
ELSEVIER

Parallel Computing 25 (1999) 361–380

PARALLEL COMPUTING

Practical aspects

Computational aspects of a code to study rotating turbulent convection in spherical shells

T.C. Clune [a], J.R. Elliott [a,*], M.S. Miesch [a], J. Toomre [a], G.A. Glatzmaier [b]

[a] *JILA, University of Colorado, Campus Box 440, Boulder, CO 80309-0440, USA*
[b] *IGPP, MS C305, Los Alamos Nat. Lab., Los Alamos, NM 87545, USA*

Received 13 February 1998; received in revised form 5 August 1998

Abstract

The coupling of highly turbulent convection with rotation within a full spherical shell geometry, such as in the solar convection zone, can be studied with the new anelastic spherical harmonic (ASH) code developed to exploit massively parallel architectures. Inter-processor transposes are used to ensure data locality in spectral transforms, a sophisticated load balancing algorithm is implemented and the Legendre transforms, which dominate the workload for large problems, are highly optimized by exploiting the features of cache memory and instruction pipelines. As a result, the ASH code achieves around 120 Mflop/s per node on the Cray T3E and scales nearly linearly for adequately large problem sizes. © 1999 Published by Elsevier Science B.V. All rights reserved.

Keywords: Convection; Anelastic spherical harmonic code

1. Introduction

The dynamics of the vigorous convection in the outer envelope of a late-type star such as the sun must determine the transport properties of energy, angular momentum and magnetic fields within that layer and consequently must be responsible for much of the observed activity seen at the surface, as well as the differential

* Corresponding author. E-mail: jelliott@solarz.colorado.edu

0167-8191/99/$ – see front matter
PII: S0167-8191(99)00009-5

rotation profile deduced from inversions of helioseismology data. Theoretical modeling of such convection suffers from the complexity of the problem: the motions are compressible, turbulent, supersonic in places and are influenced by rotation, magnetic fields, changes in the equation of state and radiative transfer effects near the surface. Although simple mixing-length models provided a good first approximation to the structure of the convection zone, most of the recent advances in the studies of convection have come from numerical simulations. These may be classified according to the simplifications adopted to make the problem more tractable. In the *f*-plane simulations [2,3] only a small, localized planar slice is considered. This has the advantage that the computational resolution can be sufficient to study highly turbulent states, even when dealing with the full, compressible Navier–Stokes fluid equations. However, in order to understand the global distribution of angular momentum in the convection zone, it is essential to deal with a full spherical shell so as to capture the effects of geometry and of the varying angle between local gravity and the rotation vector. Early studies of incompressible convection in spherical shells were carried out under the Boussinesq approximation [5,6]. Since the effects of density stratification across the depth of the convection zone are expected to be significant, we choose to use the anelastic approximation, described by Gough [14] and Latour et al. [15], and adapted to the present situation by Gilman and Glatzmaier [7]. This approximation admits mean density variations but filters out sound waves so as to prevent the relatively short acoustic time scale from limiting the time step. Sound waves are not expected to have a significant effect on transport properties in the convection zone.

Early studies of spherical shell anelastic convection were carried out by Glatzmaier [9–11]. He considered a stratified, rotating, spherical fluid shell heated from below, and expanded perturbations to the magnetic field, velocity and thermodynamic variables in spherical harmonics to resolve their horizontal structure and in Chebyshev polynomials to resolve their radial structure. He used an explicit Adams–Bashforth time integration scheme for the advection and Coriolis terms, and an implicit Crank–Nicholson treatment of the diffusion terms. Nonlinear terms were computed in physical space, while spatial derivatives were computed in spectral space. In the laminar cases which were studied, he found an equatorial acceleration driven by so-called 'banana cells', with angular velocity essentially constant on cylinders and decreasing with depth in the equatorial plane; these results were confirmed by Gilman and Miller [8], and most recently by Tilgner and Busse [17]. The sun does not show such a pattern of differential rotation, as judged so far from helioseismic observations, for its angular velocity at high latitudes appears to increase with depth, at mid-latitudes is nearly constant on radial lines, and near the equator first increases and then gently decreases with depth.

There is no evident solution yet to the solar differential rotation puzzle. A likely explanation is that the spatial resolution of early global convection models (spherical harmonic degree up to ~ 50) was capable of only describing mildly turbulent flows, whereas fully developed turbulence with its modified transport of angular momentum may yield quite different mean flows and rotation profiles.

The advent of parallel computers has enabled studies of convection in spherical shells [12] to reach higher spatial resolutions, and consequently more turbulent conditions than was previously possible. This has revealed convection that is highly time dependent, involving flows that are fairly complex, with little that resembles the banana cells of the earlier investigations. The convection involves intermittent plumes of upflow and downflow, with the downflows stronger in amplitude, though there is still some north–south alignment of structures close to the equator. The mean zonal velocity profiles have shown prograde motion (fast compared to the rotating reference frame) at low latitudes, a belt of retrograde motion (slow) at mid latitudes, and a polar vortex of prograde motion (fast) near the poles. These profiles are somewhat suggestive of the helioseismic inferences [16] at low latitude.

We aim to extend studies of convection in spherical shells to considerably higher spatial resolutions (spherical harmonic degree up to ~ 500) using our new anelastic spherical harmonic (ASH) code. The ASH code uses the basic approach of Glatzmaier [9], but is highly optimized to run efficiently on massively parallel machines. The perturbation fields are expanded as spherical harmonics to resolve their horizontal structure and in Chebyshev polynomials to resolve their radial structure. Evolution is carried out in time using the same combination of an Adams–Bashforth explicit scheme for the Coriolis and advective terms and a Crank–Nicholson implicit method for the diffusive terms. The spherically averaged mean values of the thermodynamic variables (sometimes referred to as the reference state) are updated using the spherically averaged perturbation fields; such a nonlinear feedback of the perturbations on the mean stratification is a crucial feature. Optimization is achieved in three ways: the transformation dimension for spectral transforms is made local by performing inter-processor transposes using the message-passing strategy; a sophisticated load-balancing algorithm is used to minimize processor latency; and single node performance is maximized by carefully exploiting the structures of cache memory and instruction pipelines.

The use of scaleably parallel technology and a very efficient code has allowed us to attain for the first time sufficient resolution to realize substantially turbulent solutions for convection within a full spherical shell. The results to date are very promising and justify our belief that the modified transport properties within turbulent systems provide possible explanations for the remarkable angular velocity profiles being deduced from helioseismology.

This paper is divided into six sections besides the introduction. In Section 2 we formulate the anelastic fluid equations and present our numerical algorithm for solving them in a spherical shell. The details of the parallel implementation are discussed in Section 3, with particular attention to how data structures are partitioned to ensure load balancing. Section 4 shows performance examples which demonstrate the effectiveness of this implementation. Section 5 briefly discusses results from a single run of the ASH code, with conclusions given in Section 6. For completeness, Appendix A lists the equations as they are solved by the ASH code.

