility. Thus it can be seen that the pressure correction derived using pseudo-compressibility is the same as that derived by directly considering satisfying the finite-difference approximation of the equations. Another approach is to derive an "effective" Poisson-like pressure equation, which also ends up deriving a similar pressure correction.

Step 2b, in which velocities are adjusted according to the pressure correction, was found to improve convergence compared to going directly to the next full momentum equation iteration. In this step, each velocity component is adjusted based on the pressure correction at the two adjacent pressure points (half a grid spacing either side) multiplied by the appropriate stencil weight, for example:

$$\delta v^\theta_{i-.5jk} = \frac{\delta P_{i-1jk}(\partial R^{\theta\ \mathrm{mom}}_{i-.5jk}/\partial P_{i-1jk}) + \delta P_{ijk}(\partial R^{\theta\ \mathrm{mom}}_{i-.5jk}/\partial P_{ijk})}{\partial R^{\theta\,\mathrm{mom}}_{i-.5jk}/\partial v^\theta_{i-.5jk}} \tag{12}$$

The SIMPLER scheme (Patankar, 1980) has an additional step that is left out here. This involves calculating the new pressure from a Poisson-like equation derived from the divergence of the momentum equation. This step is not necessary: the role of pressure is to enforce continuity, so using the residue of the continuity equation to determine pressure corrections is sufficient. SIMPLER also iterates on the temperature equation, which would allow it to be treated implicitly as done by (Albers, 2000; Trompert and Hansen, 1996) but implicit advection tends to be diffusive; modern advection schemes designed to minimize numerical diffusion and dispersion are generally explicit.

As an alternative to the sequential scheme described in the above steps, a "cell relaxation" or "pressure-coupled" scheme is

also implemented, in which corrections to the pressure and six surrounding velocities are calculated simultaneously using a matrix solution method, which involves a 7×7 matrix in three dimensions (Tackley, 2000a). Using this scheme, the solution converges in fewer iterations, but it takes several times as much CPU time per iteration. Thus, the overall solution time is longer. Both the point-wise and the cell-wise schemes appear to be similar in their robustness (or non-robustness) to large viscosity variations, therefore the faster point-wise scheme is favoured at present. Auth and Harder (1999) implemented a similar scheme in a 2D code except using a diagonalised version of the matrix instead of the full matrix, based on Vanka (1986).

3.4. Multigrid cycles

The multigrid method, in which the residue (error) to the equations is relaxed on a heirarchy of nested grids with different grid spacing, can dramatically accelerate the convergence rate of iterative solvers because in principle it relaxes all wavelengths of the residue simultaneously, resulting in a solution time that scales in proportion to the number of unknows (Brandt, 1982; Wesseling, 1992). For mantle convection this was applied to a staggered velocity–pressure grid by Tackley (1993). A key problem with applying the multigrid method to mantle convection is a lack of robustness to large viscosity variations, i.e., the iterations converge very slowly or diverge (e.g., that initial study only reached viscosity contrasts of 10^3). Broadly speaking this is because the coarse grids do not "see" correctly the fine-grid problem, so corrections calculated at coarse levels may actually degrade the solution at finer levels rather than improving it. Thus, over the years, several researchers have proposed improvements to staggered grid multigrid algorithm to address this problem.

In the general multigrid literature, the accepted approach to deriving coarse-grid operators, particularly in the case of strongly varying coefficients, is to use matrix-dependent prolongation and restriction operators combined with the Galerkin coarse-grid approximation (GCGA) (e.g., Wesseling, 1992). To explain the terminology: the prolongation operator P is used to interpolate coarse-grid corrections to the next finest grid, while the restriction operator R is used to restrict fine-grid residue to the next coarsest level. "Matrix-dependent" means that these operators are derived from the stencil (matrix) of the discretized equations at the finer level, e.g., A. In the Galerkin coarse-grid approximation, the coarse-grid operator is based on the fine grid operator and the prolongation and restriction operators as: $A_c = RA_fP$, the interpretation of which is that is has the same effect as prolongating the fields to the finer level, applying the fine-grid operator then restricting the result to the coarser level.

Matrix-dependent operators and the Galerkin coarse grid were implemented in a 2D finite-element mantle convection code by (Yang and Baumgardner, 2000) with apparently astonishing results, easily handling viscosity contrasts of 10^{10} between adjacent points. Unfortunately, similar robustness was not obtained when the method was implemented in the related 3D spherical shell finite-element code TERRA (J. R. Baumgardner, personal communication), the reason for which is uncertain. The method has not yet been successfully applied to a staggered grid mantle convection code because of the complexity in this case, although they have been applied to the constant viscosity incompressible Navier–Stokes equations (Zeng and Wesseling, 1992a,b).

So far, mantle convection implementations of staggered grid multigrid have instead re-discretized the equations on the coarse grid using a viscosity field that is averaged from the fine grid. Several authors have proposed improvements to the scheme of Tackley (1993). Firstly, Trompert and Hansen (1996) introduced a new averaging scheme for the coarse-grid viscosities, in which anisotropic viscosities are used to calculate the coarse-grid shear stresses. They also found that taking additional iterations on the pressure term helped overall convergence. Auth and Harder (1999) introduced pressure-coupled relaxations, found that convergence can be greatly improved by using F-cycles instead of V-cycles, and also found that arithmetic averaging of viscosities to the coarse grid gives greater robustness than harmonic averaging and a similar performance to the Trompert and Hansen (1996) scheme. Albers (2000) introduced mesh refinement, and also found that robustness is greatly improved by using multigrid cycles that conduct more iterations on the coarse grids such F-cycles, W-cycles and V-cycles with more coarse iterations. Kameyama et al. (2005) introduced a new way of conceptualising the iteration process, namely "pseudo-compressibility, and again found that taking additional coarse-grid iterations greatly improves robustness to large viscosity variations.

