INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2000; **32**: 773–797

A spectral solution of the magneto-convection equations in spherical geometry

Rainer Hollerbach*,[1]

Department of Mathematics, University of Glasgow, Glasgow, G12 8QW, U.K.

SUMMARY

A fully three-dimensional solution of the magneto-convection equations—the nonlinearly coupled momentum, induction and temperature equations—is presented in spherical geometry. Two very different methods for solving the momentum equation are presented, corresponding to the limits of slow and rapid rotation, and their relative advantages and disadvantages are discussed. The possibility of including a freely rotating, finitely conducting inner core in the solution of the momentum and induction equations is also discussed. Copyright © 2000 John Wiley & Sons, Ltd.

KEY WORDS: magneto-convection; spectral methods; spherical geometry

1. INTRODUCTION

The problem of convection in rotating spherical shells of various radius ratios is of fundamental importance in many different aspects of geophysical and astrophysical fluid dynamics. In particular, it is generally recognized that the magnetic fields not just of the Earth, but also of Mercury, Jupiter, Saturn, Uranus and Neptune, and even of some of Jupiter's moons, are generated by magnetohydrodynamic convection in their electrically conducting interiors [1–3]. On a more abstract level, the bifurcation sequences one may obtain in magneto-convection are also of considerable theoretical interest as mechanisms for symmetry breaking [4,5] and pattern selection [6]. Because of the highly nonlinear nature of the problem, much of the work is inevitably numerical. It is thus essential to develop robust numerical codes that can cope with a wide variety of parameter ranges. In this work one such code is described, focusing particular attention on the transition from slowly rotating to rapidly rotating systems.

In their simplest incompressible, Boussinesq form, the equations governing magnetohydrodynamic convection are

* Correspondence to: Department of Mathematics, University of Glasgow, Glasgow, G12 8QW, U.K. Tel.: +44 141 3306293; fax: +44 141 3304111.
[1] E-mail: rainer@maths.gla.ac.uk

Received January 1998
Revised June 1999

$$Ro\left(\frac{\partial}{\partial t}+\mathbf{U}\cdot\nabla\right)\mathbf{U}+2\hat{\mathbf{k}}\times\mathbf{U}=-\nabla p+E\nabla^2\mathbf{U}+(\nabla\times\mathbf{B})\times\mathbf{B}+q\tilde{R}a\Theta\mathbf{r} \tag{1}$$

$$\frac{\partial}{\partial t}\mathbf{B}=\nabla^2\mathbf{B}+\nabla\times(\mathbf{U}\times\mathbf{B}) \tag{2}$$

$$\left(\frac{\partial}{\partial t}+\mathbf{U}\cdot\nabla\right)\Theta=q\nabla^2\Theta \tag{3}$$

together with the solenoidal conditions $\nabla\cdot\mathbf{U}=\nabla\cdot\mathbf{B}=0$. Here, $\mathbf{U}$ is the fluid flow, $\mathbf{B}$ is the magnetic field, and Θ is the temperature.

These equations are to be solved in a spherical-shell geometry. The associated boundary conditions for the flow are typically either no-slip or stress-free at the inner and outer boundaries of the shell, and similarly for the temperature, either Θ or $\partial\Theta/\partial r$ are specified at the boundaries. The conditions on the field are a little more complicated, as the field is not confined within the shell, but extends into both the inner and outer regions—in the case of the Earth, the finitely conducting inner core and the insulating mantle. When the solution of the induction equation (2) is discussed in Section 3, it will be shown how to deal with both of these situations. Finally, in terms of initial conditions, a sufficiently robust code is developed that one can start with more or less any initial conditions and time step to some final equilibrated solution.

In these equations, length is scaled by the difference in inner and outer radii $L=r_\mathrm{o}-r_\mathrm{i}$, and time is then scaled by the magnetic diffusion time $T=L^2/\eta$. The fluid flow is scaled by $U=L/T$, so that the diffusive and advective time scales in the induction equation (2) are comparable. The magnetic field is scaled so that the Lorentz and Coriolis forces in the momentum equation (1) are comparable. These turn out to be the scalings appropriate for the generation of the Earth's magnetic field [7]. (However, they are clearly only appropriate if one is in fact in a rotating system; in a non-rotating system, where the Coriolis force $2\hat{\mathbf{k}}\times\mathbf{U}$ is absent, they would have to be modified suitably.)

Of the four non-dimensional parameters that then appear in these equations, the one we particularly wish to focus attention on is the magnetic Rossby number, $Ro=\eta/\Omega L^2$, measuring the ratio of the rotational time scale Ω^{-1} to the diffusive time scale L^2/η. It is at this point that we obtain the distinction between the slowly rotating regime, characterized by $Ro=O(1)$, and the rapidly rotating regime, characterized by $Ro\ll O(1)$. And we shall see that one can develop two quite different numerical solutions in these two regimes, each with its own advantages and disadvantages.

In order to solve these magneto-convection equations (1)–(3), we begin by decomposing as

$$\mathbf{U}=\nabla\times(e\hat{\mathbf{r}})+\nabla\times\nabla\times(f\hat{\mathbf{r}}),\qquad\mathbf{B}=\nabla\times(g\hat{\mathbf{r}})+\nabla\times\nabla\times(h\hat{\mathbf{r}}) \tag{4}$$

thereby automatically satisfying $\nabla\cdot\mathbf{U}=\mathbf{U}\cdot\mathbf{B}=0$. Conversely, any solenoidal vector can indeed be decomposed in this way [8]. We are thus left with the five scalars e, f, g, h and Θ, which we further expand in spherical harmonics as

$$e(r, \theta, \phi, t) = \sum_{m=0}^{\pm M_U} \sum_{l=m'}^{L_U} e_{lm}(r, t) P_l^{|m|}(\cos\theta)\, \mathrm{e}^{\mathrm{i}m\phi} \tag{5a}$$

$$g(r, \theta, \phi, t) = \sum_{m=0}^{\pm M_B} \sum_{l=m'}^{L_B} g_{lm}(r, t) P_l^{|m|}(\cos\theta)\, \mathrm{e}^{\mathrm{i}m\phi} \tag{5b}$$

$$\Theta(r, \theta, \phi, t) = \sum_{m=0}^{\pm M_T} \sum_{l=m}^{L_T} \Theta_{lm}(r, t) P_l^{|m|}(\cos\theta)\, \mathrm{e}^{\mathrm{i}m\phi} \tag{5c}$$

and similarly, f and h are expanded as e and g are respectively. In Equations (5a) and (5b), $m' = \max(m, 1)$, so the expansions for e, f, g and h exclude the $l = m = 0$ mode, the reason being that this mode contributes nothing when reinserted into the decompositions (4). It does, however, contribute to Θ, to which this decomposition does not apply, and so it must be included in (5c).

Note also that we explicitly allow for the possibility of having different truncations M_X and L_X for each of **U**, **B** and Θ. The reason is that these quantities may exhibit structures on quite different scales, and so we do not want to restrict ourselves by insisting on the same truncation for all three. For example, **U** will quite likely exhibit very fine structures when the Ekman number E is very small, whereas Θ will exhibit fine structures when the Roberts number q is small. So depending upon what parameters one is interested in, one may wish to have a particularly high truncation for one or the other, without having to increase the truncation for all.

Incidentally, one might also note that we used complex notation in Equation (5) purely for convenience of presentation; in virtually all of the actual code (see Section 2.2 for the one exception) we explicitly separate all quantities into coefficients of $\cos(m\phi)$ and $\sin(m\phi)$ instead. This makes the code somewhat longer, but because all calculations are then done in purely real arithmetic, it will actually run faster.

So, having expanded in spherical harmonics as in Equation (5), our task reduces to converting the original equations and boundary conditions for **U**, **B** and Θ into equations and boundary conditions for these functions e_{lm}, f_{lm}, etc., and then devising a numerical scheme to solve the latter. In the following sections we will show how that can be accomplished, starting with the momentum equation (1), for which we will devise two such schemes, corresponding to the limits of slow and rapid rotation.

2. THE MOMENTUM EQUATION

2.1. Slow rotation

We begin by taking the curl and the curl of the curl of Equation (1), thereby eliminating the pressure entirely. (Other methods of dealing with the pressure include explicitly solving for it along with the flow [9,10] or eliminating its effects by the use of suitably chosen test functions [11].) Using (5), the r components of the first and second curls become

$$\sum_{m,l} \frac{l(l+1)}{r^2}\left[Ro\,\frac{\partial}{\partial t} - EL_l\right] e_{lm}(r,t) P_l^{|m|}\, \mathrm{e}^{\mathrm{i}m\phi} = \hat{\mathbf{r}}\cdot\nabla\times\mathbf{F} \tag{6a}$$

$$-\sum_{m,l} \frac{l(l+1)}{r^2}\left[Ro\,\frac{\partial}{\partial t} - EL_l\right] L_l f_{lm}(r,t) P_l^{|m|}\, \mathrm{e}^{\mathrm{i}m\phi} = \hat{\mathbf{r}}\cdot\nabla\times\nabla\times\mathbf{F} \tag{6b}$$

where the operator

$$L_l = \frac{\partial^2}{\partial r^2} - \frac{l(l+1)}{r^2} \tag{7}$$

and where

$$\mathbf{F} = -2\hat{\mathbf{k}}\times\mathbf{U} - Ro(\mathbf{U}\cdot\nabla)\mathbf{U} + (\nabla\times\mathbf{B})\times\mathbf{B} + q\tilde{R}a\Theta\mathbf{r} \tag{8}$$

The slowly rotating regime is thus characterized by the fact that we can treat the Coriolis force explicitly (if it is present at all—slow rotation also includes no rotation).