2. Formulation

2.1. The fundamental equations

We begin with the compressible fluid equations for a rotating reference frame, comprising the continuity equation,

$$\frac{\partial \rho}{\partial t} = -\nabla \cdot (\rho \mathbf{v}), \tag{1}$$

the momentum equation,

$$\rho \frac{\partial \mathbf{v}}{\partial t} = -\rho(\mathbf{v} \cdot \nabla)\mathbf{v} - \nabla P - \nabla \cdot \mathscr{D} - \rho g \mathbf{r} + 2\rho(\mathbf{v} \times \mathbf{\Omega}) \tag{2}$$

and the energy equation,

$$\begin{aligned} \rho T \frac{\partial S}{\partial t} = & -\rho T \mathbf{v} \cdot \nabla S - \nabla \cdot (-\kappa \rho T \nabla S - \kappa_{\mathrm{rad}} \rho c_{\mathrm{P}} \nabla T) \\ & + 2\nu\rho \left[e_{ij} e_{ij} - 1/3(\nabla \cdot \mathbf{v})^2 \right], \end{aligned} \tag{3}$$

where $\mathscr{D}$ is the viscous stress tensor, with components

$$\mathscr{D}_{ij} = -2\rho\nu \left[e_{ij} - 1/3(\nabla \cdot \mathbf{v})\delta_{ij} \right], \tag{4}$$

e_{ij} is the strain rate tensor, ρ, T, S and P the density, temperature, specific entropy and pressure, respectively, c_{P} the specific heat at constant pressure, g the local gravitational acceleration, $\mathbf{r}$ a unit vector in the radial direction, $\mathbf{v}$ the velocity and $\mathbf{\Omega}$ the angular velocity of the rotating frame. Since viscous forces due to molecular viscosity are extremely small relative to inertial forces (corresponding to an extremely large characteristic Reynolds number of $O(10^{12})$) , direct simulation of stellar convection zones at scales ranging from the global scale down to the molecular dissipation scale are impossible given the speed of current computers and will remain so in the foreseeable future. For this reason, the kinematic viscosity ν should be regarded as a turbulent, *eddy* viscosity, arising from unresolved fluid motions on scales smaller than the grid spacing of the model. Such unresolved motions also give rise to a turbulent heat flux proportional to the gradient of the entropy, with κ in Eq. (3) being the corresponding diffusivity. κ_{rad} is the thermal diffusivity arising from radiative diffusion and thermal conduction [13]. ν, κ and κ_{rad} and g are fixed functions of radius alone which are set in the initial conditions.

The equation of state is taken to be that of an ideal monatomic gas,

$$P = \mathscr{R}\rho T, \tag{5}$$

where $\mathscr{R}$ is the gas constant and c_{P} is set to the ideal gas value, $5\mathscr{R}/2$.

Each of the thermodynamic variables, ρ, T, S and P is now rewritten as the sum of a spherically averaged part, denoted by a horizontal bar and a perturbation, e.g.

$$\rho(r, \theta, \phi, t) \rightarrow \overline{\rho}(r, t) + \rho(r, \theta, \phi, t), \tag{6}$$

where

$$\frac{|\rho|}{\overline{\rho}} \approx \varepsilon, \tag{7}$$

ε being an expansion parameter reflecting the amplitude of the perturbations, which scales like the square of the Mach number of the flow (the ratio of the velocity to the local sound speed). Since the Mach number is very small in the bulk of the convection zone, ε is also small, and we may make the anelastic approximation [15], which consists of assuming

$$\nabla \cdot (\overline{\rho}\mathbf{v}) = 0, \tag{8}$$

effectively filtering out sound waves from the problem. Under this approximation, the maximum time step is much longer than in the fully compressible case, since the Courant–Freidrichs–Lewy (CFL) condition applies not to the sound speed, but to the much smaller convective velocity. Notice that although the time derivative of the density does not appear in the anelastic continuity equation, the density does vary in time, but only on a convective time scale.

Using Eq. (6) and corresponding equations for T, S and P to linearize Eqs. (2) and (3), and making use of Eq. (8), we obtain

$$\frac{\partial \overline{\rho}\mathbf{v}}{\partial t} = -\overline{\rho}(\mathbf{v}\cdot\nabla)\mathbf{v} - \nabla\left(\overline{P}+P\right) - \nabla\cdot\overline{\mathscr{D}} - (\overline{\rho}+\rho)g\mathbf{r} + 2\overline{\rho}(\mathbf{v}\times\boldsymbol{\Omega}) \tag{9}$$

and

$$\begin{aligned}\overline{\rho}\overline{T}\frac{\partial}{\partial t}\left(\overline{S}+S\right) = &-\overline{\rho}\overline{T}\mathbf{v}\cdot\nabla\left(\overline{S}+S\right) + \nabla\cdot\left[\kappa\overline{\rho}\overline{T}\nabla\left(\overline{S}+S\right)\right] + \kappa_{\mathrm{rad}}\overline{\rho}c_{\mathrm{P}}\nabla\left(\overline{T}+T\right)\\ &+ 2\nu\overline{\rho}\left[e_{ij}e_{ij} - 1/3(\nabla\cdot\mathbf{v})^2\right],\end{aligned} \tag{10}$$

where the components of the viscous stress tensor $\overline{\mathscr{D}}$ are given by:

$$\overline{\mathscr{D}}_{ij} = -2\overline{\rho}\nu\left[e_{ij} - 1/3(\nabla\cdot\mathbf{v})\delta_{ij}\right]. \tag{11}$$

Linearizing the equation of state gives

$$\frac{\rho}{\overline{\rho}} = \frac{P}{\overline{P}} - \frac{T}{\overline{T}} = \frac{P}{\gamma\overline{P}} - \frac{S}{c_{\mathrm{P}}}. \tag{12}$$

In order to cut the computational workload $\overline{P}$, $\overline{T}$ and $\overline{\rho}$ are recalculated periodically rather than at every timestep, thereby reducing the implicit solve overhead (see Section 2.5). This is justified since these spherically averaged mean quantities evolve on timescales much longer than the corresponding perturbed quantities. $\overline{\rho}$ and $\overline{P}$ are updated using the spherically averaged parts of the perturbations ρ and P, and $\overline{T}$ is obtained from the zeroth-order equation of state,

$$\overline{P} = \mathscr{R}\overline{\rho}\overline{T}. \tag{13}$$

2.2. The streamfunction formalism

The system of equation (8)–(12) is solved numerically by first expressing the mass flux in terms of poloidal and toroidal streamfunctions,

$$\overline{\rho}\mathbf{v} = \nabla \times \nabla \times (W\mathbf{r}) + \nabla \times (Z\mathbf{r}). \tag{14}$$

The anelastic continuity equation, Eq. (8), is then satisfied by construction. Substituting Eq. (14) into the energy and momentum equations, and using the equation of state, Eq. (12), to eliminate ρ and T in favour of S and P, reduces the system to four equations (the energy equation and the three components of the momentum equation) in four unknowns (S, P, W and Z). We obtain a more manageable set of four equations by using the energy equation together with the radial component and horizontal divergence of the momentum equation to give three coupled equations for S, W and P, together with the radial component of the curl of the momentum equation to give an equation for Z which is coupled only nonlinearly and via the Coriolis term with the others. These equations are given explicitly in Appendix A, using a spherical polar coordinate system aligned with the angular velocity vector $\mathbf{\Omega}$ of the rotating frame.

2.3. Spatial discretization

Each of the four fields is expanded in spherical harmonics. For example, the entropy is written as

$$S(r,\theta_i,\phi_j,t) = \sum_{m=-m_{\max}}^{m_{\max}} \sum_{l=|m|}^{l_{\max}(m)} S_l^m(r,t) Y_{lm}(\theta_i,\phi_j), \tag{15}$$

where θ_i and ϕ_j represent the ith latitudinal and jth azimuthal grid points, respectively, and $Y_{lm}(\theta,\phi)$ is the spherical harmonic of degree l and order m, which is an eigenfunction of the horizontal Laplacian:

$$\nabla_\perp^2 Y_{lm}(\theta,\phi) = -\frac{l(l+1)}{r^2} Y_{lm} = -\frac{L}{r^2} Y_{lm}, \tag{16}$$

where $L = l(l+1)$. There is some freedom in choosing $l_{\max}$ in Eq. (15). The two most popular choices of truncation are triangular, $l_{\max}(m) = l_{\max} = m_{\max}$ and rhomboidal, $l_{\max}(m) = |m| + m_{\max}$. We choose the former since it is invariant to rotation and provides uniform spatial resolution over the entire sphere [1].