Most of these improvements have been tried in Stag3D, with the result that viscosity contrasts in the range 10^5 to 10^6 could be routinely handled (e.g., Ratcliff et al., 1997; Tackley, 1998a, 2000a), but this is still much lower than Earth-like. An additional scheme has been implemented in the new version, as described below.

3.5. Pressure interpolation scheme

The "standard" transfer operators used in this code are linear interpolation for prolongation and restriction, which includes arithmetic averaging of the viscosity field to the coarse grids, chosen solely for the reason that it generally gives better convergence for large viscosity contrasts, as also noted by Auth and Harder (1999). Here, it is also not attempted to implement the full matrix-dependent transfers plus GCGA, but rather the philosophy behind matrix-dependent operators is used to propose an improvement to pressure interpolation.

Based on experimentation, it has been found that the main cause of non-convergence with large viscosity variations in Stag3D is pressure corrections passed from coarse to fine grids. The pressure correction is roughly proportional to local viscosity, as discussed earlier. If a fine grid cell has a much lower viscosity than the coarse-grid cell that contains it, then the prolongated pressure correction can be much too large, making the fine grid solution worse. Indeed, this is probably why Trompert and Hansen (1996) found that taking additional pressure iterations is so helpful: the additional iterations are needed to repair the damage done by the "correction" from the coarser grid. A simple remedy would thus seem to be to adjust the pressure corrections by the ratio of fine grid to coarse-grid viscosity. This was found to give improvement in some cases, but not robustly. Instead, it is chosen to adjust the prolongated pressure according to the term $(\partial R^{\mathrm{cont}}/\partial P) \equiv (\partial(\nabla\cdot(\rho\underline{v}))/\partial P)$ introduced earlier, which can be regarded as containing a sort of weighted average of local viscosity values rather than the viscosity at an individual point. Specifically:

$$\delta P_{\mathrm{fine}} = \frac{C\delta P_{\mathrm{coarse}}}{(\partial R_{\mathrm{cont}}/\partial P)_{\mathrm{fine}}} \tag{13}$$

where C is a constant. Noting that in 3D one coarse-grid cell maps to eight fine-grid cells, C is computed using the criterion that the average pressure must be conserved, i.e.,

$$\tfrac{1}{8}\sum \delta P_{\mathrm{fine}} = \delta P_{\mathrm{course}} \tag{14}$$

leading to

$$C = 8\left(\sum \frac{1}{(\mathrm{d}R_{\mathrm{cont}}/\mathrm{d}P)_{\mathrm{fine}}}\right)^{-1} \tag{15}$$

This reduces to simple injection in the case of constant viscosity (i.e., eight fine grid pressures are set equal to the coarse-grid pressure). This scheme is something like a matrix-dependent prolongation operator for pressure. In matrix-dependent operator theory, the restriction operator should be the transpose of the prolongation operator. Curiously, this was not found to be helpful in this application. Similar operators have been tried for the velocity components, but again did not seem to help significantly. Convergence tests comparing the performance of this scheme to the standard linear interpolation are given later.

It is emphasised that pressure and velocity are treated (iterated on) together at every multigrid level, and thus a velocity–pressure solution is obtained in a single set of multigrid cycles. This is different from the common practice in finite-element codes (e.g., Moresi and Solomatov, 1995; Zhong et al., 2000), in which multigrid cycles relax only the velocity and separate, outer iterations must be done for pressure. In that case, several sets of multigrid cycles on velocity alternating with some type of iterations on pressure are required, with the result that it can take as much as 10 times more CPU time to obtain a velocity–pressure solution, compared with treating them simultaneously as is normally done in codes like StagYY (J. van Hunen, personal communication, 2006), although recent implementations of the finite-element multigrid method may be more efficient (S. Zhong, personal communication, 2008).

3.6. Advection and energy equation

The energy equation is advanced in time using an explicit method. Viscous dissipation, diffusion, and adiabatic heating/cooling are calculated using finite-differences. Latent heat effects due to phase transitions are included in the form of an effective heat capacity and thermal expansivity, as introduced by Christensen and Yuen (1985) (in previous versions these terms were treated by advecting potential temperature, but this has now been changed).

Thermal advection is performed using the finite-volume MPDATA scheme of (Smolarkiewicz, 1984), which uses a correction scheme to subtract the numerical diffusion of the upwind donor cell method. This scheme is written for a cartesian domain, so to use it without modification in spherical geometry the velocities are transformed into face mass fluxes and a correction is made for cell volume. This scheme can also be used for a non-diffusive compositional field in conjunction with the "Lenardic filter" (Lenardic and Kaula, 1993), a combination that works quite well for stable layers with sharp interfaces (e.g., as tested in Tackley and King (2003)). Tracers are advecting using a standard second-order or fourth-order Runge–Kutta, taking care to include the correction terms for spherical geometry.

3.7. Parallelisation

A simple domain decomposition is applied in all three spherical coordinate directions and the yin and yang blocks. Each subdomain contains an extra sheet of "ghost" cells that contain copies of the cells on the outer part of adjacent subdomains and act as the boundary condition for the local subdomain, a commonly used approach. After a field is updated, these ghost values are communicated. Simple-mindedly, it would seem necessary to communicate with up to 26 other nodes in order that edge and corner ghost points are correctly transferred; this can, however, be accomplished through communication with only six other nodes by using three consecutive communication steps in the three orthogonal directions, each with the two nodes that contain adjacent subdomains in the relevant direction. Tracer particles are held in the subdomains in which they are present, and are communicated with other subdomains when they cross the boundaries. If a tracer is present in the overlapping region of the yin-yang grid, then a copy is held in each subgrid.