In Section 5 we will briefly discuss how we can efficiently evaluate these forcing terms $\hat{\mathbf{r}}\cdot\nabla\times\mathbf{F}$ and $\hat{\mathbf{r}}\cdot\nabla\times\nabla\times\mathbf{F}$, and also separate them out into the different spherical harmonic components. For now, though, we note simply that, having separated them out, we can treat Equations (6a) and (6b) independently for each m and even for each l (and independently of one another, of course).

Starting with the simpler parabolic equation (6a), we further expand e_{lm} in Chebyshev polynomials as

$$e_{lm}(r,t) = \sum_{k=1}^{K_U+2} e_{klm}(t) T_{k-1}(x) \tag{9}$$

where

$$r = \frac{r_\mathrm{o}+r_\mathrm{i}}{2} + \frac{r_\mathrm{o}-r_\mathrm{i}}{2}\,x \tag{10}$$

determines x, the radial co-ordinate normalized to $(-1, 1)$ across the shell. We will then enforce (6a) at K_U collocation points, the K_U zeros of $T_{K_U}(x)$, which, together with two boundary conditions, gives K_U+2 conditions on these K_U+2 coefficients e_{klm}. Note, incidentally, how these collocation points automatically concentrate resolution close to the boundaries, where we may have to resolve various boundary layers. See, for example, Reference [12] for a general discussion of some of the advantageous features of Chebyshev and other spectral expansion methods.

To actually advance these coefficients from one time step to the next, we implement a second-order Runge–Kutta method, modified to treat the diffusive terms implicitly. We thus have the following two steps:

1. *The Predictor Step*
Given all the various spectral coefficients at time step n, evaluate the spectral coefficients of the forcing term $\hat{\mathbf{r}}\cdot\nabla\times\mathbf{F}$ at time step n. Hence evaluate the forcings at the collocation points x_j, $j=1, K_U$ and call them DE_j. Enforcing (6a) at the collocation points, we then have the equations

$$\sum_{k=1}^{K_U+2}\frac{l(l+1)}{r_j^2}\left[Ro(\tilde{e}_{klm}^{n+1}-e_{klm}^{n})-\frac{\Delta t}{2}EL_l(\tilde{e}_{klm}^{n+1}+e_{klm}^{n})\right]T_{k-1}(x)\Big|_{x=x_j}=\Delta t DE_j \tag{11a}$$

for $j=1,\ K_U$. The boundary conditions, discussed in more detail in Section 2.3, give us two more equations, so we can invert this system to obtain the preliminary coefficients $\tilde{e}_{klm}^{n+1}$.

2. *The Corrector Step*
Given this estimate of the coefficients at time step $n+1$, again evaluate the forcings at the collocation points, and call them DE_j'. To obtain an improved estimate of the coefficients at time step $n+1$, we then have the K_U equations

$$\sum_{k=1}^{K_U+2}\frac{l(l+1)}{r_j^2}\left[Ro(e_{klm}^{n+1}-e_{klm}^{n})-\frac{\Delta t}{2}EL_l(e_{klm}^{n+1}+e_{klm}^{n})\right]T_{k-1}(x)\Big|_{x=x_j}=\frac{\Delta t}{2}(DE_j'+DE_j) \tag{11b}$$

which, together with the same two boundary conditions, can again be inverted to obtain the final coefficients e_{klm}^{n+1}.

In matrix notation, equations (11) become

$$\mathbf{X}\tilde{\mathbf{e}}^{n+1}=\mathbf{Y}\mathbf{e}^{n}+\Delta t\mathbf{DE} \tag{12a}$$

$$\mathbf{X}\mathbf{e}^{n+1}=\mathbf{Y}\mathbf{e}^{n}+\frac{\Delta t}{2}(\mathbf{DE}'+\mathbf{DE}) \tag{12b}$$

where

$$\mathbf{e}^{n}=[e_{1,lm}^{n}, e_{2,lm}^{n}, \ldots, e_{K_U+2,lm}^{n}]^T \tag{13}$$

$$\mathbf{DE}=[DE_1, DE_2, \ldots, DE_{K_U}, 0, 0]^T \tag{14}$$

etc. The first K_U rows of matrices $\mathbf{X}$ and $\mathbf{Y}$ are given by

$$\mathbf{X}_{jk}=\frac{l(l+1)}{r_j^2}[Ro-0.5\Delta t EL_l]T_{k-1}(x)\big|_{x=x_j} \tag{15a}$$

$$\mathbf{Y}_{jk}=\frac{l(l+1)}{r_j^2}[Ro+0.5\Delta t EL_l]T_{k-1}(x)\big|_{x=x_j} \tag{15b}$$

The last two rows of $\mathbf{X}$ implement the boundary conditions, and the last two rows of $\mathbf{Y}$ are zero. We can then further precompute not just $\mathbf{X}^{-1}$, but also the product $\mathbf{X}^{-1}\mathbf{Y}$ (note how neither of these matrices $\mathbf{X}$ or $\mathbf{Y}$ varies from one time step to the next), and so Equations (12) represent a very efficient way of advancing (6a) in time.

Turning next to the more complicated parabolo-elliptic Equation (6b), we similarly expand f_{lm} in Chebyshev polynomials as

$$f_{lm}(r, t) = \sum_{k=1}^{K_U+4} f_{klm}(t) T_{k-1}(x) \tag{16}$$

We will then enforce (6b) at the same K_U collocation points as before, which, together with four boundary conditions now, again discussed in Section 2.3, gives us $K_U + 4$ conditions on these $K_U + 4$ coefficients f_{klm}.

Proceeding as before, the matrix system we then end up with is

$$\mathbf{X}\tilde{\mathbf{f}}^{n+1} = \mathbf{Y}\mathbf{f}^n + \Delta t \mathbf{DF} \tag{17a}$$

$$\mathbf{X}\mathbf{f}^{n+1} = \mathbf{Y}\mathbf{f}^n + \frac{\Delta t}{2}(\mathbf{DF}' + \mathbf{DF}) \tag{17b}$$

where again

$$\mathbf{f}^n = [f^n_{1,lm}, f^n_{2,lm}, \ldots, f^n_{K_U+4,lm}]^T \tag{18}$$

$$\mathbf{DF} = [DF_1, DF_2, \ldots, DF_{K_U}, 0, 0, 0, 0]^T \tag{19}$$

etc. The first K_U rows of matrices $\mathbf{X}$ and $\mathbf{Y}$ are now given by

$$\mathbf{X}_{jk} = -\frac{l(l+1)}{r_j^2}[Ro - 0.5\Delta t E L_l] L_l T_{k-1}(x)\big|_{x=x_j} \tag{20a}$$

$$\mathbf{Y}_{jk} = -\frac{l(l+1)}{r_j^2}[Ro + 0.5\Delta t E L_l] L_l T_{k-1}(x)\big|_{x=x_j} \tag{20b}$$

The last four rows of $\mathbf{X}$ implement the boundary conditions, and the last four rows of $\mathbf{Y}$ are zero.

In fact, if we implement this procedure (17), with $\mathbf{X}$ and $\mathbf{Y}$ given by (20), and attempt to use it to reproduce the (analytically derivable) free decay rates (corresponding to $\mathbf{F} \equiv 0$) with either no-slip or stress-free boundary conditions, we find that it only works for stress-free; for no-slip it turns out to be unstable, even for extremely small time steps. This instability comes about entirely because we insisted on time stepping (6b) as a single parabolo-elliptic equation. If we instead introduced a new variable $f'_{lm} = L_l f_{lm}$, we would have a parabolic equation for f'_{lm}, which could be time stepped just like (6a), together with an elliptic equation for f_{lm}, which could be inverted at each time step. The challenge with this procedure would be how to

implement the boundary conditions—the original four conditions do not convert nicely into two on f'_{lm}, and two on f_{lm}.

Although these difficulties could be overcome (and have been overcome in a corresponding plane-layer code [13]), we decided that before devising such a completely different procedure just for this one case, we would try various minor modifications of the existing one. In particular, we might try treating the diffusive terms more implicitly, i.e. instead of weighting the known and unknown coefficients equally with 0.5 as in (11), we could try weighting the unknown coefficients with 0.6, and the known with 0.4, say. After all, one of the reasons why we treated the diffusive terms implicitly in the first place was to improve the stability of the scheme, so making it more implicit might be expected to make it more stable. So, we merely change (20) to

$$\mathbf{X}_{jk} = -\frac{l(l+1)}{r_j^2}\left[Ro - 0.6\Delta t E L_l\right] L_l T_{k-1}(x)\big|_{x=x_j} \tag{21a}$$

$$\mathbf{Y}_{jk} = -\frac{l(l+1)}{r_j^2}\left[Ro + 0.4\Delta t E L_l\right] L_l T_{k-1}(x)\big|_{x=x_j} \tag{21b}$$

and just see if that works for no-slip boundary conditions. It is certainly such a trivial change to implement that it is worth a try! And indeed it turns out that it does work.

Of course, by changing the weighting in this way, we may have improved the stability, but we have also degraded the accuracy. That is, the other reason why we treated the diffusive terms implicitly is that with the weights at 0.5/0.5, this procedure is second-order accurate, whereas with any other weights it is only first-order accurate with respect to the diffusion terms. Tables I and II show how the numerically computed free decay rates with (20) for stress-free and (21) for no-slip boundary conditions compare with the analytically derived values, in a spherical shell of radius ratio 1:3, and with $Ro = E = 1$ (the Rossby and Ekman numbers can of course be scaled out when one is only considering free decay). The radial truncation has been set at a sufficiently high value ($K_U \approx 20$ is sufficient) that all of the error in Tables I and II is indeed from the discretization in time, and none from the discretization in space. And sure enough, one notes that for the stress-free case (Table I) this procedure is second-order accurate, that is, when Δt is reduced by a factor of 10 the error is reduced by a factor of 100, whereas for the no-slip case (Table II) it is only first-order accurate.