Gaussian quadrature techniques are used to maximize the accuracy of the inverse transforms. The N_θ grid points in the θ direction and the N_ϕ grid points in the ϕ direction are placed at the corresponding Gaussian abscissae. This yields a uniform grid in ϕ, along with a θ grid in which the θ_i are the zeros of the Legendre polynomial of degree N_θ. The density of the latitudinal grid points according to this prescription increases slightly towards the poles, but does not differ greatly from a uniform grid. The number of spatial grid points is chosen to be sufficient to resolve the quadratic nonlinear terms (such as the velocity advection terms), which are computed in physical space and then transformed back to spectral space. To avoid aliasing errors, we require

$$N_\theta \geqslant \frac{3l_{\max}+1}{2} \tag{17}$$

and

$$N_\phi \geqslant 3m_{\max}+1. \tag{18}$$

Derivatives in the θ and ϕ directions may easily be accomplished in spectral space by making use of the appropriate Legendre and Fourier recursion relations, respectively.

Radial derivatives are computed by expanding the spherical harmonic coefficients (e.g. $S_l^m(r,t)$) in terms of Chebyshev polynomials (chosen for their high degree of accuracy over the whole range of approximation), applying the appropriate recursion relations, and then transforming back to physical space. For reasons of numerical accuracy and computational simplicity, the grid points are chosen to be the appropriate Gaussian abscissae. This yields a higher density of points near the ends of the range, and fewer near the middle. Since the problems under investigation are expected to exhibit small-scale structure near the interface between the stable and unstable regions, it is desirable to have good radial resolution in this region. For this reason two stacked Chebyshev domains are used, having an interface at or near the boundary between the stable and unstable regions. The upper and lower domains have N_1 and N_2 grid points, respectively, with the total number of radial grid points being N_r. Denoting the inner and outer radii by r_i and r_o, respectively, and the interface radius by r_a, the appropriate expansion for $S_l^m(r,t)$ is

$$S_l^m(r_k,t) = \begin{cases} \frac{2}{N_1-1} \sum_{n=1}^{\prime\prime N_1} S_{ln}^m(t) T_n(r_k) & 1 \leqslant k \leqslant N_1, \\ \frac{2}{N_2-1} \sum_{n=N_1+1}^{\prime\prime N_r} S_{ln}^m(t) T_n(r_k) & N_1+1 \leqslant k \leqslant N_r, \end{cases} \tag{19}$$

where

$$r_k = \begin{cases} \frac{1}{2}\left[r_o + r_a + (r_o - r_a)\cos\left(\frac{k-1}{N_1-1}\pi\right)\right] & 1 \leqslant k \leqslant N_1, \\ \frac{1}{2}\left[r_a + r_i + (r_a - r_i)\cos\left(\frac{k-N_1-1}{N_2-1}\pi\right)\right] & N_1+1 \leqslant k \leqslant N_r, \end{cases} \tag{20}$$

are the Chebyshev collocation points, and

$$T_n(r_k) = \begin{cases} \cos\left[\frac{(k-1)(n-1)}{N_1-1}\pi\right] & 1 \leqslant k,n \leqslant N_1, \\ \cos\left[\frac{(k-N_1-1)(n-N_1-1)}{N_2-1}\pi\right] & N_1+1 \leqslant k,n \leqslant N_r \end{cases} \tag{21}$$

the Chebyshev polynomials evaluated at the corresponding points. The symbol $\sum''$ denotes a summation where the first and last terms in the series are multiplied by 1/2. Similar equations are used for the W, P and Z spherical harmonic coefficients. The forward transformation, which is easily derived from Eq. (19) using the orthogonality properties of cosines, is given by:

$$S_{ln}^m(t) = \begin{cases} \sum_{k=1}^{\prime\prime N_1} S_l^m(r_k,t) T_n(r_k) & 1 \leqslant n \leqslant N_1, \\ \sum_{k=N_1+1}^{\prime\prime N_r} S_l^m(r_k,t) T_n(r_k) & N_1+1 \leqslant n \leqslant N_r. \end{cases} \tag{22}$$

As mentioned above, the choice of grid points in Eq. (20) not only ensures high accuracy, but also permits the use of fast numerical algorithms, since the transformations in Eqs. (19) and (22) are simply discrete Fourier cosine transforms.

For the purposes of Section 2.4, Eq. (19) may be written as the matrix equation:

$$S_l^m(r_k,t) = \mathscr{T}_{kn} S_{ln}^m(t), \tag{23}$$

where the summation convention applies to the repeated index n. The elements of the matrix $\mathscr{T}$ may be read off from Eq. (19).

2.4. Time evolution

The evolution in time of the fields may be written:

$$\frac{\partial}{\partial t} S_l^m(r_k,t) = \mathscr{L}_{lmk}^S(t) + \mathscr{N}_{lmk}^S(t), \tag{24}$$

$$\frac{\partial}{\partial t} W_l^m(r_k,t) = \mathscr{L}_{lmk}^W(t) + \mathscr{N}_{lmk}^W(t), \tag{25}$$

$$\frac{\partial^2}{\partial t \partial r} W_l^m(r_k,t) = \mathscr{L}_{lmk}^P(t) + \mathscr{N}_{lmk}^P(t), \tag{26}$$

$$\frac{\partial}{\partial t} Z_l^m(r_k,t) = \mathscr{L}_{lmk}^Z(t) + \mathscr{N}_{lmk}^Z(t), \tag{27}$$

where $\mathscr{L}_{lmk}(t)$ denotes the linear terms and $\mathscr{N}_{lmk}(t)$ denotes the nonlinear terms in each equation. In the equations for W and Z, Coriolis forces, although linear, are included in the $\mathscr{N}_{lmk}(t)$ part of the right-hand sides because they involve a coupling between different l values. Since radial derivatives are involved in these equations, it is convenient to work with the Chebyshev-space representations of the spectral fields S, W, P and Z. Combining the first three of these equations into a single equation, and concatenating the Chebyshev-space representations of S, W and P into a single vector $\Psi_{ln}^m(t)$, such that

$$\Psi_{l1}^m \ldots \Psi_{l3N} = \left\{ S_{l1}^m \ldots S_{lN}^m, W_{l1}^m \ldots W_{lN}^m, P_{l1}^m \ldots P_{lN}^m \right\}, \tag{28}$$

we obtain:

$$\frac{\partial}{\partial t} \mathscr{T}_{kn}^{\Psi} \Psi_{ln}^m(t) = \mathscr{L}_{lmk}^{\Psi}(t) + \mathscr{N}_{lmk}^{\Psi}(t), \tag{29}$$

where

$$\mathscr{T}^{\Psi} = \begin{pmatrix} \mathscr{T} & 0 & 0 \\ 0 & \mathscr{T} & 0 \\ 0 & \mathscr{T}' & 0 \end{pmatrix} \tag{30}$$

and $\mathscr{L}_{lmk}^{\Psi}(t)$ and $\mathscr{N}_{lmk}^{\Psi}(t)$ express the concatenated S, W and P linear and nonlinear terms, respectively. Introducing another matrix $\mathscr{K}$, the linear terms may be expressed in terms of $\Psi_{ln}^m(t)$ as follows:

$$\mathscr{L}_{lmk}^{\Psi}(t) = \mathscr{K}_{l;kn}^{\Psi} \Psi_{ln}^m(t), \tag{31}$$

where summation is implied with respect to the n index, but *not* with respect to the l index. This is valid since $\mathscr{L}^{\Psi}_{lmk}(t)$ does not include terms which couple different l values, such as the Coriolis and advection terms. Eq. (29) thus becomes

$$\frac{\partial}{\partial t}\mathscr{T}^{\Psi}_{kn}\Psi^{m}_{ln}(t) = \mathscr{K}^{\Psi}_{l;kn}\Psi^{m}_{ln}(t) + \mathscr{N}^{\Psi}_{lmk}(t). \tag{32}$$

Rewriting Eq. (27) in a similar fashion gives the equation

$$\frac{\partial}{\partial t}\mathscr{T}_{kn}Z^{m}_{ln}(t) = \mathscr{K}^{Z}_{l;kn}Z^{m}_{ln}(t) + \mathscr{N}^{Z}_{lmk}(t). \tag{33}$$