If a single grid block is being used (cartesian or regional spherical) then the communication patterns remain simple regardless of the number of nodes being used, but this is not the case for communication between the "yin" grid and the "yang" subgrid. If each of these subgrids is split more than four ways (i.e., bisected in both the theta and phi directions), then communication along the boundary requires each node to communicate with two or more nodes on the other grid. Thus, for simplicity the decomposition is presently limited to a maximum of four ways, resulting in an eight-way azimuthal decomposition. The grid is also split in the radial direction, typically eight ways so that cases can be run on 64 CPUs.

The multigrid algorithm involves going to very coarse grids, on which calculations are not efficient in parallel (or there might even be fewer points than CPUs). Thus, at some point in the coarsening process the calculation is moved to a single CPU (or actually, duplicated on all CPUs), and then split again during the coarse-to-fine process. With a relatively small number of CPUs like 64, this is done only for the very coarsest grid. A future expansion to more CPUs might require introducing intermediate steps in this process.

The code is parallelised for distributed memory computers using the MPI message-passing library, although it may also be run on a machine without MPI installed by linking to a file containing dummy MPI calls. The code is written as a stand-alone code that can be run on any computer with a Fortran 95 compiler, with no libraries required.

3.8. Rigid body rotation

In a spherical shell with free slip upper and lower boundaries, the solution is undetermined to an arbitrary net (rigid body) rotation. Although this has not been found to be a problem for short-term calculations, over many timesteps net rotation can built up and create problems. Thus, at every timestep the code calculates the net rotation relative to three orthogonal axes and subtracts it from the solution. This net rotation calculation can be done either for the entire volume, or for just the outermost layer. These do not necessarily produce the same answer because there are good reasons for the interior to display net rotation relative to the lithosphere (e.g., Ricard et al., 1991). In order to make it easier to analyse rigid lids and plate tectonics, the usual choice is to subtract the net rotation of the outermost layer.

In a cartesian domain, such a concern also applies in the case of periodic side boundaries. In this case, the mean horizontal flow is calculated and subtracted at every timestep.

4. Results

4.1. Parallel performance

The performance and scaling of StagYY on up to 64 CPUs of a Beowulf cluster with resolutions from 25 million to more than 1 billion unknowns is shown in Fig. 2. The number of unknowns is four times the number of grid points. For this test, the Gonzales cluster at ETH was used, which consists of nodes containing dual AMD Opteron 250 CPUs connected with a Quadrics QsNet II interconnect. Fig. 2 (top row) shows the time taken per multigrid F-cycle. In Fig. 2a, perfect scaling would be a line of slope −1. In Fig. 2b, perfect scaling would be a straight line with a slope of +1. These graphs indicate that the time is roughly proportional to the number of unknowns and inversely proportional to the number of CPUs, as hoped for. One F-cycle with 1.2 billion unknowns