However, one notes that even for this first-order procedure one already has quite good accuracy, to within 1 per cent, even for $\Delta t = 10^{-3}$. And of course one must remember that in any real calculation, such as those in Section 2.5, the diffusive time scale is likely to be one of the longest of the various time scales in the problem. So if Δt is chosen to be short compared with the shortest time scale, as it must be, then it will typically be very short compared with the diffusive time scale, and one will therefore be in this regime where even this nominally first-order procedure is giving very good accuracy.

To summarize then, the procedure we use to solve the momentum equation in the limit of slow rotation is the following: We first make one call to the routine that evaluates the forcing terms and separates them out into the different spherical harmonic components. We then do all the matrix multiplications in (12a) and (17a), with the **X**s and **Y**s given by (15) and either

Table I. The analytically derived lowest free decay rates λ of (6b) with stress-free boundary conditions, and the relative error in the numerically computed decay rates $(\lambda_{\text{num}}-\lambda)/\lambda$ for $\Delta t = 10^{-3}$ and 10^{-4}.

	$l=1$	$l=2$	$l=3$	$l=4$
Analytic	12.0478955	15.4390734	20.9063004	28.2118123
$\Delta t = 10^{-3}$	0.0000121	0.0000199	0.0000364	0.0000663
$\Delta t = 10^{-4}$	0.0000001	0.0000002	0.0000004	0.0000007

Table II. The analytically derived lowest free decay rates λ of (6b) with no-slip boundary conditions, and the relative error in the numerically computed decay rates $(\lambda_{\text{num}}-\lambda)/\lambda$ for $\Delta t = 10^{-3}$, 10^{-4} and 10^{-5}.

	$l=1$	$l=2$	$l=3$	$l=4$
Analytic	37.6096886	36.5341303	38.3227479	43.3706793
$\Delta t = 10^{-3}$	−0.0036303	−0.0035301	−0.0036966	−0.0041636
$\Delta t = 10^{-4}$	−0.0003748	−0.0003641	−0.0003819	−0.0004320
$\Delta t = 10^{-5}$	−0.0000376	−0.0000365	−0.0000383	−0.0000434

(20) or (21), to obtain all the preliminary coefficients $\tilde{e}_{klm}^{n+1}$ and $\tilde{f}_{klm}^{n+1}$. We then make another call to the routine that evaluates the forcings, and finally we do the matrix multiplications in (12b) and (17b) to obtain the final coefficients e_{klm}^{n+1} and f_{klm}^{n+1}.

In terms of CPU usage, for each time step we thus have to do $O(L_U M_U)$ matrix multiplications, where each one involves $O(K_U^2)$ floating point operations, plus whatever computational effort is involved in the two calls to the forcing routine. This routine will in fact turn out to make up the dominant part of the total computational effort, and so this detailed estimate of the effort involved in these matrix multiplications was not really necessary. It does, however, serve to illustrate an important point, namely that up to now we have been slightly careless in our notation. That is, there are really far more than just two of these **X** (and **Y**) matrices, one for the es and one for the fs; if we have to do $O(L_U M_U)$ matrix multiplications, then clearly there also have to be $2L_U M_U$ of these matrices, one for the es and one for the fs all right, but separately for each m and each l. We agreed, after all, that procedures (12) and (17) whereby we would time step (6) were to apply separately to each m and l, and so all the various quantities that are used in these procedures must be defined separately for each m and l.

However, and this is the great advantage of this slow rotation method, if we look at how we defined these **X**s and **Y**s, we note that while they certainly do depend on l, they do not actually depend on m at all (we will see below that the rows that implement the boundary conditions also do not depend on m). That, incidentally, is also why we never needed to specify m in Tables I and II, only l. But if these matrices do not depend on m, we do not need to set up all $2L_U M_U$ of them, only $2L_U$, and then we just re-use the same matrix for all the different m values. Incidentally, one might also note that because all of these matrices are purely real, it is indeed very easy to split up the $\exp(im\phi)$ into $\cos(m\phi)$ and $\sin(m\phi)$ components instead,

and again just re-use the same matrix for both. We thus see that the memory requirements of this slow rotation method are relatively modest. In contrast, in the next section we will see that the memory requirements of the rapid rotation method are exorbitant, precisely because the corresponding matrices do depend on m as well, and so one cannot re-use them. For comparison, at a truncation of $K_U = 40$, $L_U = 60$, $M_U = 24$ one would need less than 50 Mbytes with the slow rotation method, but more than 500 Mbytes with the rapid rotation method.

2.2. Rapid rotation

Having devised this one method, one might wonder why we would want to devise another one as well, particularly when it will turn out to be so much more memory intensive. The reason is that this method we have developed so far also has a very considerable disadvantage, namely that it does not work very well for very small Rossby numbers. In particular, it is quite clear that with this method we will have to reduce the time step in line with Ro, so if $Ro = O(10^{-8})$—as it is in the Earth's core [7]—we would have to take an impossibly large number of time steps to cover even one magnetic diffusion time.

The difficulty of course is that if Ro and E are both small, we have two small terms on the left-hand side of (6), and various large terms on the right, which is not an ideal situation. If we could only move one of the large terms from the right-hand side to the left, it would be considerably improved, and we could almost certainly take substantially larger time steps. The most obvious candidate for moving from the right to the left is of course the Coriolis force. The rapidly rotating regime is thus defined by the fact that we must treat the Coriolis force implicitly.

To see how we might do that, we must first work out how the Coriolis force will appear when expressed in terms of the e_{lm} and f_{lm}. After a little algebra, one obtains as the analogue of (6),

$$\sum_{m,l}\left[\frac{l(l+1)}{r^2}\left(Ro\,\frac{\partial}{\partial t}-EL_l\right)-\frac{2\mathrm{i}m}{r^2}\right]e_{lm}(r,t)P_l^{|m|}(\cos\theta)\,\mathrm{e}^{\mathrm{i}m\phi}$$
$$+\frac{2}{r^2}\left(\frac{l(l+1)}{r}-\frac{\partial}{\partial r}\right)f_{lm}(r,t)\sin\theta\,\frac{\mathrm{d}}{\mathrm{d}\theta}P_l^{|m|}(\cos\theta)\,\mathrm{e}^{\mathrm{i}m\phi}$$
$$+2\,\frac{l(l+1)}{r^2}\left(\frac{2}{r}-\frac{\partial}{\partial r}\right)f_{lm}(r,t)\cos\theta\,P_l^{|m|}(\cos\theta)\,\mathrm{e}^{\mathrm{i}m\phi}=\hat{\mathbf{r}}\cdot\nabla\times\mathbf{F} \tag{22a}$$

$$\sum_{m,l}-\left[\frac{l(l+1)}{r^2}\left(Ro\,\frac{\partial}{\partial t}-EL_l\right)-\frac{2\mathrm{i}m}{r^2}\right]L_l f_{lm}(r,t)P_l^{|m|}(\cos\theta)\,\mathrm{e}^{\mathrm{i}m\phi}$$
$$+\frac{2}{r^2}\left(\frac{l(l+1)}{r}-\frac{\partial}{\partial r}\right)e_{lm}(r,t)\sin\theta\,\frac{\mathrm{d}}{\mathrm{d}\theta}P_l^{|m|}(\cos\theta)\,\mathrm{e}^{\mathrm{i}m\phi}$$
$$+2\,\frac{l(l+1)}{r^2}\left(\frac{2}{r}-\frac{\partial}{\partial r}\right)e_{lm}(r,t)\cos\theta\,P_l^{|m|}(\cos\theta)\,\mathrm{e}^{\mathrm{i}m\phi}=\hat{\mathbf{r}}\cdot\nabla\times\nabla\times\mathbf{F} \tag{22b}$$

where L_l is as before, but $\mathbf{F}$ no longer includes the Coriolis force.

And one sees immediately that (22) will be rather more difficult to solve than (6), because the different spherical harmonics no longer separate out quite so nicely. That is, we can certainly still treat each m independently, but not each l. Fortunately, even this newly introduced coupling in l turns out to be relatively simple; using the recursion relations [14]

$$\sin\theta \frac{\mathrm{d}}{\mathrm{d}\theta} P_l^{|m|}(\cos\theta) = \frac{l(l-|m|+1)}{2l+1} P_{l+1}^{|m|} - \frac{(l+|m|)(l+1)}{2l+1} P_{l-1}^{|m|} \tag{23a}$$

$$\cos\theta\, P_l^{|m|}(\cos\theta) = \frac{l-|m|+1}{2l+1} P_{l+1}^{|m|} + \frac{l+|m|}{2l+1} P_{l-1}^{|m|} \tag{23b}$$

we find from (22a) that e_{lm} couples only to $f_{l\pm1,m}$, and from (22b) that f_{lm} couples only to $e_{l\pm1,m}$. We thus find, first of all, that the l structure decouples into two symmetry classes, odd l for e_l and even l for f_l for the first, and vice versa for the second. We can then time step each symmetry class by the same procedure as (12) or (17), except that the number and size of the matrices involved is now very different. That is, instead of having $2L_U$ small **X** and **Y** matrices, applied separately to the different e_l and f_l, we will now have two large **X** and **Y** matrices, applied separately to the two different symmetry classes.