2.5. Time discretization

Discretization in time is achieved by using an implicit Crank–Nicholson method for the linear terms and an explicit Adams–Bashforth scheme for the nonlinear terms. Eq. (32) may be discretized as follows:

$$\mathscr{T}^{\Psi}_{kn}\left[\frac{\Psi^{m}_{ln}(t_{\sigma+1}) - \Psi^{m}_{ln}(t_{\sigma})}{\Delta t}\right] = \mathscr{K}^{\Psi}_{l;kn}\left[\alpha\Psi^{m}_{ln}(t_{\sigma+1}) + (1-\alpha)\Psi^{m}_{ln}(t_{\sigma})\right] + \mathscr{N}^{\Psi}_{lmk}(t_{\sigma}) + \frac{1}{2}\frac{\Delta t}{\Delta t_{\text{old}}}\left[\mathscr{N}^{\Psi}_{lmk}(t_{\sigma}) - \mathscr{N}^{\Psi}_{lmk}(t_{\sigma-1})\right], \tag{34}$$

where $\Delta t = t_{\sigma+1} - t_{\sigma}$, $\Delta t_{\text{old}} = t_{\sigma} - t_{\sigma-1}$, and α is a parameter which characterizes the Crank–Nicholson method. Rearranging this equation gives:

$$\left(\mathscr{T}^{\Psi}_{kn} - \alpha\Delta t\mathscr{K}^{\Psi}_{l;kn}\right)\Psi^{m}_{ln}(t_{\sigma+1}) = \left(\mathscr{T}^{\Psi}_{kn} + (1-\alpha)\Delta t\mathscr{K}^{\Psi}_{l;kn}\right)\Psi^{m}_{ln}(t_{\sigma}) + \Delta t\mathscr{N}^{\Psi}_{lmk}(t_{\sigma}) + \frac{1}{2}\frac{\Delta t^{2}}{\Delta t_{\text{old}}}\left[\mathscr{N}^{\Psi}_{lmk}(t_{\sigma}) - \mathscr{N}^{\Psi}_{lmk}(t_{\sigma-1})\right]. \tag{35}$$

Rewriting the left-hand side in terms of a new matrix $\mathscr{A}$ defined by:

$$\mathscr{A}^{\Psi}_{l;kn} = \mathscr{T}^{\Psi}_{kn} - \alpha\Delta t\mathscr{K}^{\Psi}_{l;kn} \tag{36}$$

and rewriting the right-hand side as a single vector $\mathscr{B}^{\Psi}_{lmk}$ gives

$$\mathscr{A}^{\Psi}_{l;kn}\Psi^{m}_{ln}(t_{\sigma+1}) = \mathscr{B}^{\Psi}_{lmk}. \tag{37}$$

The corresponding equation for Z is

$$\mathscr{A}^{Z}_{l;kn}Z^{m}_{ln}(t_{\sigma+1}) = \mathscr{B}^{Z}_{lmk}. \tag{38}$$

A complete iteration consists essentially of two steps. The first is the computation of the right-hand sides of Eqs. (37) and (38), $\mathscr{B}^{\Psi}$ and $\mathscr{B}^{Z}$. The stored Adams–Bashforth terms from the previous step ($\mathscr{N}^{\Psi}_{lmk}(t_{\sigma-1})$ and $\mathscr{N}^{Z}_{lmk}(t_{\sigma-1})$) form part of these terms, so only the new Adams–Bashforth terms and the Crank–Nicholson term need be calculated. This is done by evaluating the required angular and radial derivatives in spectral and Chebyshev space, respectively (as described in Section 3), and transforming to physical space. The appropriate multiplications are performed

to calculate $\mathscr{B}^{\Psi}$ and $\mathscr{B}^{Z}$ in physical space, after which a transformation back to spectral space gives the required quantities.

The second step, which is to solve the matrix Eqs. (37) and (38), is referred to as the *implicit solve* step. It is carried out by first LU decomposing the matrices $\mathscr{A}^{\Psi}$ and $\mathscr{A}^{Z}$, and subsequently forming the solution for each separate right-hand side. The LU decomposition need only be performed when the matrices change, such as when the timestep is changed or the mean thermodynamic variables are updated. Finally the solution is transformed back from Chebyshev space to physical space in the radial dimension.

2.6. Boundary and continuity conditions

The coupled equations for S, W and P are sixth order in r and therefore require six boundary conditions, while the equation for Z is second order and requires another two boundary conditions. These are implemented in Chebyshev space in rows $k = 1, N_r, N_r+1, 2N_r, 2N_r+1$ and $3N_r$ of matrix Eq. (37) and rows $k = 1$ and N_r of matrix Eq. (38). The entropy boundary conditions comprise a fixed entropy at the outer boundary,

$$S_l^m(r_o, t) = 0 \tag{39}$$

and a fixed heat flux at the inner boundary,

$$-\kappa\overline{\rho}\,\overline{T}\,\frac{\partial(\bar{S}+S)}{\partial r}\bigg|_{r=r_i} - \kappa_{\mathrm{rad}}\overline{\rho}c_P\,\frac{\partial(\bar{T}+T)}{\partial r}\bigg|_{r=r_i} = \frac{L_\odot}{4\pi r_i^2}, \tag{40}$$

where $L_\odot$ is the solar luminosity. Since the inner and outer boundaries are both assumed to be impenetrable, W vanishes at both points, giving the next two boundary conditions to be applied in Eq. (37):

$$W_l^m(r_i, t) = W_l^m(r_o, t) = 0. \tag{41}$$

The final four boundary conditions, two of which are applied in Eq. (37) and two of which are applied in Eq. (38), arise from the assumption that the tangential stresses also vanish at the inner and outer boundaries:

$$\left[\frac{\partial^2 W_l^m}{\partial r^2}(r,t) - \left(\frac{2}{r} + \frac{\mathrm{d}\ln\overline{\rho}}{\mathrm{d}r}\right)\frac{\partial W_l^m}{\partial r}(r,t)\right]_{r=r_i,r_o} = 0, \tag{42}$$

$$\left[\frac{\partial Z_l^m}{\partial r}(r,t) - \left(\frac{2}{r} + \frac{\mathrm{d}\ln\overline{\rho}}{\mathrm{d}r}\right)Z_l^m(r,t)\right]_{r=r_i,r_o} = 0. \tag{43}$$

In addition to the above boundary conditions, continuity conditions must be applied at the interface between the two Chebyshev domains. As before, since Eqs. (37) and (38) are 6th and 2nd order, respectively, a total of eight continuity conditions can be applied. They are implemented in rows $k = N_1, N_1+1, N_r+N_1, N_r+N_1+1, 2N_r+N_1$ and $2N_r+N_1+1$ of Eq. (37) and rows $k = N_1$ and N_1+1 of Eq. (38), and apply to the quantities W, $\partial W/\partial r$, P, $\partial^2 W/\partial r^2$, S, $\partial S/\partial r$, Z,

and $\partial Z/\partial r$, respectively. These conditions ensure the continuity of all thermodynamic and velocity variables, as well as fluid stresses and the diffusive eddy heat flux.

3. Implementation

3.1. General considerations

The ASH code is written in FORTRAN 90, taking advantage of the modular framework, flexible data structures, dynamic memory allocation, and user-defined procedural interfaces, as well as the wide availability of efficient compilers. Parallelization is achieved using the message-passing strategy, implemented on the Cray T3E with the SHMEM library, and on the IBM SP-2 with the MPI library.

In a parallel-computing environment, two issues can greatly affect the execution speed of a particular code. First, it is most efficient to apply spectral transformations to data local to each processor, with inter-processor transposes being performed when necessary to arrange for the transformation dimension to be local. This approach minimizes the communication volume and as a result becomes most efficient as the problem size and the number of processors in increased [4]. Secondly, the load on the nodes should be balanced so that no single node has an unduly large load and presents a bottleneck. This section describes in detail how these two issues are addressed.

