

Article

Not peer-reviewed version

Convection in a Rapidly Rotating Spherical Shell: Newton's Method Using Implicit Coriolis Integration

Juan Cruz Gonzalez Sembla , Camille Rambert , Alan Riquier , Fred Feudel , [Laurette S. Tuckerman](#) *

Posted Date: 2 April 2025

doi: 10.20944/preprints202504.0214.v1

Keywords: rotating fluids; Krylov methods; preconditioning; bifurcation; rotating waves



Preprints.org is a free multidisciplinary platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This open access article is published under a Creative Commons CC BY 4.0 license, which permit the free download, distribution, and reuse, provided that the author and preprint are cited in any reuse.

Article

Convection in a Rapidly Rotating Spherical Shell: Newton's Method Using Implicit Coriolis Integration

Juan Cruz Gonzalez Sembla ¹, Camille Rambert ¹, Alan Riquier ² , Fred Feudel ³ ,
Laurette S. Tuckerman ^{1,*} 

¹ PMMH, CNRS, ESPCI Paris, Université PSL, Sorbonne Université, Université de Paris, 75005 Paris, France

² Département de Mathématiques et Applications, École Normale Supérieure - PSL, 75005 Paris, France

³ Institut für Physik und Astronomie, Universität Potsdam, 14476 Potsdam, Germany

* Correspondence: laurette@pmmh.espci.fr

Abstract: Geophysical and astrophysical flows are characterized by rapid rotation. Simulating these flows requires small timesteps to achieve stability and accuracy. Numerical stability can be greatly improved by implicit integration of the terms that are most responsible for destabilizing the numerical scheme. We have implemented an implicit treatment of the Coriolis force in a rotating spherical shell driven by a radial thermal gradient. We have modified the resulting timestepping code to carry out steady-state solving via Newton's method, which has no time-stepping error. The implicit terms have the effect of preconditioning the linear systems, which can then be rapidly solved by a matrix-free Krylov method. We compute branches of rotating waves with azimuthal wavenumbers ranging from 4 to 12. As the Ekman number (the non-dimensionalized inverse rotation rate) decreases, the flows are increasingly axially independent and localized near the inner cylinder, in keeping with well-known theoretical predictions and previous experimental and numerical results. The advantage of the implicit over the explicit treatment also increases dramatically with decreasing Ek, reducing the cost of computation by as much as a factor of 10 for Ekman numbers of order 10^{-5} . We carry out continuation in both Rayleigh and Ekman number and obtain interesting branches in which the drift velocity remains unchanged between pairs of saddle-node bifurcations.

Keywords: rotating fluids; Krylov methods; preconditioning; bifurcation; rotating waves

MSC: 35B32; 37M20; 65F08; 65M70; 65N35; 65P30; 76E06; 76U60

1. Introduction

Convection in a rotating spherical annulus is a classic configuration in hydrodynamics [1–4]. At moderate parameter values, it provides beautiful and deep examples of symmetry-breaking bifurcations leading to complex patterns, e.g. [5–13]. At more extreme parameter values, the turbulent flow it engenders is among the fundamental processes in planets and stars, e.g. [14,15]. We do not attempt here to survey the considerable literature devoted to this configuration nor to derive new results in either of these well-studied regimes. Instead we describe the implementation and capabilities of a pair of numerical codes – two drivers relying on a common set of subroutines – that incorporate an implicit treatment of the Coriolis force to carry out both timestepping and also continuation via Newton's method.

We will illustrate, test, and compare the codes on the simplest convective states, which are rotating waves (RW). Their appearance at onset is a mathematical consequence of the invariance of the system under azimuthal rotation, but not under azimuthal reflection, which is broken by the rotation of the system [16–18]. We have carried out numerical continuation in both Rayleigh and Ekman numbers. We have found interesting examples of branches which contain saddle-node bifurcations separating plateaus in drift frequency.

Our numerical algorithm uses the basic framework described in [11–13,19] and the spherical harmonic library of [20]. The research which is closest to ours is that of [21–24], who, like us, have carried out implicit timestepping and continuation for convection in a rotating spherical shell and explored various interesting parameter regimes. There are a number of differences between our continuation method and theirs, some of which have been analyzed in [25] and which we will discuss after we have described our numerical methods. We mention here also that [14,15,26] have implemented implicit Coriolis timestepping.