Rather than explicitly writing out how these matrices are defined, as in (15), (20) or (21), it is probably easiest at this point just to schematically illustrate their general structure, which is all that matters for the following discussion. Figure 1 shows this structure for the first symmetry class, for $m = 0$ or 1 (for which l starts at 1), for a truncation of $K_U = 5$, $L_U = 6$. One notes first of all that both **X** and **Y** have a block tri-diagonal structure, with the diagonal blocks being very similar to the previously defined **X**s and **Y**s. In particular, the first, third and fifth rows of the blocks are obtained by enforcing (22a) at the same K_U collocation points as

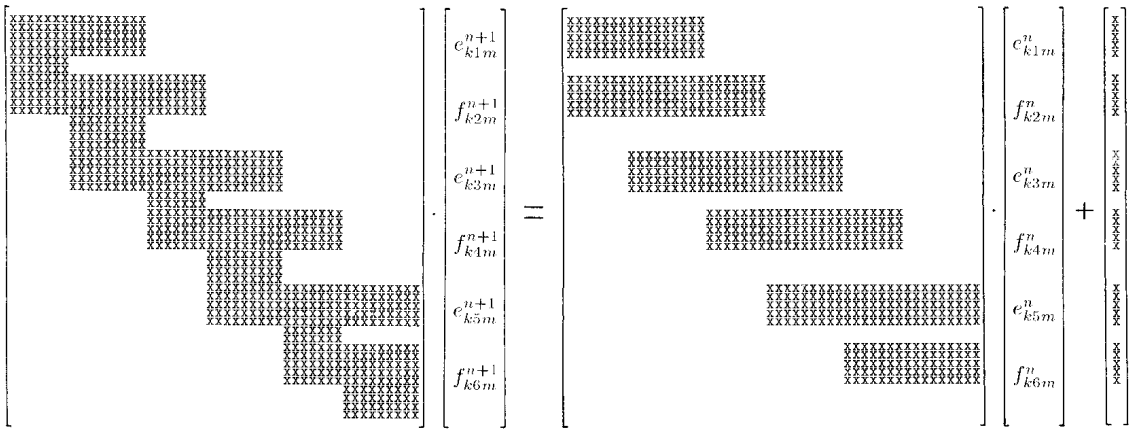

Figure 1. The block tri-diagonal structure of the first of the two matrix systems one must solve in the rapid rotation method. The entries e_{klm} and f_{klm} in these column vectors are of course themselves columns of all the ks.

before, and simply projecting out the $l = 1, 3, 5$ contributions, whereas the second, fourth and sixth rows of the blocks are obtained by enforcing (22b) at the collocation points, and projecting out the $l = 2, 4, 6$ contributions. That is, except for the additional $-2im/r^2$ terms, the first, third and fifth diagonal blocks are just like (15) for $l = 1, 3, 5$, whereas the second, fourth and sixth diagonal blocks are just like (20) or (21) for $l = 2, 4, 6$. And of course the off-diagonal blocks then implement the coupling to the $f_{l\pm1}$ terms in (22a) and the $e_{l\pm1}$ terms in (22b); the details of what goes into these blocks are tedious but straightforward. Note also that the boundary conditions are enforced exactly as before, by the last two rows in each diagonal block for the e_ls, and the last four rows in each diagonal block for the f_ls.

Having set up matrices $\mathbf{X}$ and $\mathbf{Y}$ for the two symmetry classes, the rest of the time stepping procedure is much the same as before, except that we no longer pre-compute the product $\mathbf{X}^{-1}\mathbf{Y}$, which would disrupt the banded structure of the matrix. Instead, we pre-compute the LU decomposition of $\mathbf{X}$, using for example the NAG routine F01NAF, and then at each time step we can still very efficiently invert the system using the routine F04NAF. Each call to this routine turns out to involve $O(L_U K_U^2)$ floating point operations, and remembering that we have to do $O(M_U)$ of them, we find that the CPU usage per time step is of the same order as it was for the slow rotation method. In fact, these two calls to the same forcing routine as before still make up the dominant part of the total computational effort, and so the computational effort per time step is essentially the same for the slow and rapid rotation methods. If this rapid rotation method therefore allows us to take time steps that are even just slightly larger, it could be quite advantageous indeed.

However, before congratulating ourselves on having devised such a vastly superior method, we should realize that there is also a very considerable price to be paid, in terms of the memory requirements. If we consider for the moment the memory needed for just a single azimuthal mode, we find it is already considerably greater than it was before, for two reasons. First, because of the presence of all of these off-diagonal blocks, and because the Numerical Algorithms Group (NAG) routines need a bit extra to do the LU decomposition accurately, the memory is already increased by a factor of roughly four. Second, because these matrices are now complex, as a result of the $-2im/r^2$ terms, it is increased by another factor of two. (Incidentally, it is precisely because these matrices are now complex that it is more convenient at this point to recombine the $\cos(m\phi)$ and $\sin(m\phi)$ coefficients into a single complex coefficient instead.) The result is that the memory needed for just a single azimuthal mode is indeed substantial; for example, at the by no means excessive truncation of $K_U = 40$, $L_U = 60$, one needs 20 Mbytes just to store the LU decomposition of the two $\mathbf{X}$ matrices. And worst of all, at this point we cannot re-use the same matrices for all the different m values, because they clearly do depend on m now, both through the $-2im/r^2$ terms in the diagonal blocks as well as through the use of the recursion relations (23) to obtain the off-diagonal blocks. We thus need over 20 Mbytes for each m, which will severely restrict how many azimuthal modes we can afford to include in our calculation. Thus, while this rapid rotation method may well allow us to take larger time steps than before, the price in terms of extra memory requirements is indeed very substantial.

And finally, we might ask ourselves whether even this rapid rotation method will allow us to take so much larger time steps that we can reach $Ro = O(10^{-8})$. Probably not, for the simple reason that these very rapid time scales will still be present in the equations, and our

time step still cannot be large compared with the shortest time scale present in the equations. So, we could do one of two things. We could either just reduce Ro as far as we can, down to $Ro = O(10^{-4})$ perhaps, and hope that the results we obtain then are not too dissimilar from the ones we would have obtained at $Ro = O(10^{-8})$. Or, we could filter out these very rapid time scales entirely by setting $Ro \equiv 0$, and hope the results we obtain then are not too dissimilar from the ones we would have liked to have obtained.

It is in fact possible to develop an extreme version of this rapid rotation method in which inertia is neglected entirely. Indeed, historically this is the method that was developed first [15,16], and is the one used in all the geophysical applications of various versions [3,17,18] of this code (see also Reference [19] for a similar method). If inertia is to be neglected entirely, we note first of all that the whole character of the momentum equation changes dramatically, in that it is no longer time stepped at all, but is simply solved directly at each time step of the induction and temperature equations. However, this direct solution then proceeds much as before. In particular, the general structure of these two (for each m) $\mathbf{X}$ matrices is exactly the same, so we can still pre-compute their LU decompositions, and multiply them onto the output of the forcing routine (note that when $Ro = 0$, $\mathbf{F}$ is indeed independent of $\mathbf{U}$) to obtain the solution. We thus see that setting Ro identically equal to zero rather than letting it be small but non-zero is in fact quite advantageous not only because it filters out some of these very rapid time scales, but also because it completely eliminates the need for these $\mathbf{Y}$ matrices, thereby reducing the memory requirements, somewhat anyway.

However, one should be aware that even after filtering out the very rapid time scales associated with the extreme smallness of the Rossby number, there are still very rapid time scales associated with the extreme smallness of the Ekman number ($E = O(10^{-12})$ in the Earth's core [7]). It turns out that torsional oscillations about a balance known as a Taylor state [20] can occur on time scales as rapid as $O(E^{1/2})$ (see, for example, Reference [7] for a discussion of these torsional oscillations). One will therefore still not be able to take very large time steps in the induction and temperature equations. Indeed, it has even been suggested [21] that restoring inertia, at least in the axisymmetric mode in which these torsional oscillations arise, may help in damping out these very rapid time scales; if $Ro > O(E^{1/2})$ including inertia should stabilize the system by suppressing all time scales more rapid than $O(Ro)$, whereas if $Ro < O(E^{1/2})$ including inertia should destabilize the system by introducing new time scales more rapid than $O(E^{1/2})$. One thus sees that, whichever version of the rapid rotation method—Ro small but non-zero, or Ro identically zero—is most appropriate depends on precisely what parameter values one is most interested in. Either way, however, the geophysically relevant limit of extremely small Rossby and Ekman numbers is a very stiff problem, and will always require considerable computational resources.

2.3. Boundary conditions

In the previous two sections we have already alluded to the two boundary conditions on e_{lm}, and the four boundary conditions on f_{lm}, and how they can (supposedly) be implemented as various rows of these $\mathbf{X}$ matrices; in this section we want to examine that claim more closely. As previously noted, the two types of physical boundary conditions we might be interested in imposing are no-slip, for which we want

$$U_r = 0, \qquad U_\theta = 0, \qquad U_\phi = 0 \quad \text{at } r = r_\mathrm{i},\ r_\mathrm{o} \tag{24a}$$

or stress-free, for which we want

$$U_r = 0, \qquad \frac{\partial}{\partial r}\left(\frac{U_\theta}{r}\right) = 0, \qquad \frac{\partial}{\partial r}\left(\frac{U_\phi}{r}\right) = 0 \quad \text{at } r = r_\mathrm{i},\ r_\mathrm{o} \tag{24b}$$

Translated into conditions on e_{lm} and f_{lm}, these become

$$f_{lm} = 0, \qquad \frac{\mathrm{d}}{\mathrm{d}r} f_{lm} = 0, \qquad e_{lm} = 0 \tag{25a}$$

for no-slip, or

$$f_{lm} = 0, \qquad \frac{\mathrm{d}}{\mathrm{d}r}\left(\frac{1}{r^2}\frac{\mathrm{d}}{\mathrm{d}r} f_{lm}\right) = 0, \qquad \frac{\mathrm{d}}{\mathrm{d}r}\left(\frac{1}{r^2} e_{lm}\right) = 0 \tag{25b}$$

for stress-free. It is important to emphasize also that these conditions (25) implement precisely Equations (24), and nothing else. In particular, although this decomposition (4) does raise the order of the equations, it does not require extra boundary conditions. The reason for this is that because the unit vector $\hat{\mathbf{r}}$ in (4) is purely radial, the extra derivatives are purely angular—the $l(l+1)/r^2$ factors in (6) or (22)—and hence do not lead to extra boundary conditions being required in r.