Suppose that the total number of nodes being used is N_{CPU}. These are divided into N_{IO} nodes which deal with input and output, and $N_{\mathrm{ANG}}N_{\mathrm{RAD}}$ computing nodes. N_{ANG} represents the number of nodes allocated to each spherical shell, while N_{RAD} represents the number of nodes allocated to each radial line. For consistency,

$$N_{\mathrm{IO}} + N_{\mathrm{ANG}}N_{\mathrm{RAD}} = N_{\mathrm{CPU}}. \tag{44}$$

3.2. Primary configuration

The *primary configuration* is the principal data-storage arrangement used in the code. The four fields (W, P, S, Z) are stored in (r, l, m) space, with the m values distributed over N_{ANG} angular nodes, and the r values distributed over N_{RAD} radial nodes. For any particular (m, r) pair, all the corresponding l values are dealt with by the same node; l is therefore said to be *in processor*. The distribution of the m and r values is determined by the requirements of load balancing.

Successful load balancing between separate nodes in a parallel machine is one of the most important aspects of producing an efficient code. For problems in which the data lies on a rectangular grid, such balancing is relatively straightforward to achieve. In the case of the ASH code, where the grid is effectively triangular, load balancing is more difficult. The m values $0, \ldots, l_{\max}$ are distributed over N_{ANG} nodes. The strategy employed is to divide the m values $0, \ldots, 2N_{\mathrm{P}}N_{\mathrm{ANG}} - 1$, where N_{P} is the largest integer satisfying $2N_{\mathrm{P}}N_{\mathrm{ANG}} - 1 \leqslant l_{\max}$, into the balanced pairs (0, $2N_{\mathrm{P}}N_{\mathrm{ANG}} - 1$), (1, $2N_{\mathrm{P}}N_{\mathrm{ANG}} - 2$), …, ($N_{\mathrm{P}}N_{\mathrm{ANG}} - 1$, $N_{\mathrm{P}}N_{\mathrm{ANG}}$). Each node is then responsible for N_{P} of these pairs. The remaining (up to $2N_{\mathrm{ANG}} - 1$) m values are

divided among the same nodes such that each node is responsible for up to two additional m values. In the radial direction, the r values are simply divided evenly among the N_{RAD} radial nodes.

An example of the primary configuration is shown in Fig. 1(a). In this case, $N_{\mathrm{RAD}} = 2$, $N_{\mathrm{ANG}} = 5$ and $l_{\max} = 11$. The 10 computing nodes are numbered from 0 to 9. There are five balanced pairs of m values, with two remaining ($l_{\max}$ and $l_{\max} - 1$).

3.3. Implicit configuration

The implicit-solve matrices, $\mathscr{A}^{\Psi}$ and $\mathscr{A}^{Z}$, depend on l, but are independent of m. It is therefore expedient to have all the m values for a particular l value in processor at the same time when performing the implicit solve. Additionally, all the r values must simultaneously be in processor. A different configuration is therefore adopted in performing the implicit solve; it is illustrated graphically in Fig. 1(b). A very similar procedure to that described in the previous section is adopted in dividing the l values $0, \ldots, l_{\max}$ between the $N_{\mathrm{RAD}}N_{\mathrm{ANG}}$ computing nodes. If $N_{\mathrm{RAD}} > 1$, then the division is less evenly balanced since the same load is divided between a larger number of nodes; indeed, if $N_{\mathrm{RAD}}N_{\mathrm{ANG}} > l_{\max} + 1$ then some computing nodes will not be assigned any l values in this configuration. However, since the implicit solve represents only a relatively small fraction of the total operation count, such an imbalance is not particularly important.

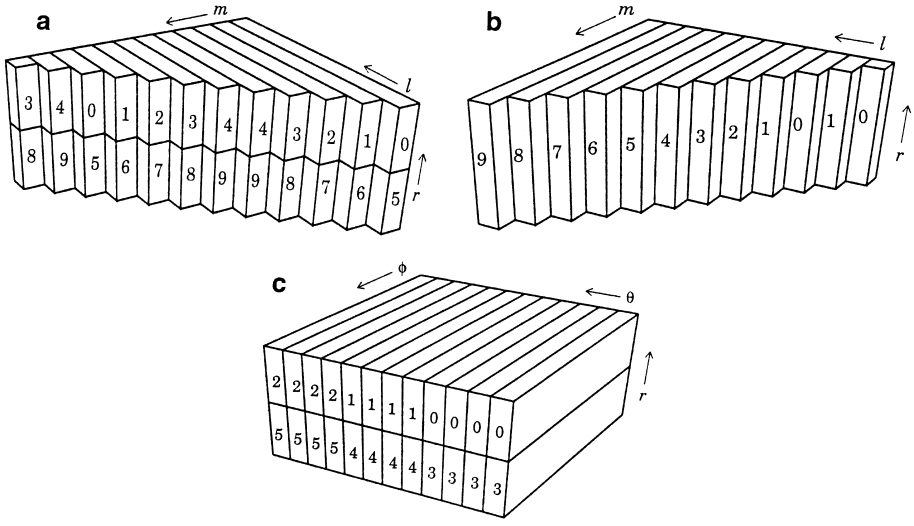

Fig. 1. (a) Primary configuration (l in processor, m and r distributed). The numbers correspond to the computing nodes $0, \ldots, 9$; in this case $N_{\mathrm{RAD}} = 2$, $N_{\mathrm{ANG}} = 5$ and $l_{\max} = 11$. (b) Corresponding implicit-solve configuration (m and r in processor, l distributed). (c) Physical-space configuration (ϕ in processor, θ and r distributed) for $N_{\mathrm{RAD}} = 2$, $N_{\mathrm{ANG}} = 3$ and $N_{\theta} = 12$.

3.4. Explicit terms

The explicit (Adams–Bashforth) terms form the right-hand sides ($\mathscr{B}^{\Psi}$ and $\mathscr{B}^{Z}$) of the implicit-solve equations. The multiplications required to compute these terms are performed in physical space, with the result being transformed back to the primary (spectral) configuration. The transformation from primary (spectral) configuration to physical space proceeds via three steps:

1. *Legendre transform*: In the primary configuration, l is in processor, so no transpose is required to perform this step.
2. *Transpose*: In order to perform the Fourier transform, m must be in processor, and a transpose is required to achieve this.
3. *Fast Fourier transform*: After this operation, the fields are in physical (r, θ, ϕ) space.

The reverse transformation from physical space back to spectral space proceeds by performing the inverse of the above operations in the reverse order.

The physical-space configuration is illustrated in Fig. 1(c) for the case $N_\theta = 12$, $N_{\mathrm{ANG}} = 3$, $N_{\mathrm{RAD}} = 2$. The distribution of θ and r values over computing nodes is straightforward in this case, owing to the rectangular grid.

Certain derivatives are required in calculating the explicit terms. Derivatives in the ϕ and θ directions are calculated in spectral space (using the appropriate recurrence relation in the case of θ derivatives), the results then being transformed to physical space. Radial derivatives are calculated by transforming to Chebyshev space, applying the appropriate recurrence relations, and then transforming back to physical space; if N_{RAD} is greater than 1, then this operation requires an additional transpose so that r is in processor for the Chebyshev transform.

4. Performance

4.1. Computing-node configuration

In this section the effect of the computing node configuration on the speed and load balancing of the ASH code is investigated. Sixteen computing nodes are used for this example, allowing five possible $N_{\mathrm{RAD}} \times N_{\mathrm{ANG}}$ combinations; in production running of larger problems, a considerably greater number of nodes is employed. Fig. 2(a) shows graphically the results for the case $N_\theta = 128$, with two stacked 33-point Chebyshev expansions in the radial direction; the five bars correspond to the five possible combinations of compute nodes. Each bar gives the time required for one iteration, and is subdivided into six pieces whose meanings are given in the key at the top of the plot; these pieces represent cumulative times and may include several repeats of an individual operation.