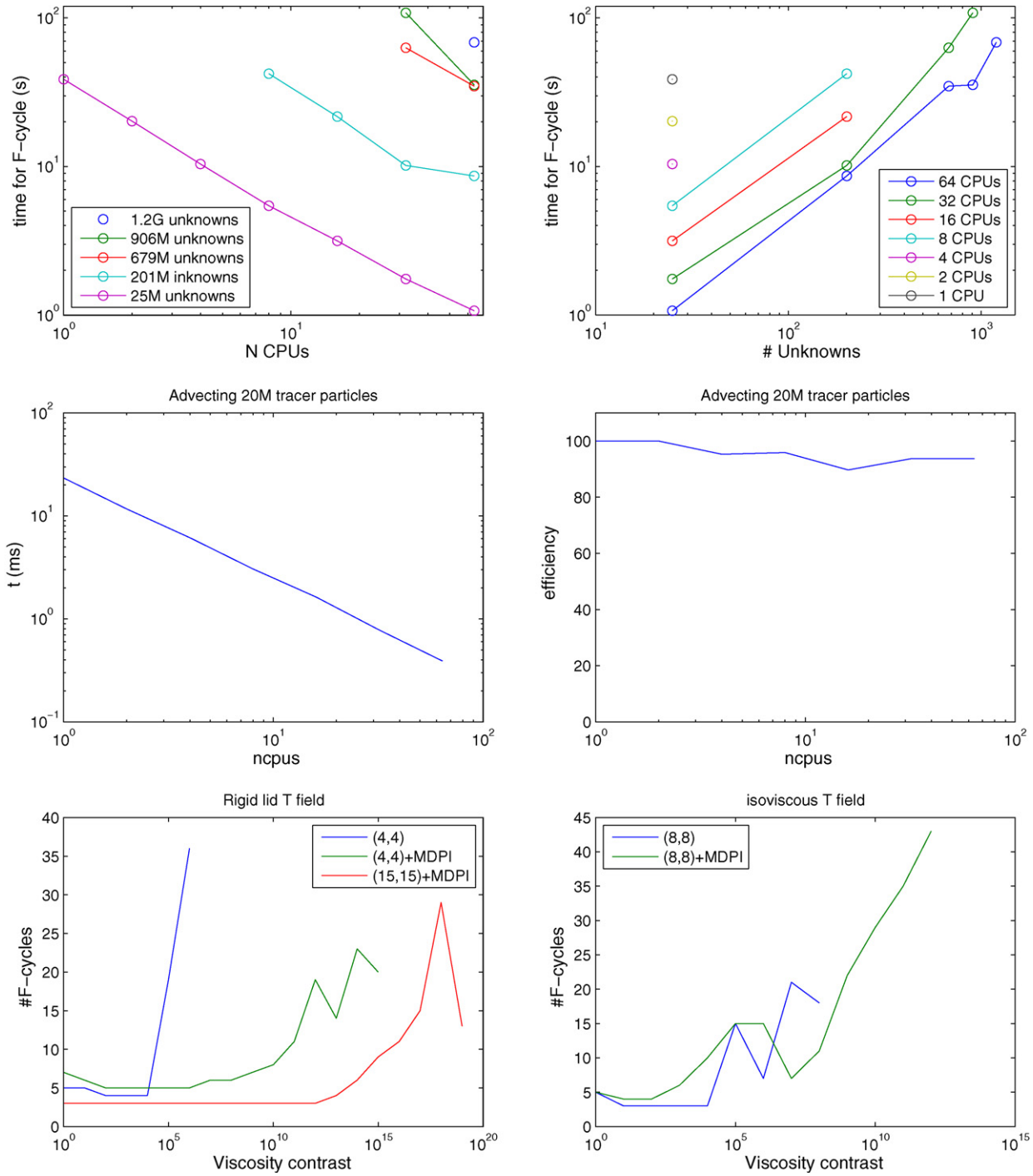

Fig. 2. Various tests of code performance. (Top row) scaling of time for a multigrid F-cycle with number of CPUs and number of unknowns. (Middle row) time required to advect 20 million tracer particles as a function of number of CPUs, and the parallel efficiency. (Bottom row) number of multigrid F-cycles required for 3 orders of magnitude of convergence as function of temperature-dependent viscosity contrast with either a rigid lid temperature field or an isoviscous convection temperature field.

takes about 68 s on 64 CPUs (i.e., 32 dual-CPU nodes), allowing a velocity–pressure solution to be obtained from scratch in less than 10 min. It is encouraging that over 1 billion unknowns can be solved for in relatively few CPUs compared to the number available on many modern supercomputers. The grid in that case is 256 in radius by 1532×512 azimuthally for each of the two blocks.

The middle two parts of Fig. 2 show the time required to advect 20 million tracer particles on different numbers of CPUs, and the parallel efficiency of the process. The efficiency remains about 90% on up to 64 CPUs.

In general, the speed of this new, Fortran 95 version of the code is almost a factor of 2 slower than the original version written in Fortran 77, even for cartesian geometry, and even though the number of floating point operations has been reduced by pre-calculating and storing the finite-difference stencils rather than putting finite-difference operators directly in the residue calculations. One possible explanation for this is the much greater memory access required to retrieve these finite-difference weights from the memory, i.e., retrieving them from memory might take longer than calculating them on the fly. Another possible explanation is the

use of Fortran 95 dynamically allocated arrays and defined types, whereas the original version was written in Fortran 77 with much simpler data structures such as compiled-in array sizes. More work is needed to determine this and optimise the single-CPU floating-point performance of the latest version.

4.2. Multigrid convergence

In general, the multigrid convergence obtained with the yin-yang grid is not as good as that obtained in a single spherical block or in cartesian geometry. Plotting the location of the residue indicates that this is due to the interface between the two subgrids. Further work is needed to understand and improve this situation, but the present convergence is certainly rapid enough for the code to be a useful scientific tool. As with previous studies discussed earlier, it is found that F-cycles give better convergence than V-cycles, and that additional iterations at coarse levels further enhance performance. Thus, the standard iteration parameters are F-cycles with four times as many iterations at the coarse levels as at the finest level, plus various points discussed earlier: red–black iterations with $\alpha_m = 0.7$ and $\alpha_c = 1.0$, geometrical interpolation of viscosity to the shear stress points and arithmetic averaging of viscosity to coarse grids. Note that it is fine to use a different average for coarse-grid viscosity because coarse-grid corrections do not affect the accuracy of the final solution, only the convergence rate towards that solution. All averaging or interplation is thus done using a linear (arithmetic) method, except for the interpolation of viscosity on the same grid level, which is geometric. Another point is that when calculating the rms. residue, residues are normalised by the local viscosity as represented by the stencil weight. The effect of this is to measure the error in absolute velocity, otherwise in regions of high viscosity small velocity variations give enormous residues. In the case of non-convergence, the code includes an automated process that reduces the fraction of the prolongated coarse-grid correction that is applied to the fine grid, which often helps but indicates the need for optimal prolongation and restriction operators.

Fig. 2 (bottom row) shows the number of F-cycles needed to reduce the residue to 10^{-3} of its initial value, as a function of viscosity contrast and the iteration parameters such as number of iterations at each level. Two initial conditions are used: rigid lid convection with a viscosity contrast of 10^6, and constant-viscosity convection. Of particular interest is the effect of the "matrix-dependent" pressure-interpolation (MDPI) scheme described above. Similar plots were made by Albers (2000) and Kameyama et al. (2005) to show the effect of various smoothers and multigrid cycles.

Using the rigid lid T field, F-cycles with four smoothing iterations at each step and no MDPI are able to handle a maximum of 6 orders of magnitude viscosity contrast, with the number of F-cycles increasing rapidly above 4 orders of magnitude. Switching on the MDPI scheme increases the maximum viscosity contrast to 15 orders of magnitude, with the required number of cycles increasing significantly only above about 10 orders of magnitude. Additional robustness can be obtained by increasing the number of smoothing iterations at each level, for example with 15 iterations the maximum viscosity contrast increases to 19 orders of magnitude.

With this rigid lid T field, most of the viscosity contrast occurs gradually over the conductive lid, such that the maximum contrast between adjacent points is a factor 54,000 with 19 orders of magnitude global contrast. Starting from an isoviscous T field is more challenging because the large temperature hence viscosity contrasts occur over very short lengthscales; this is an "unrealistic" viscosity field but useful for testing purposes. Thus, it was found necessary to increase the number of smoothing iterations to eight at each level to get reasonable convergence (Fig. 2 bottom right), and with this up to 8 orders of magnitude were possible without MDPI. With MDPI this increases to 12 orders of magnitude but a large number of F-cycles is necessary. In that case the contrast between adjacent points is as much as 196,000.

The main conclusion is thus that MDPI can dramatically improve the robustness to large viscosity variations. Of course, this refers to the convergence of the numerical scheme to the discretized solution, not the accuracy of the discretized solution, which needs to be tested separately. At low viscosity contrasts MDPI sometimes increases the needed number of F-cycles: this is probably because the interpolation scheme is lower order, i.e., injection rather than linear.