Since we have expansions (9) and (16) for e_{lm} and f_{lm}, and noting of course that we can exactly evaluate the Chebyshev polynomials and their derivatives at the endpoints $x = \pm 1$, these boundary conditions (25) do indeed convert into linear constraints on the Chebyshev coefficients. For example, $f_{lm} = 0$ at $r = r_\mathrm{i},\ r_\mathrm{o}$, which must be satisfied for both no-slip and stress-free, becomes simply

$$\sum_{k=1}^{K_U+4} (-1)^k f_{klm} = \sum_{k=1}^{K_U+4} f_{klm} = 0 \tag{26}$$

Thus, the two rows of $\mathbf{X}$ that implement these conditions are nothing other than $[-1,\ 1,\ -1,\ 1,\ \ldots\ ]$ and $[1,\ 1,\ 1,\ 1,\ \ldots\ ]$. All the other conditions are implemented similarly. And we note that since none of these conditions (25) involves m, none of these rows of the $\mathbf{X}$s will involve m either, and so in the slow rotation method we can indeed re-use the same $\mathbf{X}$s for all the different m values.

2.4. Inner core rotation

Strictly speaking, we were a little hasty in asserting in the previous section that the no-slip boundary conditions would include $U_\phi = 0$ at $r = r_\mathrm{i}$; if the inner core is free to rotate about the vertical axis we would have to impose instead

$$U_\phi = \Omega r \sin\theta \quad \text{at } r = r_\mathrm{i} \tag{27}$$

where Ω is this inner core rotation rate, which of course must also be specified in some way. In this section we will consider how to implement (27), and how to determine Ω.

Translated again into conditions on e_{lm} and f_{lm}, (27) becomes

$$e_{10} = \Omega r_\mathrm{i}^2 \tag{28}$$

and the same as before for all the other modes. That is, a solid body rotation of the inner core alters only the boundary condition on this one single mode e_{10}. We thus see that, provided we know what Ω is, we can include a rotation of the inner core simply by imposing an inhomogeneous rather than a homogeneous boundary condition on e_{10}. And of course imposing such an inhomogeneous boundary condition is no problem at all; we merely set the appropriate entry of the column vector **DE** to Ωr_i^2 instead of to 0.

In some problems, such as the spherical Couette flow that we will consider in the next section, Ω is indeed known, because it is externally imposed. In other problems, however, such as the geodynamo, Ω is not externally imposed, but is to be determined as part of the solution. The equation that then determines Ω is simply the torque balance on the inner core [22]

$$C\frac{\mathrm{d}\Omega}{\mathrm{d}t} = \Gamma \tag{29}$$

where C is its polar moment of inertia, and

$$\Gamma = E\int r\frac{\partial}{\partial r}\left(\frac{U_\phi}{r}\right)\bigg|_{r=r_\mathrm{i}} r_\mathrm{i}\sin\theta\ \mathrm{d}S + \int B_\phi B_r\bigg|_{r=r_\mathrm{i}} r_\mathrm{i}\sin\theta\ \mathrm{d}S \tag{30}$$

is the total axial torque acting upon it, consisting of a viscous and an electromagnetic contribution. So, we just time step (29) along with all the other equations, using the same (now entirely explicit) second-order Runge–Kutta method, and then at each time step we do know Ω, and so we know what the above inhomogeneous boundary condition on e_{10} should be.

The only difficulty with this approach arises if we are using the inertia-less version of the rapid rotation method, when consistency suggests that we should also neglect the inertia of the inner core. In the Earth's core the inertial time scale of the inner core certainly is comparable to the inertial time scales of the outer core [23], so if we filter out the latter we probably should also filter out the former. However, if we do neglect the inertia of the inner core, Equation (29) for determining Ω becomes simply $\Gamma = 0$, which does not seem to involve Ω at all! Fortunately, even this situation can be salvaged; if we explicitly work out the viscous torque in terms of the e_{lm} and f_{lm}, we find that it too involves only the one single mode e_{10}. As a result, setting $\Gamma = 0$ yields

$$\frac{8\pi}{3}Er_\mathrm{i}^4\frac{\mathrm{d}}{\mathrm{d}r}\left(\frac{1}{r^2}e_{10}\right)\bigg|_{r=r_i} = -\int B_\phi B_r\bigg|_{r=r_\mathrm{i}} r_\mathrm{i}\sin\theta\ \mathrm{d}S \tag{31}$$

And we now realize that we can use this as our inhomogeneous boundary condition on e_{10}, since of course **B** is known (at each time step of the induction equation). And having solved for e_{10}, it is Equation (28) that then determines Ω. So we see that in the inertia-less version an inner core rotation is implemented in a very roundabout manner, with the original equation for Ω being used as the boundary condition, and the original boundary condition then determining Ω. However, the important point to note is that we can indeed include an inner core rotation, both externally imposed as well as internally determined, in both the slow and rapid rotation methods, and even in both versions of the rapid rotation method.

2.5. Sample results

Having described the basic numerical method, we next present some sample results, on the theory that any code is only as good as the results it produces. Since the rapid rotation method has already been extensively used in the past [15–18], we will restrict attention to the slow rotation method. The particular problem we will consider is spherical Couette flow, in which a fluid in a spherical shell is driven by fixing the outer sphere and rotating the inner one. This problem will illustrate, incidentally, that although this code was developed primarily with various geophysical and astrophysical applications in mind, it is sufficiently general that it can be applied to model laboratory experiments as well.

Most earlier experimental [24,25] and numerical [26,27] work in spherical Couette flow has been in the thin-gap limit. While this limit presents some interesting numerical challenges as well, in particular the need for relatively high truncation in θ (which we will also encounter below), it turns out that most of the solutions one obtains remain purely axisymmetric. Since we claim, however, that this code can cope with non-axisymmetric solutions just as readily as with axisymmetric ones, we would like something a little more challenging. Very conveniently, a remarkable set of experiments [28] has recently been done in the wide-gap limit, in which a non-axisymmetric instability was not only observed, but measured with sufficient accuracy that one should be able to do a detailed numerical comparison.

The equation one must solve in spherical Couette flow is

$$\frac{\partial}{\partial t}\mathbf{U} + (\mathbf{U}\cdot\nabla)\mathbf{U} = -\nabla p + Re^{-1}\nabla^2\mathbf{U} \tag{32}$$

where the Reynolds number $Re = \Omega r_i^2/\nu$ is a measure of the imposed rotation of the inner sphere Ω. One thus notes, incidentally, that it is usual in this area to non-dimensionalize length by r_i rather than by $r_o - r_i$, as we did here. However, such minor differences are easily accommodated, and we realize that we should then be able to solve (32) using our slow rotation method, with no-slip boundary conditions and this externally imposed rotation of the inner sphere.

The other parameter characterizing the system is the aspect ratio $\beta = (r_o - r_i)/r_i$. These experiments [28] then revealed a particularly interesting instability for the two aspect ratios $\beta = 1/2$ and $1/3$ (in fact, inserting the precise values of r_i and r_o, one obtains $\beta = 0.498$ and 0.336). For sufficiently small Reynolds numbers one obtains a steady axisymmetric flow having the equatorial symmetry of the imposed forcing. However, if one gradually increases *Re*

one eventually reaches a critical value Re_c, 1244 for $\beta = 1/2$ and 2628 for $\beta = 1/3$, at which this solution becomes unstable to a non-axisymmetric disturbance. Precisely at Re_c, this disturbance will of course consist of just a single non-axisymmetric mode, which is observed to be $m_c = 5$ for $\beta = 1/2$ and $m_c = 6$ for $\beta = 1/3$.

Furthermore, this single mode will also have a definite equatorial symmetry, and from the fact that the total flow, consisting of the axisymmetric background plus the non-axisymmetric disturbance, is observed to be equatorially asymmetric, we deduce that it must have the opposite symmetry of the basic axisymmetric flow. So, for these two different aspect ratios, we have here a clear experimental prediction of the critical Reynolds number for the onset of the most unstable mode, as well as the azimuthal wavenumber and equatorial symmetry of that mode, and we would just like to see how closely our numerical results agree.

We begin by restricting attention to purely axisymmetric solutions, to obtain this steady background flow. Having time stepped to the equilibrium solution at some particular value of Re (which in itself is non-trivial at these large Reynolds numbers), we linearize about this axisymmetric flow to test the stability of single non-axisymmetric modes at a time. We thus solve a sequence of two-dimensional problems rather than a single three-dimensional one. If the axisymmetric flow turns out to be stable to all the non-axisymmetric modes, and both equatorial symmetries for each, we increase Re and repeat the process, until we find the particular value Re_c where the axisymmetric flow first becomes unstable to one of the modes, i.e. where we first get an exponentially growing rather than decaying mode. Proceeding directly to the results, we find that for $\beta = 1/2$, $Re_c = 1244$, and $m_c = 5$, whereas for $\beta = 1/3$, $Re_c = 2684$ and $m_c = 6$. For $\beta = 1/2$ we thus have perfect agreement, so good that we cannot even say how small the error is, whereas for $\beta = 1/3$ it is 2 per cent. It is of course disappointing that we did not obtain perfect agreement for both aspect ratios. However, if one considers the experimental measurement errors [28], one finds that Re is only accurate to within 1 per cent, and β to within 0.5 per cent. And considering that β clearly has a very strong influence on Re_c (we in fact used the precise values 0.498 and 0.336), one should not expect agreement to within better than about 2 per cent.

Figures 2 and 3 show the solutions right at Re_c for $\beta = 1/2$ and $1/3$ respectively. The first two quantities shown are the angular velocity, varying from one on the inner sphere to zero on the outer, as it must, and the induced meridional circulation of the axisymmetric basic state. The third quantity is a meridional section of the non-axisymmetric instability, showing the U_ϕ component only (for the sake of definiteness the particular meridional section shown is the one for which $\int_0^{\pi/2} \int_{r_i}^{r_o} U_\phi r \, dr \, d\theta = 0$). We note first of all that the equatorial symmetry of the instability is indeed the opposite of the basic state's, with U_ϕ being symmetric for the basic state but antisymmetric for the instability.