In the case of the Legendre transform, the unbalanced load (represented by the second bar up) is calculated as the difference between the longest time required by any one compute node and the time averaged over all the compute nodes (which is representative of the ideal load distribution, and is shown by the bottom bar). The

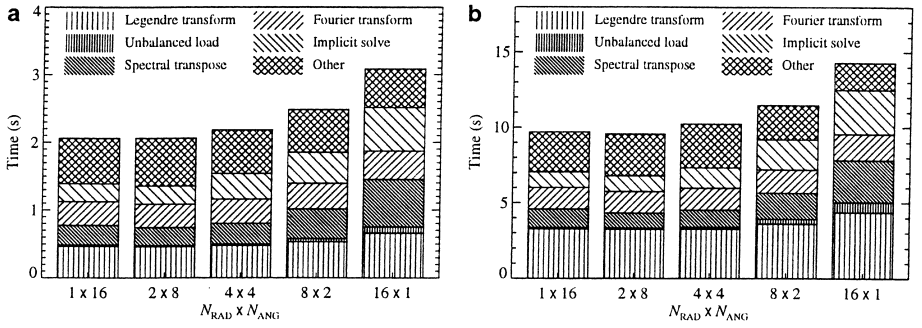

Fig. 2. (a) Timings for a single iteration with $N_r = 66$, $N_\theta = 128$, $N_\phi = 256$ on 16 compute nodes (300 MHz clock speed) of the T3E. (b) Corresponding timings for $N_\theta = 256$, $N_\phi = 512$.

effectiveness of the load-balancing strategy described in Section 2 is attested to by the small size of the unbalanced load relative to the total time for the Legendre transfor m. The minimum unbalanced load occurs for the $N_{\mathrm{RAD}} = 2$, $N_{\mathrm{ANG}} = 8$ case: for a larger value of N_{ANG}, the angular load-balancing strategy described in Section 2 does not produce a very even load distribution, while for a larger value of N_{RAD}, the radial division of the problem limits the load balancing. The variation of the total Legendre transform load with the configuration of the compute nodes is a consequence of the storage of arrays in memory – extreme values of N_{RAD} and N_{ANG} reduce the number of unit strides in accessing arrays, and therefore reduce the overall speed of the code.

The third bar up represents the total time spent performing spectral transposes (both forward and reverse). Since the number of operations involved in each spectral transpose increases with N_{RAD}, the size of this bar increases from the left to the right of the plot. The fourth bar up represents the time spent computing Fourier transforms; it stays roughly constant irrespective of the way the computing nodes are subdivided.

The fifth bar up corresponds to the time required for the implicit solve, including the forward and reverse implicit transposes. It varies with the distribution of the compute nodes principally through the variation of the speed of the implicit transposes – as N_{RAD} is increased, these take longer to execute, and the time for the implicit solve increases. The top bar includes the contribution to the loop time from all other sources not so far mentioned.

Fig. 2 shows that the choice of subdivision of the computing nodes can affect the execution speed of the ASH code by up to about 20%. For the case in question, the optimum speed is achieved with either $N_{\mathrm{RAD}} = 2$ or $N_{\mathrm{RAD}} = 4$.

In Fig. 2(b) N_θ is increased from 128 to 256 and N_ϕ is similarly doubled. The most noticeable effect is that the Legendre transform increasingly becomes the dominant contribution to the total loop time. This is because the time required for the Legendre transform increases like $N_r N_\theta^3$, a higher power of θ than for any other operation (e.g. the time for the Fourier transform only increases like $N_r N_\theta^2 \log N_\theta$). If the number of radial points were increased proportionally to N_θ, then the time required for the implicit solve would increase at the same rate as

that required for the Legendre transform (since the number of operations is proportional to $N_r^2 N_\theta^2$).

In summary, for increasing problem sizes with the number of radial points fixed, the time required for the Legendre transform is dominant in the limit (increasing faster than the time for any other operations), while for increasing problem sizes with $N_\theta \propto N_r$, the times for the implicit solve and Legendre transform dominate and increase proportionally (although typically the Legendre transform still requires considerably more time).

4.2. Scalability

One of the most important considerations affecting parallel codes is scalability, which describes the extent to which the total execution speed is proportional to the number of processing nodes. Preferably, the execution speed should increase linearly with the number of nodes, though in practice this is difficult to achieve. One problem is that it becomes increasingly difficult to achieve good load balancing for large numbers of nodes. Another is that as the number of nodes is increased, the communication between nodes becomes less efficient as the packets being transferred become smaller and smaller.

Fig. 3 shows the execution speed per node as a function of total number of nodes for a problem of size $N_r = 98$, $N_\theta = 256$, $N_\phi = 512$. It illustrates the good scalability of the ASH code for a large problem on the Cray T3E computer and shows the excellent performance of up to 120 Mflops per node. The total performance suffers by less than 20% as the number of nodes is increased from 16 to 128, which is a

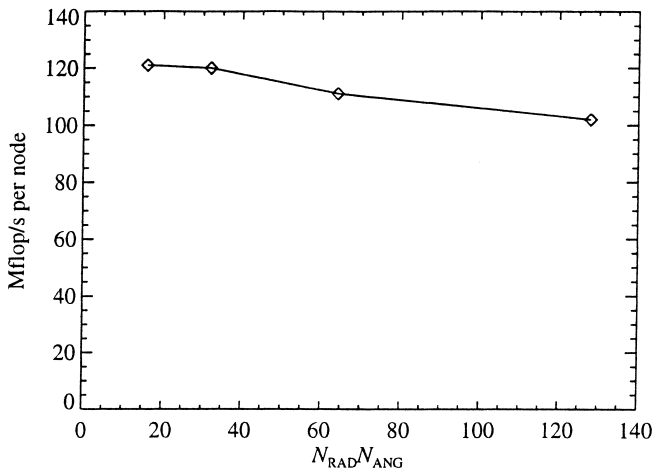

Fig. 3. Execution speed per node in Mflop/s (on 300 MHz T3E processors) for $N_r = 98$, $N_\theta = 256$, $N_\phi = 512$, plotted as a function of the number of compute nodes, $N_{\mathrm{RAD}}N_{\mathrm{ANG}}$.

promising result for a spectral code. In practice, when we move to larger machine configurations, we will increase the problem size rather than keep it constant as in Fig. 3, which will therefore decrease the load imbalance and the relative communication costs.

5. Properties of a solar convection solution

Fig. 4 shows an example of a solution obtained for convection in a spherical shell using the ASH code. Solar values are taken for the heat flux, rotation rate, mass and radius, while a solar model is used to provide initial values for $\overline{P}$, $\overline{\rho}$, and $\kappa_{\rm rad}$. The computational domain extends from 0.63 $R_\odot$ to 0.96 $R_\odot$, where $R_\odot$ is the solar radius, thereby including a region of stable stratification below the primary unstable zone in which effects of penetrative convection can also be studied. The shell currently has a density contrast across it of about 60, and thus compressibility effects are substantial. The viscosity, ν, and the entropy diffusivity, κ, are constant throughout the convecting region, while their ratio, the Prandtl number, is chosen to have a value of 0.1, which enhances the influence of intricate vortex dynamics. The spatial resolution is $N_r = 65 + 33$, $N_\theta = 512$, $N_\phi = 1024$, achieved using spherical harmonics up to degree $l_{\max} = 340$ with an imposed fourfold angular periodicity in longitude. The initial conditions are uniform rotation with no fluid motion, but with a small, random entropy perturbation. This perturbation initially grows linearly, but is ultimately limited by nonlinear feedback. When thermal equilibrium is established, the mid-layer Rayleigh number,

$$Ra = \frac{(\partial\rho/\partial S)_p \Delta S g d^3}{\overline{\rho}\nu\kappa} \tag{45}$$