4.3. Benchmark tests

Extensive testing and benchmarking of Stag3D in cartesian geometry has been performed and reported in previous publications. For thermal convection with constant or temperature-dependent viscosity, Stag3D can successfully reproduce the two-dimensional benchmark cases in Blankenbach et al. (1989) and three-dimensional benchmark cases in Busse et al. (1994), as detailed in Tackley (1994) and summarized in Tackley (1996). For cases with self-consistently generated plate tectonics, in which large viscosity contrasts occur over very short lengthscales, a convergence test was presented in the Appendix of Tackley (2000b). This test showed the effect of resolution, varying in factors of 2 from $16 \times 16 \times 4$ to $256 \times 256 \times 64$, on the outcome of a case with self-consistent plates. Remarkably, plate-like behaviour was obtained with all resolutions above $32 \times 32 \times 8$, but at lower resolutions the weak zones ("plate boundaries") tend to follow grid lines. For thermo-chemical convection, Tackley and King (2003) presented benchmark tests and convergence tests for both Stag3D and the finite-element code ConMan (King et al., 1990). These demonstrated that Stag3D can reproduce the earlier benchmark tests of van Keken et al. (1997), and compared convergence tests for tracer-based and grid-based methods of representing composition in thermo-chemical convection, determining the needed resolution and number of tracers.

Here some tests are presented to benchmark StagYY against results obtained by other codes. For steady-state basal-heated Boussinesq convection with $Ra_{1/2} = 7000$ and a tetrahedral arrangement of upwelling plumes, Stemmer et al. (2006) compiled Nusselt numbers and rms velocities for eight different codes including their own, namely those by Bercovici et al. (1989b), Harder (1998), Iwase (1996), Ratcliff et al. (1996b), Tabata and Suzuki (2000), Yoshida and Kageyama (2004), Zhong et al. (2000), so this provides an ideal test case. Table 1 gives results for StagYY compared to the Romberg extrapolated results from Stemmer et al. (2006) (Table 5, see also Table 1 and Fig. 4 for other codes). Temperature isosurfaces for these two cases are shown in Fig. 1d and e.

For the most accurate case (196,600 cells, second-order advection) the results are very consistent with those summarised in Stemmer et al. (2006). The order of the advection scheme makes more difference than the number of cells—of course the first-order scheme is hopeless for real applications but it is interesting to see the difference that advection makes. This suggests that an even more accurate advection scheme could yield even better results. The MPDATA scheme is known to have a significant amount of numerical diffusion (e.g., Muller, 1992) so it is planned to implement a modern, less diffusive scheme in StagYY.

For higher viscosity contrasts and Rayleigh number, cases at $Ra = 10^5$ and viscosity contrast up to 10^6 are presented for comparison with Stemmer et al. (2006) and Ratcliff et al. (1996a, 1997). These cases are slightly time-dependent and do not have a par-

Table 1
StagYY results for the tetrahedral pattern at Ra = 7000, isoviscous or viscosity contrast 20

nr, nt, np, nb	# Cells	Adv. order	$\Delta\eta = 1$:Nu	v	$\Delta\eta = 20$:Nu	v
$16 \times 16 \times 48 \times 2$	24,576	1	3.62	32.87	3.26	25.61
		2	3.49	32.40	3.13	25.51
$32 \times 32 \times 96 \times 2$	196,608	1	3.57	32.98	3.22	25.85
		2	3.48	32.57	3.15	25.71
Stemmer et al.	Extrapolated		3.4949	32.6234	3.1526	25.76

The last row lists the results from Stemmer et al. (2006) Table 5 which use a Romberg extrapolation of results at increasing resolutions.

Table 2
Basally heated cases with $Ra_{1/2} = 10^5$ and viscosity contrast up to 10^6

$\Delta\eta$	Nu	v
1	7.27	160.2
10^3	6.06	96.8
10^6	6.50	40.4

These are with 196,608 cells as listed above and second order advection.

ticular symmetry. They are started from a conductive profile with random perturbations. Due to the random nature, the resulting pattern is not expected to exactly match previously published cases, but the results are within the range of previous results, as illustrated in Fig. 1g–i and Table 2.

For a more complex scenario including compressibility and a phase transition, an attempt is made to reproduce the result of Tackley et al. (1993). The present reference state uses a different parameterisation but is tuned to match depth profiles of physical properties as closely as possible. The results (Fig. 1f) show a good resemblance, with the upper mantle containing linear, time-dependent downwellings and the lower mantle containing cylindrical "avalanches". It is reassuring that two codes using completely different numerical methods (spectral versus finite-difference) are able to obtain similar results with relatively complex physics.

4.4. Other example results

Scientific findings using StagYY are being detailed in other papers; here a few sample results are included for illustration. Fig. 1j–l shows a thermo-chemical, multiple phase transition case reported fully in Nakagawa and Tackley (2008). This case uses tracers to represent the compositional field, using the ratio method (Tackley and King, 2003). Chemically dense material is swept into piles in upwelling regions, and the locations where the post-perovskite phase are present (Fig. 1l) are anticorrelated with these piles.

Self-consistent generation of plate tectonics has been a major interest in the field, so Fig. 1m–o reproduces a case similar to those in Tackley (2000a,b) except in spherical rather than cartesian geometry. In these cases, plastic yielding breaks the rigid lid. For this particular parameter combination a novel platform is found consisting of an approximately great circle downwelling (Fig. 1o) (van Heck and Tackley, in press).

5. Discussion and future directions

This paper documents how the use of the yin-yang grid has allowed an existing cartesian code, Stag3D (Tackley, 1993), to be straightforwardly converted to model a 3D spherical shell. Although there are now two other mantle convection codes that use the yin-yang grid (those of Yoshida and Kageyama (2004) and Kameyama et al., 2008), the one reported here is the latest evolution of a code that has been in continuous use and development for over 14 years so has more features and implemented physics. Compared to the code of Yoshida and Kageyama (2004) these include compressibility, phase transitions, compositional variations, non-linear rheology, parallelisation, tracers to track composition, and the ability to model spherical patches, cartesian boxes, and various 2D geometries by changing one input switch. Tests presented show that StagYY produces results that are consistent with previously published results, and so it is being used to perform new scientific studies. StagYY is designed to be a stand-alone application with no libraries required, but if MPI is present it can be run in parallel.