We note next that both the basic state as well as the instability exhibit considerable structure, particularly in θ. And indeed, the truncations (K_U, L_U) we used to obtain these results were correspondingly high, (30, 160) for $\beta = 1/2$ and (30, 240) for $\beta = 1/3$. That the thinner gap requires even higher truncation in θ is readily understandable when one looks at Figures 2 and 3; the instability seems to be triggered by this equatorial tongue in the angular velocity, and the thickness of this feature seems to be a more or less constant fraction of the gap width, so a thinner gap implies a thinner tongue, which implies finer structure in θ. This would suggest, incidentally, that if one wanted to do a fully three-dimensional calculation of

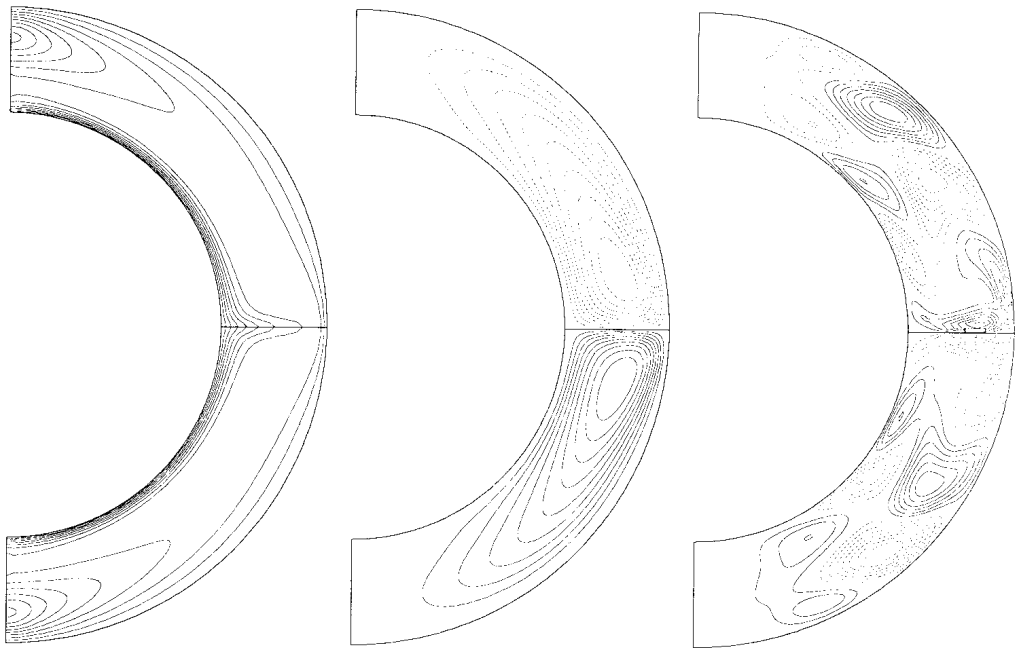

Figure 2. From left to right, contour plots of the angular velocity and meridional circulation of the axisymmetric basic state, and the U_ϕ component of the non-axisymmetric instability, at Re_c, and for $\beta = 1/2$.

this instability in the supercritical regime, which would definitely be worthwhile, since there were some very interesting subsequent transitions observed in the experiments [28], one would be well advised to do it for $\beta = 1/2$ rather than $1/3$.

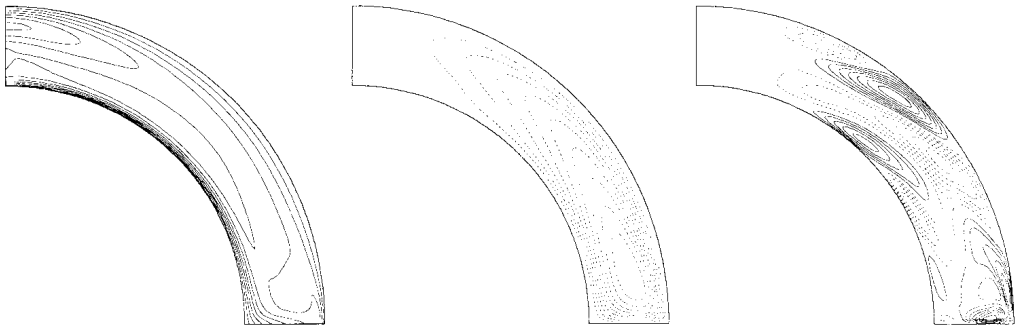

Figure 3. As in Figure 2, but for $\beta = 1/3$. Also, to save space we are only showing one hemisphere here; the equatorial symmetry is indeed also as in Figure 2.

Finally, one might say a few words about the time step used, particularly since this is the no-slip case for which the time stepping of the diffusive terms for f is nominally only first-order accurate. For the axisymmetric basic flow, of course, one can make the time step as large as stability considerations will allow, since any steady state will always be exactly the same, independent of the size of the time step or the order of the method. Indeed, it is advantageous at this point to change these weightings of the diffusive terms for e as well, to make the procedure even more stable and hence allow a larger time step. If one sets the weightings for both f and e to 0.6/0.4, one can take time steps as large as 0.02, at which point a few thousand time steps are enough to converge to the steady state solution.

However, for the non-axisymmetric instability, which is oscillatory rather than steady, we are rather more reluctant to accept such a large time step, so we try $\Delta t = 0.01$, which certainly works, and yields a drifting-wave solution, just as we expect. If we then repeat the calculation with $\Delta t = 5 \cdot 10^{-3}$ and for good measure even with $\Delta t = 2 \cdot 10^{-3}$ we find that we get the same solution with all three time steps, indicating that even $\Delta t = 0.01$ is already small enough to give a very accurate answer. The reason for this is of course precisely as indicated above: Δt is 0.01 as measured on the rotational time scale, which corresponds to $0.01\ Re^{-1}$ as measured on the diffusive time scale, and this is so small that one should expect extremely good accuracy even from a first-order procedure. We thus see that even though the time stepping of the diffusive terms is nominally only first-order accurate, because of this factor Re disparity between the rotational and diffusive time scales, all terms are in fact time stepped extremely accurately.

3. THE INDUCTION EQUATION

Turning next to the induction equation (2), and again using (5), the r components of it and its curl yield

$$\sum_{m,l} \frac{l(l+1)}{r^2} \left[\frac{\partial}{\partial t} - L_l \right] h_{lm}(r, t) P_l^{|m|} \, \mathrm{e}^{\mathrm{i}m\phi} = \hat{\mathbf{r}} \cdot \nabla \times (\mathbf{U} \times \mathbf{B}) \tag{33a}$$

$$\sum_{m,l} \frac{l(l+1)}{r^2} \left[\frac{\partial}{\partial t} - L_l \right] g_{lm}(r, t) P_l^{|m|} \, \mathrm{e}^{\mathrm{i}m\phi} = \hat{\mathbf{r}} \cdot \nabla \times \nabla \times (\mathbf{U} \times \mathbf{B}) \tag{33b}$$

These equations (33) for g_{lm} and h_{lm} are the same as our original Equation (6a) for e_{lm}, and can thus be time stepped in exactly the same way. However, one might just note that when we expand g_{lm} and h_{lm} in terms of $K_B + 2$ Chebyshev polynomials now, there is no reason why K_B should have to equal K_U, just like we already agreed that we might want to allow for different angular truncations in the expansions (5). Of course, if K_U and K_B are different, the momentum and induction equations will be enforced at different radial collocation points, but this poses no problems.

The only thing we still need to verify is that the boundary conditions, which we have not yet considered in terms of the underlying physics either, can also be implemented in the same way as before. In terms of the physics, the condition to be imposed in an insulating region is that the current density $\mathbf{J} = \nabla \times \mathbf{B}$ must vanish. Analytically solving for such so-called potential

fields, and then matching **B** across the boundary, one can ultimately obtain conditions involving only the g_{lm} and h_{lm} at the boundary [8]. If we take both the regions $r < r_i$ and $r > r_o$ to be insulators, then performing this matching yields

$$g_{lm} = 0, \qquad \left(\frac{\mathrm{d}}{\mathrm{d}r} - \frac{l+1}{r}\right) h_{lm} = 0 \tag{34}$$

at $r = r_i$, and

$$g_{lm} = 0, \qquad \left(\frac{\mathrm{d}}{\mathrm{d}r} + \frac{l}{r}\right) h_{lm} = 0 \tag{35}$$

at $r = r_o$. We emphasize also that these boundary conditions implement precisely this matching to potential fields, and nothing else. In particular, we note that Equations (33) involve only two radial derivatives, just like the original Equation (2), and so once again the decomposition (4) does not lead to extra boundary conditions being required. Having obtained these boundary conditions, we then note that we can indeed implement them in exactly the same way as before. In particular, because these conditions also do not involve m, none of the corresponding rows of the **X** matrices will involve m either, and so we can again re-use the same **X**s for all the different ms, just as in the slow rotation method for the momentum equation.

Of course, in most geophysical or astrophysical applications the region $r < r_i$ probably should not be taken to be an insulator; in the case of the geodynamo in particular it has been shown [29,30] that the finite conductivity of the inner core can play a vital role in the dynamics. And indeed the inclusion of a finitely conducting inner core is by now fairly standard [30,31], certainly in the various versions [3,17,18] of this code. However, we have never described in detail precisely how it is implemented, so we would like to consider this problem here.