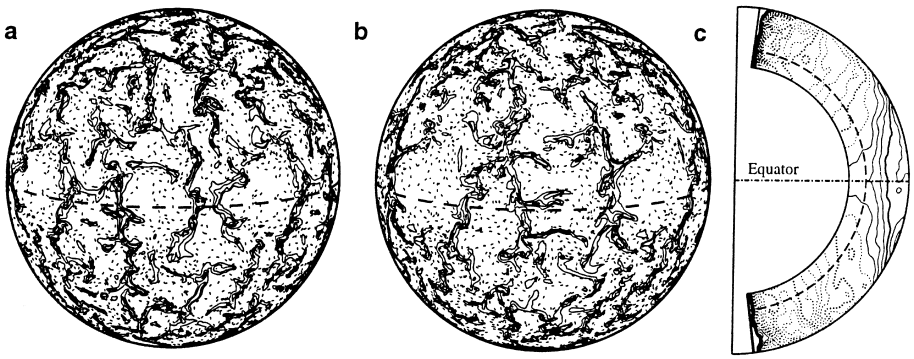

Fig. 4. Contours of radial velocity (with solid lines corresponding to downward flows) a quarter of the way into the unstable region at two instants in time (a, b) and contours of the resulting time-averaged differential rotation at 10 nHz intervals, with solid lines corresponding to prograde rotation relative to the rotating reference frame (c). The equator is shown dashed in (a) and (b); the dashed line in (c) shows the position of the interface between the stable and unstable regions.

and Taylor number,

$$Ta = \frac{4\Omega^2 d^4}{\nu^2}, \tag{46}$$

where d is the thickness of the convection zone and ΔS is the entropy change across the convection zone, are approximately 10^6 and 6×10^6, respectively. Fig. 4 shows contours of the radial velocity (with solid contours indicating downflows) at two instants in time about a quarter of the way into the unstable region, along with a contour plot of the rotation rate (averaged over about a month of solar time).

The radial velocity contour plots show little evidence of the so-called banana cells (sheared convective rolls) seen in more laminar solutions. The convection is now dominated by intermittent plumes of upflow and stronger downflow, some possessing a distinctive cyclonic swirl, although there is still some semblance of north–south alignment of structures in the equatorial region. The contours of rotation rate show that the angular velocity is nearly constant on radial lines throughout much of the convection zone at mid latitudes, and that there is a systematic decrease of rotation rate with latitude in going from equator to the poles. Prior to this, most models yielded a fast polar vortex, in addition to angular velocity constant on cylinders aligned with the rotation axis. The sun's shear layer of angular velocity speed-up just below the surface at low to mid latitudes is not realized in our simulations, but that is a likely consequence of supergranulation effects which we are currently not capturing. Further, the narrow transition region from differential rotation to rigid body rotation at the base of the convection zone (known as the "tachocline") is not obtained within our simulations since they operate at Reynolds number where viscous coupling can imprint the local rotation rate of the unstable zone upon the stable region. Currently we are encouraged to see that the angular velocity profiles within the convection zone itself have shifted to patterns much closer to the helioseismic findings. Further, we can understand such behavior in terms of the modified mean Reynolds stresses established by the turbulent convection as contrasted to that of more laminar flows.

6. Conclusion

We have shown how careful attention to optimization issues, including the implementation of inter-processor transposes and sophisticated load balancing, enables the ASH code to make very effective use of massively parallel architectures, allowing convection in spherical shells to be investigated at higher resolutions than was previously possible. This in turn has enabled us to reach higher levels of turbulence, where transport properties are influenced by highly time-dependent and complex flows. We have found differential rotation which is substantially different from that found by previous studies, and which more closely resembles that of the sun.

We believe these preliminary results to be very encouraging. It is now our goal to explore the parameter space as far as computing resources allow, and increase our understanding of how complex, turbulent flows achieve organized angular momentum transport in such systems. We believe that despite the relative simplicity of

our numerical model, such studies will lead us to a much greater understanding of the processes which shape the angular momentum distribution in the solar convection zone.

Acknowledgements

This work is supported in part by the National Aeronautics and Space Administration through grants NAG5-2256, NAG5-3077 and NCCS5-157, and by the National Science Foundation through grant ECS-9217394. Computational facilities were provided by the Pittsburgh Supercomputing Center and the Cornell Theory Center through MetaCenter grant MCA93S005P.

Appendix A. The equations

The momentum equation is given by:

$$\frac{\partial \overline{\rho}\mathbf{v}}{\partial t} = -\overline{\rho}(\mathbf{v}\cdot\nabla)\mathbf{v} - \nabla\left(\overline{P}+P\right) - \nabla\cdot\overline{\mathscr{D}} - (\overline{\rho}+\rho)g\mathbf{r} + 2\overline{\rho}(\mathbf{v}\times\mathbf{\Omega}). \tag{A.1}$$

Taking the radial component of this equation leads to the evolution equation for W:

$$\begin{aligned}\frac{\partial W}{\partial t} = &-\frac{r^2}{L}\left[\frac{\partial\left(\overline{P}+P\right)}{\partial r} + (\overline{\rho}+\rho)g\right] + \nu\left[-\frac{4}{3}\left(\alpha\beta + \frac{\mathrm{d}\beta}{\mathrm{d}r} + \frac{3\alpha+\beta}{r} + \frac{L}{r^2}\right)W\right.\\ &\left.+\frac{6\alpha-\beta}{3}\frac{\partial W}{\partial r} + \frac{\partial^2 W}{\partial r^2}\right] - \frac{\overline{\rho}r}{L}\left[rv_r\frac{\partial v_r}{\partial r} + v_\theta\frac{\partial v_r}{\partial\theta} + \frac{v_\phi}{\sin\theta}\frac{\partial v_r}{\partial\phi} - v_\theta^2 - v_\phi^2\right]\\ &+2\frac{\Omega r}{L}\left[-\sin\theta\frac{\partial Z}{\partial\theta} + \frac{\partial^2 W}{\partial r\partial\theta}\right],\end{aligned} \tag{A.2}$$

where $\alpha = \partial\ln\nu/\partial r$, $\beta = \partial\ln\rho/\partial r$ and $\Omega = |\mathbf{\Omega}|$. Taking the radial component of the curl of the momentum equation leads to the evolution equation for Z:

$$\begin{aligned}\frac{\partial Z}{\partial t} = &\nu\left[\left(-2\frac{\alpha+\beta}{r} - \beta\alpha - \frac{\mathrm{d}\beta}{\mathrm{d}r} - \frac{L}{r^2}\right)Z + (\alpha-\beta)\frac{\partial Z}{\partial r} + \frac{\partial^2 Z}{\partial r^2}\right]\\ &-\frac{\overline{\rho}}{L\sin\theta}\left\{\frac{\partial}{\partial\theta}\left[v_r\sin\theta\frac{\partial(rv_\phi)}{\partial r} + v_\phi\frac{\partial v_\phi}{\partial\phi} + \frac{\partial}{\partial\theta}\left(\sin\theta v_\phi\right)\right]\right.\\ &\left.-\frac{\partial}{\partial\phi}\left[v_r\frac{\partial(rv_\theta)}{\partial r} + v_\theta\frac{\partial v_\theta}{\partial\theta} + \frac{v_\phi}{\sin\theta}\frac{\partial v_\theta}{\partial\phi} - \frac{\cos\theta}{\sin\theta}v_\phi^2\right]\right\}\\ &+\frac{2\Omega}{L}\left[-\frac{L}{r\sin\theta}\frac{\partial}{\partial\theta}\left(W\sin^2\theta\right) + \left(L\cos\theta + \sin\theta\frac{\partial}{\partial\theta}\right)\frac{\partial W}{\partial r} + \frac{\partial Z}{\partial\phi}\right].\end{aligned} \tag{A.3}$$