Convergence of a multigrid solver in the presence of realistically large viscosity variations has always been a problem with such codes, as discussed earlier. In this paper a new pressure interpolation scheme is presented that can dramatically improve the robustness of the iterations to large viscosity variations, with up to 19 orders of magnitude variation in the presented tests. One goal for the future is to further investigate such prolongation and restriction schemes in an attempt to arrive at a perfectly robust solution scheme. Use of the Galerkin coarse-grid approximation may be an ingredient to such a scheme. A related promising technology is the use of algebraic multigrid, which is designed to overcome several limitations of geometric multigrid as used here.

Another goal for the future is to implement grid refinement, which is straightforwardly treated in multigrid schemes by going to finer grid levels in some areas than in others, as illustrated for mantle convection by Albers (2000). This could be fixed, e.g., with the upper mantle being better resolved than the lower mantle and the lithosphere being better resolved still, or adaptive.

Some technical issues remain regarding code performance. Firstly, StagYY is around a factor of two slower than the old version (Stag3D) even in cartesian geometry, which could be due to the higher memory bandwidth required due to the storage of finite-difference weights in memory, or due to more complex data structures made possible by Fortran 95. Investigation and remedy of this should allow substantial performance gains to be realized. Secondly, multigrid cycles converge more slowly with the yin-yang grid than with a single spherical or cartesian block, with the edges somehow slowing things down. Nevertheless, the code is already a useful tool for investigating mantle processes.

Acknowledgements

The author thanks Takashi Nakagawa and Hein van Heck for their images, and Masanori Kameyama, Guillaume Richard and Dani Schmid for constructive reviews.

References

Albers, M., 2000. A local mesh refinement multigrid method for 3-D convection problems with strongly variable viscosity. J. Comput. Phys. 160 (1), 126–150.
Auth, C., Harder, H., 1999. Multigrid solution of convection problems with strongly variable viscosity. Geophys. J. Int. 137 (3), 793–804.
Balachandar, S., Yuen, D.A., Reuteler, D.M., 1996. High Rayleigh number convection at infinite Prandtl number with weakly temperature-dependent viscosity. Geophys. Astrophys. Fluid Dyn. 83 (1–2), 79–117.
Baumgardner, J.R., 1985. 3-dimensional treatment of convective flow in the earths mantle. J. Stat. Phys. 39 (5–6), 501–511.