The first point to note is that one cannot implement a finitely conducting inner core simply by imposing appropriate boundary conditions at $r = r_i$ as we did for an insulating inner core above. In a finitely conducting inner core, the evolution of the magnetic field is determined not only by changes on the boundary, but also by its own past history. Indeed, it is precisely the presence of this new time scale, the magnetic diffusion time of the inner core, which seems to have such a powerful effect [29,30]. We thus realize that we will have to solve for the field in the inner core as well. The equations we want to solve in the inner core are of course very similar to (33), namely

$$\sum_{m,l} \frac{l(l+1)}{r^2} \left[\frac{\partial}{\partial t} - \sigma^{-1} L_l\right] \hat{h}_{lm}(r, t) P_l^{|m|} \mathrm{e}^{\mathrm{i}m\phi} = \hat{\mathbf{r}} \cdot \nabla \times (\mathbf{U} \times \mathbf{B}) \tag{36a}$$

$$\sum_{m,l} \frac{l(l+1)}{r^2} \left[\frac{\partial}{\partial t} - \sigma^{-1} L_l\right] \hat{g}_{lm}(r, t) P_l^{|m|} \mathrm{e}^{\mathrm{i}m\phi} = \hat{\mathbf{r}} \cdot \nabla \times \nabla \times (\mathbf{U} \times \mathbf{B}) \tag{36b}$$

where σ is the ratio of inner to outer core conductivities ($\sigma = 1$ is in fact the case one will almost always take, but we might as well consider the general case here), and where $\mathbf{U} = \mathbf{\Omega} r \sin\theta \hat{\mathbf{e}}_\phi$ is now the purely solid-body rotation of the inner core (so the terms on the right-hand sides just amount to a rotation of the phases of the various components, and can be worked out analytically).

However, even though the equations are essentially the same, the Chebyshev expansions will have to be slightly different, because we are now including the origin. In particular, one can show (see, for example, the appendix of Reference [32] for a particularly illuminating derivation) that the functions $\hat{g}_{lm}$ and $\hat{h}_{lm}$ must have the radial symmetry

$$\hat{p}_{lm}(-r) = (-1)^{l+1}\hat{p}_{lm}(r) \tag{37}$$

where $\hat{p}_{lm}(r)$ stands for either $\hat{g}_{lm}$ or $\hat{h}_{lm}$. One can show further that close to the origin these radial functions $\hat{p}_{lm}(r)$ must tend to zero at least as quickly as r^2 for the even modes and as r^3 for the odd modes. This suggests that one possible Chebyshev expansion is

$$\hat{p}_{lm}(r) = \sum_{k=1}^{K'_B+1} \hat{p}_{klm} T_{2k-l}(r/r_\mathrm{i}) r^{l'} \tag{38}$$

where l' is 1 for odd l and 2 for even l. This is certainly a little more complicated than before, since we now have slightly different radial expansion functions depending on whether l is even or odd, but since we set up these $\mathbf{X}$ matrices separately for each l anyway, it amounts to nothing more than a minor detail, and even then only in the pre-computation program.

With these expansions in place, we then enforce (36) at K'_B collocation points, the K'_B zeros of $T_{2K'_B+1}(r/r_\mathrm{i})$ on (0, 1). That the origin is therefore not included as one of the collocation points is sufficiently important that it is probably worth mentioning explicitly, as this essentially enables us to avoid the co-ordinate singularity entirely. Indeed, because we are now only using this half-interval (0, 1), the collocation points are not even especially closely spaced near the origin, which is precisely what one wants, since physically there is nothing special about it, and so there is no reason why one would want to concentrate resolution there.

Having decided on the radial expansion functions and collocation points for (33) in the outer core, and (36) in the inner, the only remaining question is what matching conditions to impose at $r = r_\mathrm{i}$ and how to implement them in the usual way, as rows of the usual $\mathbf{X}$ matrices. These matching conditions are, first of all,

$$h_{lm} = \hat{h}_{lm}, \qquad \frac{\mathrm{d}}{\mathrm{d}r} h_{lm} = \frac{\mathrm{d}}{\mathrm{d}r}\hat{h}_{lm} \tag{39}$$

$$g_{lm} = \hat{g}_{lm}, \qquad \frac{\mathrm{d}}{\mathrm{d}r} g_{lm} = \sigma^{-1}\frac{\mathrm{d}}{\mathrm{d}r}\hat{g}_{lm} \tag{40}$$

representing continuity of the three components of the magnetic field, (39a), (39b) and (40a), and continuity of the tangential components of the associated electric field (40b).

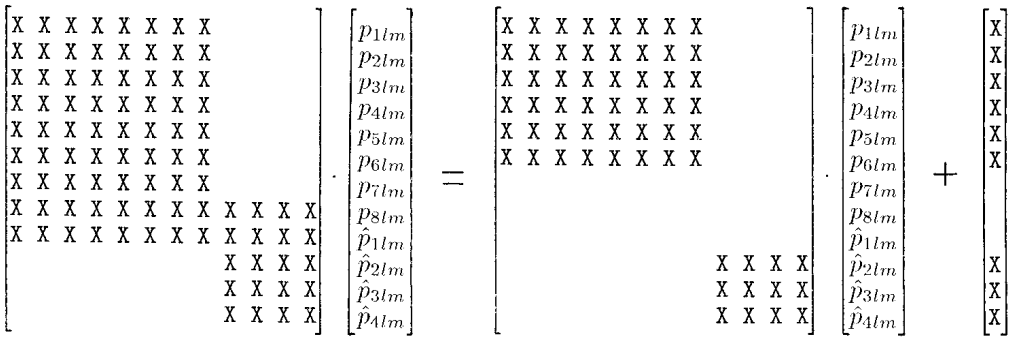

Figure 4. The structure of the matrix system one must solve when implementing a finitely conducting inner core; p stands for either g or h, with the p_{klm} being the Chebyshev coefficients in the outer core, and the $\hat{p}_{klm}$ the coefficients in the inner core. Also, the usual superscripts $n+1$ on the left and n on the right have been omitted for clarity.

Figure 4, which schematically illustrates the equivalent of the matrix equations (12) or (17), shows how both the inner and outer equations (33) and (36), the matching conditions (39) or (40), and the outer boundary condition (35a) or (35b), are then all implemented simultaneously. The first K_B rows implement (33a) or (33b) at the K_B collocation points in the outer core, exactly as before. The next row implements the outer boundary condition (35a) or (35b), again as before. The next two rows then implement the two matching conditions (39) or (40), and finally the last K'_B rows implement (36a) or (36b) at the K'_B collocation points in the inner core. Incidentally, one notes that in this example we chose $K'_B = K_B/2$; in a shell of radius ratio 1/3 (corresponding to the Earth) this will ensure that one has roughly equal resolution just inside and outside the inner core boundary. And although this latest manifestation of these ubiquitous **X** matrices is somewhat more complicated than it was for an insulating inner core, we note that the most crucial feature is preserved, namely that the matching conditions and therefore the matrices once again do not involve m, and so we can re-use the same **X**s for all the different ms yet again.

Having set up these matrices as in Figure 4, one can again check how well the numerically computed free decay rates compare with the analytically derived values, for various conductivity ratios σ. One obtains excellent agreement in all cases, with Table III showing some representative results. And since free decay is the only process going on in the inner core (apart from this simple rotation of the phases due to the possible rotation of the inner core), one can then be quite confident that everything is working properly. We thus see that with Chebyshev expansions of this type, implementing a finitely conducting inner core is hardly more complicated than implementing an insulating one.

4. THE TEMPERATURE EQUATION

After the various options that presented themselves in the course of solving the momentum and induction equations, the temperature equation (3) is refreshingly straightforward. Using

Table III. The analytically derived lowest free decay rates of the system shown in Figure 4, and the relative error in the numerically computed decay rates, for the $l = 5$ modes at the two values $\sigma = 1/10$ and 10.

	$\sigma = 1/10$		$\sigma = 10$	
	G	h	g	h
Analytic	38.9940319	29.7588808	33.2660269	26.0379254
$K_B = 8$	0.0000001	0.0000016	0.0022187	0.0010660
$K_B = 10$	0.0000000	0.0000000	0.0005254	0.0002015
$K_B = 12$	0.0000000	0.0000000	−0.0000192	−0.0000180
$K_B = 14$	0.0000000	0.0000000	−0.0000005	0.0000013
$K_B = 16$	0.0000000	0.0000000	0.0000000	0.0000000

Also, since Tables I and II already show similar results at fixed truncation and varying time step, we show here results at fixed time step ($\Delta t = 10^{-6}$, so small that essentially all the error is from the truncation) and varying truncation. In each case $K'_B = K_B/2$.

Equation (5c), it becomes simply

$$\sum_{m,l} \left[\frac{\partial}{\partial t} - q\left(L_l + \frac{2}{r}\frac{\partial}{\partial r} \right) \right] \Theta_{lm}(r, t) P_l^{|m|} \, e^{im\phi} = -\mathbf{U}\cdot\nabla\Theta \tag{41}$$

which is sufficiently similar to (6a) and (33) that it too can be time stepped in the same way, with the usual boundary conditions on Θ_{lm} or $d\Theta_{lm}/dr$ also not involving m. The only other point to note about the temperature equation is to remind ourselves once again that here we must include the $l = m = 0$ mode.

5. NONLINEAR COUPLING TERMS

Finally, we discuss very briefly how to compute the nonlinear forcing terms that appear on the right-hand sides of the various equations. The five terms we need to compute are the r components of the first and second curls of $\mathbf{F}$ for the momentum equation and of $\mathbf{U} \times \mathbf{B}$ for the induction equation, and $-\mathbf{U}\cdot\nabla\Theta$ for the temperature equation. The task we are then faced with is the following:

Given the spectral coefficients of e, f, etc., how do we compute $\mathbf{U}$, $\mathbf{B}$, etc., how do we compute these various curls and grads, and finally, how do we separate these quantities back into the different spherical harmonic components we need to time step the equations? The method we use to carry out all of these operations is the well-known pseudo-spectral method [12], in which one repeatedly switches back and forth between spectral and real space. That is, one evaluates $\mathbf{U}$, $\mathbf{B}$, etc. at certain points in real space, does the multiplications to obtain $\mathbf{F}$ and $\mathbf{U} \times \mathbf{B}$ in real space, then switches back to spectral space to do the derivatives and thereby finish the calculation. As this method is entirely standard, we omit the details, and content ourselves instead with a brief discussion of how to choose these real-space points at which the various quantities are to be evaluated.