Finally, taking the horizontal divergence of the momentum equation leads to the equation for P:

$$
\begin{aligned}
-\frac{\partial}{\partial t}\left(\frac{\partial W}{\partial r}\right) = P + \nu\Bigg[&\left(\alpha+\frac{2}{r}+\frac{2\beta}{3}\right)\frac{LW}{r^2}+\left(\alpha\beta+\frac{\mathrm{d}\beta}{\mathrm{d}r}+2\frac{\alpha+\beta}{r}+\frac{L}{r^2}\right)\frac{\partial W}{\partial r}\\
&+(\beta-\alpha)\frac{\partial^2 W}{\partial r^2}-\frac{\partial^3 W}{\partial r^3}\Bigg]\\
&-\frac{\overline{\rho}}{L\sin\theta}\Bigg\{\frac{\partial}{\partial\theta}\left[v_r\sin\theta\frac{\partial(rv_\theta)}{\partial r}+v_\theta\sin\theta\frac{\partial v_\theta}{\partial\theta}+v_\phi\frac{\partial v_\theta}{\partial\phi}-\cos\theta v_\phi^2\right]\\
&+\frac{\partial}{\partial\phi}\left[v_r\frac{\partial(rv_\phi)}{\partial r}+\frac{v_\theta}{\sin\theta}\frac{\partial}{\partial\theta}\left(\sin\theta v_\phi\right)+\frac{v_\phi}{\sin\theta}\frac{\partial v_\phi}{\partial\phi}\right]\Bigg\}\\
&+\frac{2\Omega}{L}\left[\left(L\cos\theta+\sin\theta\frac{\partial}{\partial\theta}\right)Z+\left(\frac{\partial^2 W}{\partial r\partial\phi}+\frac{L}{r}\frac{\partial W}{\partial\phi}\right)\right].
\end{aligned}
\tag{A.4}
$$

The energy equation is given by:

$$
\begin{aligned}
\overline{\rho}\overline{T}\frac{\partial\left(\overline{S}+S\right)}{\partial t} = &-\overline{\rho}\overline{T}\mathbf{v}\cdot\nabla\left(\overline{S}+S\right)+\nabla\cdot\left[\kappa\overline{\rho}\overline{T}\nabla\left(\overline{S}+S\right)+\kappa_{\mathrm{rad}}\overline{\rho}c_{\mathrm{P}}\nabla\left(\overline{T}+T\right)\right]\\
&+2\nu\overline{\rho}\left[e_{ij}e_{ij}-1/3(\nabla\cdot\mathbf{v})^2\right].
\end{aligned}
\tag{A.5}
$$

After some manipulation, this leads to

$$
\begin{aligned}
\frac{\partial S}{\partial t} = &-\frac{L}{r^2}\left(\kappa S+\frac{\kappa_{\mathrm{rad}}c_{\mathrm{P}}}{\overline{T}}T\right)+\kappa\left[\frac{\partial^2}{\partial r^2}+\frac{\partial\ln\left(r^2\kappa\overline{\rho}\overline{T}\right)}{\partial r}\frac{\partial}{\partial r}\right]\left(S+\overline{S}\right)\\
&+\frac{\kappa_{\mathrm{rad}}c_{\mathrm{P}}}{\overline{T}}\left[\frac{\partial^2}{\partial r^2}+\frac{\partial\ln\left(r^2\kappa_{\mathrm{rad}}\overline{\rho}c_{\mathrm{P}}\right)}{\partial r}\frac{\partial}{\partial r}\right]\left(T+\overline{T}\right)-v_r\frac{\partial}{\partial r}\left(S+\overline{S}\right)\\
&-\frac{v_\theta}{r}\frac{\partial S}{\partial\theta}-\frac{v_\phi}{r\sin\theta}\frac{\partial S}{\partial\phi}+\frac{\nu}{\overline{T}}\Bigg[2\left(\frac{\partial v_r}{\partial r}\right)^2+2\left(\frac{1}{r}\frac{\partial v_\theta}{\partial\theta}+\frac{v_r}{r}\right)^2\\
&-\frac{2}{3}(v_r\beta)^2+2\left(\frac{1}{r\sin\theta}\frac{\partial v_\phi}{\partial\phi}+\frac{v_r}{r}+\frac{\cos\theta}{r\sin\theta}v_\theta\right)^2+\left(\frac{\partial v_\theta}{\partial r}+\frac{1}{r}\frac{\partial v_r}{\partial\theta}-\frac{v_\theta}{r}\right)^2\\
&+\left(\frac{\partial v_\phi}{\partial r}+\frac{1}{r\sin\theta}\frac{\partial v_r}{\partial\phi}-\frac{v_\phi}{r}\right)^2+\left(\frac{1}{r}\frac{\partial v_\phi}{\partial\theta}+\frac{1}{r\sin\theta}\frac{\partial v_\theta}{\partial\phi}-\frac{\cos\theta}{r\sin\theta}v_\phi\right)^2\Bigg].
\end{aligned}
\tag{A.6}
$$

In all the above equations, the quadratic (advection and viscous heating) terms are written in terms of the three components of the velocity in spherical polar coordinates, while all the other terms are written in terms of W, Z, and S; this reflects how the terms are calculated in the ASH code.

References

[1] J.P. Boyd, Chebyshev and Fourier Spectral Methods Springer, Berlin, 1989.
[2] N.H. Brummell, F. Cattaneo, J. Toomre, Turbulent dynamics in the solar convection zone, Science 269 (1995) 1370–1379.
[3] N.H. Brummell, N.E. Hurlburt, J. Toomre, Turbulent compressible convection with rotation. I. Flow structure and evolution, Astrophys. J. 473 (1996) 494–513.
[4] I.T. Foster, P.H. Worley, Parallelizing the spectral transform method: A comparison of alternative parallel algorithms, SIAM J. Sci. Comput. 18 (1997) 806–837.
[5] P.A. Gilman, Linear simulations of Boussinesq convection in a deep rotating spherical shell, J. Atmos. Sci. 32 (1975) 1331–1352.
[6] P.A. Gilman, Nonlinear dynamics of Boussinesq convection in a deep rotating spherical shell, Geophys. Astrophys. Fluid Dyn. 8 (1977) 93–1357.
[7] P.A. Gilman, G.A. Glatzmaier, Compressible convection in a rotating spherical shell. I. Anelastic equations, ApJS 45 (1981) 335–349.
[8] P.A. Gilman, J. Miller, Nonlinear convection of a compressible fluid in a rotating spherical shell, ApJS 61 (1986) 585–608.
[9] G.A. Glatzmaier, Numerical simulations of stellar convective dynamos. I. The model and method, J. Comput. Phys. 55 (1984) 461–484.
[10] G.A. Glatzmaier, Numerical simulations of stellar convective dynamos. II. Field propagation in the convection zone, ApJ 291 (1985) 300–307.
[11] G.A. Glatzmaier, A review of what numerical simulations tell us about the internal rotation of the sun, in: B.R. Durney, S. Sofia (Eds.), The Internal Solar Angular Velocity, Reidel, Dordrecht, 1987, pp. 263–274.
[12] G.A. Glatzmaier, J. Toomre, Global-scale solar turbulent convection and its coupling to rotation, in: R.K. Ulrich, E.J. Rhodes, W. Däppen (Eds.), ASP Conf. Ser., vol. 76, GONG '94: Helio- and Astero-Seismology from Earth and Space ASP, San Francisco, 1995, pp. 200–203.
[13] R. Kippenhahn, A. Weigert, Stellar Structure and Evolution, Springer, Berlin, 1990.
[14] D.O. Gough, The anelastic approximation for thermal convection, J. Atmos. Sci. 26 (1969) 448–456.
[15] J. Latour, E.A. Spiegel, J. Toomre, J.-P. Zahn, Stellar convection theory I: the anelastic modal equations, ApJ 207 (1976) 233–243.
[16] M.J. Thompson, Differential rotation and dynamics of the solar interior, Science 272 (1996) 1300–1305.
[17] A. Tilgner, F.H. Busse, Finite amplitude convection in rotating spherical fluid shells, J. Fluid Mech. 332 (1997) 359–376.