Baumgardner, J.R., 1988. Application of supercomputers to 3-D mantle convection. In: Runcorn, S.K. (Ed.), The Physics of the Planets. Wiley, New York, pp. 199–231.
Bercovici, D., Schubert, G., Glatzmaier, G.A., 1989a. 3-dimensional spherical-models of convection in the earths mantle. Science 244 (4907),950–955.
Bercovici, D., Schubert, G., Glatzmaier, G.A., Zebib, A., 1989b. 3-dimensional thermal-convection in a spherical-shell. J. Fluid Mech. 206 (Sep.), 75–104.
Blankenbach, B., et al., 1989. A benchmark comparison for mantle convection codes. Geophys. J. Int. 98 (1), 23–38.
Brandt, A., 1982. Guide to multigrid development. Lect. Notes Math. 960,220–312.
Busse, F.H., et al., 1994. 3D convection at infinite Prandtl number in Cartesian geometry—a benchmark comparison. Geophys. Astrophys. Fluid Dyn. 75 (1),39–59.
Choblet, G., 2005. Modelling thermal convection with large viscosity gradients in one block of the 'cubed sphere'. J. Comput. Phys. 205 (1), 269–291.
Christensen, U., Harder, H., 1991. 3-D convection with variable viscosity. Geophys. J. Int. 104 (1), 213–226.
Christensen, U.R., Yuen, D.A., 1985. Layered convection induced by phase transitions. J. Geophys. Res. 90 (B12), 10291–10300.
Connolly, J.A.D., 2005. Computation of phase equilibria by linear programming: a tool for geodynamic modeling and an application to subduction zone decarbonation. Earth Planet. Sci. Lett. 236, 524–541.
Deubelbeiss, Y., Kaus, B.J.P., 2008. A comparison of Eulerian and Lagrangian numerical techniques for the Stokes equation in the presence of strongly varying viscosity. Phys. Earth Planet. Int. 171, 92–111.
Gerya, T.V., Connolly, J.A.D., Yuen, D.A., Gorczyk, W., Capel, A.M., 2006. Seismic implications of mantle wedge plumes. Phys. Earth Planet. Int. 156, 59–74.
Glatzmaier, G.A., 1988. Numerical simulations of mantle convection—time-dependent, 3-dimensional, compressible, spherical-shell. Geophys. Astrophys. Fluid Dyn. 43 (2), 223–264.
Harder, H., 1998. Phase transitions and the three-dimensional planform of thermal convection in the Martian mantle. J. Geophys. Res. 103 (E7), 16775–16797.
Harder, H., Christensen, U.R., 1996. A one-plume model of martian mantle convection. Nature 380 (6574), 507–509.
Harder, H., Hansen, U., 2005. A finite-volume solution method of thermal convection and dynamo problems in spherical shells. Geophys. J. Int. 161, 522–532.
Hernlund, J.W., Tackley, P.J., 2008. Modeling mantle convection in the spherical annulus. Phys. Earth Planet. Int. 171, 48–54.
Hernlund, J.W., Tackley, P.J., 2003. Three-dimensional spherical shell convection at infinite Prandtl number using the 'cubed sphere' method, Second MIT Conference on Computational Fluid and Solid Mechanics. Elsevier, MIT.
Iwase, Y., 1996. Three-dimensional infinite Prandtl-number convection in a spherical shell with temperature-dependent viscosity. J. Geomag. Geoelectr. 48, 1499–1514.
Kageyama, A., Sato, T., 2004. The "yin-yang grid": an overset grid in spherical geometry. Geochem. Geophys. Geosyst. 5(Q09005): doi:10.1029/2004GC000734.
Kameyama, M., Kageyama, A., Sato, T., 2005. Multigrid iterative algorithm using pseudo-compressibility for three-dimensional mantle convection with strongly variable viscosity. J. Comput. Phys. 206, 162–181.
Kameyama, M., Kageyama, A., Sato, T., 2008. Multigrid-based simulation code for mantle convection in spherical shell using Yin–Yang grid. Phys. Earth Planet. Int. 171, 19–32.
King, S.D., Raefsky, A., Hager, B.H., 1990. Conman—vectorizing a finite-element code for incompressible 2-dimensional convection in the earths mantle. Phys. Earth Planet. Inter. 59 (3), 195–207.
Lenardic, A., Kaula, W.M., 1993. A numerical treatment of geodynamic viscous-flow problems involving the advection of material interfaces. J. Geophys. Res. Solid Earth 98 (B5), 8243–8260.
Machetel, P., Thoraval, C., Brunet, D., 1995. Spectral and geophysical consequences of 3-D spherical mantle convection with an endothermic phase-change at the 670 km discontinuity. Phys. Earth Planet. Inter. 88 (1), 43–51.
Monnereau, M., Quere, S., 2001. Spherical shell models of mantle convection with tectonic plates. Earth Planet. Sci. Lett. 184, 575–587.
Moresi, L.N., Solomatov, V.S., 1995. Numerical investigation of 2D convection with extremely large viscosity variations. Phys. Fluids 7 (9), 2154–2162.
Muller, R., 1992. The performance of classical versus modern finite-volume advection schemes for atmospheric modeling in a one-dimensional test-bed. Mon. Weather Rev. 120 (7), 1407–1415.
Nakagawa, T., Tackley, P.J., 2005. Three-dimensional numerical simulations of thermo-chemical multiphase convection in Earth's mantle. In: Proceedings of the Third MIT Conference on Computational Fluid and Solid Mechanics.
Nakagawa, T., Tackley, P.J., 2008. Lateral variations in CMB heat flux and deep mantle seismic velocity caused by a thermal-chemical-phase boundary layer in 3D spherical convection. Earth Planet. Sci. Lett. 271, 348–358.
Ogawa, M., Schubert, G., Zebib, A., 1991. Numerical simulations of 3-dimensional thermal convection in a fluid with strongly temperature-dependent viscosity. J. Fluid Mech. 233, 299–328.
Patankar, S.V., 1980. Numerical Heat Transfer and Fluid Flow. Hemisphere Publishing Corporation, New York.
Press, W.H., Teulolsky, S.A., Vetterling, W.T., Flannery, B.P., 1992. Numerical Recipes. Cambridge University Press, Cambridge, UK.
Ratcliff, J.T., Schubert, G., Zebib, A., 1995. 3-dimensional variable viscosity convection of an infinite Prandtl number Boussinesq fluid in a spherical-shell. Geophys. Res. Lett. 22 (16), 2227–2230.
Ratcliff, J.T., Schubert, G., Zebib, A., 1996a. Effects of temperature-dependent viscosity on thermal convection in a spherical shell. Phys. D 97 (1–3),242–252.
Ratcliff, J.T., Schubert, G., Zebib, A., 1996b. Steady tetrahedral and cubic patterns of spherical-shell convection with temperature-dependent viscosity. J. Geophys. Res. Solid Earth 101 (B11), 25473–25484.
Ratcliff, J.T., Tackley, P.J., Schubert, G., Zebib, A., 1997. Transitions in thermal convection with strongly variable viscosity. Phys. Earth Planet. Inter. 102, 201–212.
Ricard, Y., Doglioni, C., Sabadini, R., 1991. Differential rotation between lithosphere and mantle—a consequence of lateral mantle viscosity variations. J. Geophys. Res. Solid Earth Planets 96 (B5), 8407–8415.
Ronchi, C., Iacono, R., Paolucci, P.S., 1996. The "cubed sphere": a new method for the solution of partial differential equations in spherical geometry. J. Comput. Phys. 124 (1), 93–114.
Schmeling, H., Babeyko, A.Y., Enns, A., Faccenna, C., Funiciello, F., Gerya, T., Golabek, G.J., Grigull, S., Kaus, B.J.P., Morra, G., Schmalholz, S.M., van Hunen, J., 2008. A benchmark comparison of spontaneous subduction models—towards a free surface. Phys. Earth Planet. Inter. 171, 198–223.
Schubert, G., Turcotte, D.L., Olson, P., 2000. Mantle Convection in the Earth and Planets. Cambridge University Press.
Smolarkiewicz, P.K., 1984. A fully multidimensional positive definite advection transport algorithm with small implicit diffusion. J. Comput. Phys. 54 (2), 325–362.
Stemmer, K., Harder, H., Hansen, U., 2006. A new method to simulate convection with strongly temperature- and pressure-dependent viscosity in a spherical shell: applications to the Earth's mantle. Phys. Earth Planet. Inter. 157 (3–4), 223–249.
Tabata, M., Suzuki, A., 2000. A stabilized finite element method for the Rayleigh–Benard equations with infinite Prandtl number in a spherical shell. Comput. Methods Appl. Mech. Eng. 190, 387–402.
Tackley, P.J., 1993. Effects of strongly temperature-dependent viscosity on time-dependent, 3-dimensional models of mantle convection. Geophys. Res. Lett. 20 (20), 2187–2190.
Tackley, P.J., 1994. Three-dimensional models of mantle convection: influence of phase transitions and temperature-dependent viscosity. Ph.D. Thesis. California Institute of Technology, Pasadena, 299 pp.
Tackley, P.J., 1996. Effects of strongly variable viscosity on three-dimensional compressible convection in planetary mantles. J. Geophys. Res. 101, 3311–3332.
Tackley, P.J., 1998a. Self-consistent generation of tectonic plates in three-dimensional mantle convection. Earth Planet. Sci. Lett. 157, 9–22.
Tackley, P.J., 1998b. Three-dimensional simulations of mantle convection with a thermochemical CMB boundary layer: D? In: Gurnis, M., Wysession, M.E., Knittle, E., Buffett, B.A. (Eds.), The Core-Mantle Boundary Region. Geodynamics. American Geophysical Union, pp. 231–253.
Tackley, P.J., 2000a. Self-consistent generation of tectonic plates in time-dependent, three-dimensional mantle convection simulations. Part 1. Pseudo-plastic yielding. Geochem., Geophys., Geosys. Volume 1, Paper number 2000GC000036 [14,503 words, 21 figures, 1 table].
Tackley, P.J., 2000b. Self-consistent generation of tectonic plates in time-dependent, three-dimensional mantle convection simulations. Part 2. Strain weakening and asthenosphere. Geochem., Geophys., Geosys. Volume 1: Paper number 2000GC000043 [14,420 words, 15 figures, 1 table].
Tackley, P.J., 2002. Strong heterogeneity caused by deep mantle layering. Geochem. Geophys. Geosys. 3(4), doi:10.1029/2001GC000167.
Tackley, P.J., King, S.D., 2003. Testing the tracer ratio method for modeling active compositional fields in mantle convection simulations. Geochem. Geophys. Geosyst. 4(4), doi:10.1029/2001GC000214.
Tackley, P.J., Stevenson, D.J., Glatzmaier, G.A., Schubert, G., 1993. Effects of an endothermic phase transition at 670 km depth in a spherical model of convection in the Earth's mantle. Nature 361 (6414), 699–704.
Tackley, P.J., Stevenson, D.J., Glatzmaier, G.A., Schubert, G., 1994. Effects of multiple phase transitions in a 3-dimensional spherical model of convection in Earth's mantle. J. Geophys. Res. 99 (B8), 15877–15901.
Tackley, P.J., Xie, S., 2003. Stag3D: a code for modeling thermo-chemical multiphase convection in Earth's mantle. In: Second MIT Conference on Computational Fluid and Solid Mechanics, Elsevier, MIT, pp. 1–5.
Trompert, R.A., Hansen, U., 1996. The application of a finite-volume multigrid method to 3-dimensional flow problems in a highly viscous fluid with a variable viscosity. Geophys. Astrophys. Fluid Dyn. 83 (3–4), 261–291.
van Heck, H., Tackley, P.J., Planforms of self-consistently generated plate tectonics in 3-D spherical geometry, Geophys. Res. Lett., in press.
van Keken, P.E., King, S.D., Schmeling, H., Christensen, U.R., Neumeister, D., Doin, M.P., 1997. A comparison of methods for the modeling of thermochemical convection. J. Geophys. Res. 102 (B10), 22477–22495.
Vanka, S.P., 1986. Block implicit multigrid solution of Navier–Stokes equations with primitive variables. J. Comput. Phys. 65, 138–158.
Wesseling, P., 1992. An Introduction to Multigrid Methods. John Wiley and Sons, 284 pp.
Xie, S., Tackley, P.J., 2004. Evolution of helium and argon isotopes in a convecting mantle. Phys. Earth Planet. Inter. 146 (3–4), 417–439.
Yang, W.-S., Baumgardner, J.R., 2000. A matrix-dependent transfer multigrid method for strongly variable viscosity infinite Prandtl number thermal convection. Geophys. Astrophys. Fluid Dyn. 92, 151–195.
Yoshida, M., Iwase, Y., Honda, S., 1999. Generation of plumes under a localized high viscosity lid in 3-D spherical shell convection. Geophys. Res. Lett. 26 (7), 947–950.