In ϕ we take $2M$ uniformly spaced points, where M satisfies

$$2M \geq \max(3M_U + 1, M_U + 2M_B + 1, M_U + 2M_T + 1) \tag{42}$$

This ensures that all the nonlinear terms are exactly dealiased in ϕ. For example, the term $-Ro(\mathbf{U}\cdot\nabla)\mathbf{U}$ in $\mathbf{F}$ will generate Fourier modes up to $\exp(\mathrm{i}2M_U\phi)$, so if we wanted to recover all $2M_U$ of them correctly we would need to evaluate $\mathbf{F}$ at, at least, $4M_U + 1$ points. However, remembering that we only want the first M_U of them correctly, it turns out we need to evaluate $\mathbf{F}$ only at at least $3M_U + 1$ points. This result is well known [12], and yields the first of the inequalities in (42). Exactly analogous considerations of the other nonlinear terms—that is, how many Fourier modes will they generate, and how many of them do we want to recover correctly—then yield the remaining inequalities.

Similarly, in θ we take the L zeros of $P_L^0(\cos\theta)$ for the points, where L must satisfy

$$2L \geq \max(3L_U + 1, L_U + 2L_B + 1, L_U + 2L_T + 1) \tag{43}$$

to ensure all terms are exactly dealiased in θ. The considerations that lead to (43) are exactly analogous to the considerations that lead to (42), just applied to the Legendre rather than the Fourier transform. See, for example, [33] for a general discussion of the Legendre transform.

Finally, in r we take the K zeros of $T_K(x)$, where K satisfies an inequality similar to (42) and (43). We note though that unlike the ϕ and θ structures, the r structure will still not be exactly dealiased, because various factors of r^{-1} throughout do not have finite Chebyshev spectra. However, for any reasonable radius ratio the Chebyshev spectrum of r^{-1} does drop off very rapidly, so if we choose K to satisfy an inequality similar to (42) and (43), the r structure too will be essentially exactly dealiased.

Having decided at which points we need to do these real-space evaluations, we are in a position to discuss the computational effort involved. If fast transforms were used in all three directions it would scale as $O(N^3 \ln N)$, whereas if even one slow transform is used it will scale as $O(N^4)$. N here is the number of points in any one direction, for simplicity assumed to be comparable in all three. Unfortunately, at least one of the transforms will inevitably be slow, namely the Legendre transform in θ. A fast Legendre transform does exist [34], but only becomes competitive at truncations beyond what one can do in a fully three-dimensional calculation. The computational effort is thus quartic in the number of points in any one direction, and hence also quartic in the truncation in any one direction. Nominally, it is thus of the same order as the time stepping parts of the code discussed in Section 2. However, it turns out that the bulk of the computational effort is in fact spent on this evaluation of the nonlinear terms, primarily because according to (42) and (43) the number of points in any one direction is at least 1.5 times greater than the corresponding number of spectral coefficients. And finally, we might just note that the memory requirements of this evaluation are cubic in the truncation, regardless of whether fast or slow transforms are used.

6. CONCLUSION

In this work we have presented various methods of solving the magneto-convection equations in spherical geometry. In particular, we showed how one can solve the momentum equation in

two different ways, depending upon whether one treats the Coriolis force explicitly or implicitly. We discovered that in terms of the memory requirements, the explicit method is cubic in the truncation, whereas the implicit method is quartic. However, the implicit method does have the advantage that it is far better at exploring the limit of rapid rotation. It is by using both methods that we are thus best able to explore a broad range of problems. For example, for the spherical Couette flow problem presented in Section 2.5 here, as well as for a recent astrophysical application [35], the explicit slow rotation method was most suitable, whereas for our geophysical applications [3,17,18] the implicit rapid rotation method was.

ACKNOWLEDGMENTS

The author thanks Gary Glatzmaier, Glenn Ierley, and Chris Jones for many enjoyable discussions on the finer points of numerical methods. This work was supported by the Nuffield Foundation under the 'Awards to Newly Appointed Science Lecturers' Programme, and by PPARC grant GR/L40922.

REFERENCES

1. Russell CT. Magnetic fields of the terrestrial planets. *Journal of Geophysical Research E* 1993; **98**: 18681–18695.
2. Connerney JEP. Magnetic fields of the outer planets. *Journal of Geophysical Research E* 1993; **98**: 18659–18679.
3. Sarson GR, Jones CA, Zhang K, Schubert G. Magnetoconvection dynamos and the magnetic fields of Io and Ganymede. *Science* 1997; **276**: 1106–1108.
4. Crawford JD, Knobloch E. Symmetry and symmetry-breaking bifurcations in fluid dynamics. *Annual Reviews of Fluid Mechanics* 1991; **23**: 341–387.
5. Knobloch E. Symmetry and instability in rotating hydrodynamic and magnetohydrodynamic flows. *Physics of Fluids* 1996; **8**: 1446–1454.
6. Clune T, Knobloch E. Pattern selection in three-dimensional magnetoconvection. *Physica D* 1994; **74**: 151–176.
7. Hollerbach R. On the theory of the geodynamo. *Physics of the Earth and Planetary Interiors* 1996; **98**: 163–185.
8. Bullard EC, Gellman H. Homogeneous dynamos and terrestrial magnetism. *Philosophical Transactions of the Royal Society of London A* 1954; **247**: 213–278.
9. Glatzmaier GA. Numerical simulations of stellar convective dynamos. 1. The model and method. *Journal of Computational Physics* 1984; **55**: 461–484.
10. Quartapelle L, Verri M. On the spectral solution of the three-dimensional Navier–Stokes equations in spherical and cylindrical regions. *Computer Physics Communications* 1995; **90**: 1–43.
11. Dumas G, Leonard A. A divergence-free spectral expansions method for three-dimensional flows in spherical-gap geometries. *Journal of Computational Physics* 1994; **111**: 205–219.
12. Canuto C, Hussaini MY, Quarteroni A, Zang TA. *Spectral Methods in Fluid Dynamics*. Springer: Berlin, 1987.
13. Cox SM, Matthews PC. A pseudo-spectral code for convection with an analytical/numerical implementation of horizontal boundary conditions. *International Journal for Numerical Methods in Fluids* 1997; **25**: 151–166.
14. Abramowitz M, Stegun IA (eds). *Handbook of Mathematical Functions*. Dover: New York, 1968.
15. Hollerbach R. Magnetohydrodynamic Ekman and Stewartson layers in a rotating spherical shell. *Proceedings of the Royal Society of London A* 1994; **444**: 333–346.
16. Hollerbach R. Imposing a magnetic field across a non-axisymmetric shear layer in a rotating spherical shell. *Physics of Fluids* 1994; **6**: 2540–2544.
17. Jones CA, Longbottom AW, Hollerbach R. A self-consistent convection driven geodynamo model, using a mean field approximation. *Physics of the Earth and Planetary Interiors* 1995; **92**: 119–141.
18. Sarson GR, Jones CA. A convection driven geodynamo reversal model. *Physics of the Earth and Planetary Interiors* 1999; **111**: 3–20.
19. Glatzmaier GA, Roberts PH. A three-dimensional convective dynamo solution with rotating and finitely conducting inner core and mantle. *Physics of the Earth and Planetary Interiors* 1995; **91**: 63–75.
20. Taylor JB. The magnetohydrodynamics of a rotating fluid and the Earth's dynamo problem. *Proceedings of the Royal Society of London A* 1963; **274**: 274–283.
21. Jault D. Model-Z by computation and Taylor's conditions. *Geophysical and Astrophysical Fluid Dynamics* 1995; **79**: 99–124.

22. Glatzmaier GA, Roberts PH. Rotation and magnetism of Earth's inner core. *Science* 1996; **274**: 1887–1891.
23. Gubbins D. Rotation of the inner core. *Journal of Geophysical Research B* 1981; **86**: 11695–11699.
24. Sawatzki O, Zierep J. Das Stromfeld im Spalt zwischen zwei konzentrischen Kugelflächen, von denen die innere rotiert. *Acta Mechanica* 1970; **9**: 13–35.
25. Wimmer M. Experiments on a viscous fluid flow between concentric rotating spheres. *Journal of Fluid Mechanics* 1976; **78**: 317–335.
26. Dennis SCR, Quartapelle L. Finite difference solution to the flow between two rotating spheres. *Computers and Fluids* 1984; **12**: 77–92.
27. Marcus PS, Tuckerman LS. Simulation of flow between concentric rotating spheres. *Journal of Fluid Mechanics* 1987; **185**: 1–65.
28. Egbers C, Rath HJ. The existence of Taylor vortices and wide-gap instabilities in spherical Couette flow. *Acta Mechanica* 1995; **111**: 125–140.
29. Hollerbach R, Jones CA. Influence of the Earth's inner core on geomagnetic fluctuations and reversals. *Nature* 1993; **365**: 541–543.
30. Glatzmaier GA, Roberts PH. A three-dimensional self-consistent computer simulation of a geomagnetic field reversal. *Nature* 1995; **377**: 203–209.
31. Kuang W, Bloxham J. An Earth-like numerical dynamo model. *Nature* 1997; **389**: 371–374.
32. Kerswell RR, Davey A. On the linear instability of elliptic pipe flow. *Journal of Fluid Mechanics* 1996; **316**: 307–324.
33. Lloyd D, Gubbins D. Toroidal fluid motion at the top of the Earth's core. *Geophysical Journal International* 1990; **100**: 455–467.
34. Driscoll JR, Healy DM. Computing Fourier transforms and convolutions on the 2-sphere. *Advances in Applied Mathematics* 1994; **15**: 202–250.
35. Drecker A, Hollerbach R, Rüdiger G. Viscosity alpha in rotating spherical shear flows with an external magnetic field. *Monthly Notices of the Royal Astronomical Society* 1998; **298**: 1030–1034.