Yoshida, M., Kageyama, A., 2004. Application of the yin-yang grid to a thermal convection of a Boussinesq fluid with infinite Prandtl number in a three-dimensional spherical shell. Geophys. Res. Lett. 31(L12609), doi:10.1029/2004GL019970.

Yoshida, M., Kageyama, A., 2006. Low-degree mantle convection with strongly temperature- and depth-dependent viscosity in a three-dimensional spherical shell. J. Geophys. Res. 111(B03412), doi:10.1029/2005JB003905.

Young, R.E., 1974. Finite-amplitude thermal convection in a spherical shell. J. Fluid Mech. 63, 695–721.

Zeng, S., Wesseling, P., 1992a. An efficient algorithm for the computation of Galerkin coarse-grid approximation for the incompressible Navier–Stokes equations, Faculty of Technical Mathematics and Informatics, Delft University of Technology.

Zeng, S., Wesseling, P., 1992b. Galerkin coarse grid approximation for the incompressible Navier–Stokes equations in general coordinates, Faculty of Technical Mathematics and Informatics, Delft University of Technology.

Zhang, S.X., Christensen, U., 1993. Some effects of lateral viscosity variations on geoid and surface velocities induced by density anomalies in the mantle. Geophys. J. Int. 114 (3), 531–547.

Zhang, S.X., Yuen, D.A., 1996. Various influences on plumes and dynamics in time-dependent compressible mantle convection in 3-D spherical-shell. Phys. Earth Planet. Inter. 94 (3–4), 241–267.

Zhong, S., Zuber, M.T., Moresi, L., Gurnis, M., 2000. Role of temperature-dependent viscosity and surface plates in spherical shell models of mantle convection. J. Geophys. Res. 105 (B5), 11063–11082.