2. Physical Description

2.1. Governing Equations

We study classical Rayleigh-Bénard convection in a spherical fluid shell rotating with constant angular velocity Ω about the z axis. The shell is heated from within by imposing a temperature difference ΔT between the inner and outer spherical boundaries. Lengths are non-dimensionalized with the gap size d , so that the dimensionless outer and inner radii are r_{out} and $r_{\text{in}} = r_{\text{out}} - 1$, respectively. The aspect ratio can be specified via r_{out} or via $\eta \equiv r_{\text{in}}/r_{\text{out}} = (r_{\text{out}} - 1)/r_{\text{out}}$. Time is scaled by the viscous diffusion time d^2/ν , where ν is the kinematic viscosity and the flow velocity \mathbf{u} is scaled by the viscous diffusion velocity ν/d . Measuring temperature from a reference temperature T_0 at which the mass density is ρ_0 , we nondimensionalize pressure by $\rho_0\nu$, and temperature by ΔT .

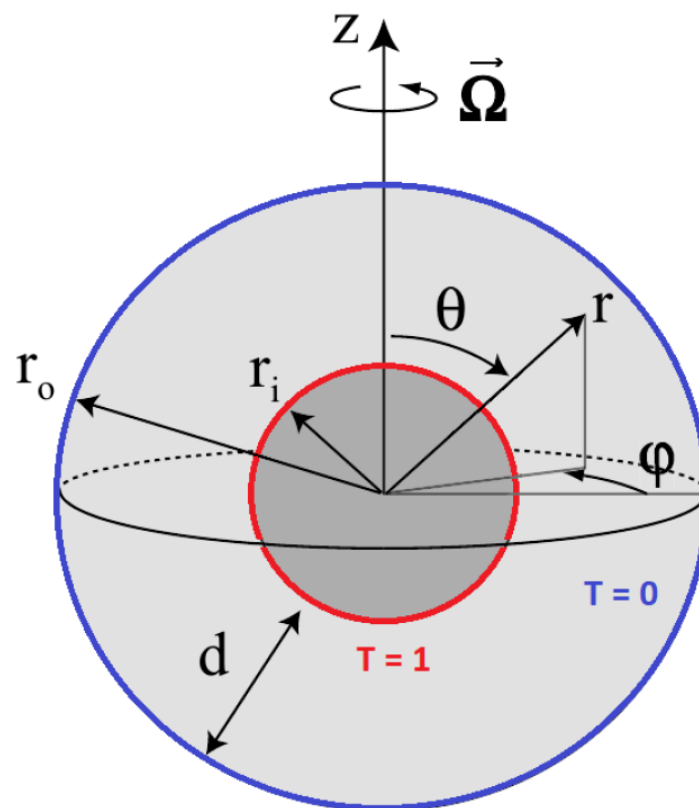


Figure 1. Geometry. The domain is the spherical annulus between an inner and outer sphere of radius r_{in} and r_{out} whose temperatures are fixed at $T = 1$ and $T = 0$. Both spheres rotate around the z axis with angular velocity Ω .

The gravitational acceleration g is assumed to be proportional to the distance r from the center of the sphere (as is the case for the self-gravity in the interior of a spherical body with constant mass

density) and is thus of the form $g = -(g_o/r_{\text{out}})r$, where g_o is the acceleration at radius r_{out} . The resulting nondimensional Boussinesq equations read:

$$\text{Ek} \left[\frac{\partial \mathbf{U}}{\partial t} + (\mathbf{U} \cdot \nabla) \mathbf{U} - \nabla^2 \mathbf{U} \right] = -\nabla P + \text{Ra} T \frac{\mathbf{r}}{r_{\text{out}}} - 2\mathbf{e}_z \times \mathbf{U} \quad (1a)$$

$$\frac{\partial T}{\partial t} + \mathbf{U} \cdot \nabla T = \frac{1}{\text{Pr}} \nabla^2 T \quad (1b)$$

$$\nabla \cdot \mathbf{U} = 0 \quad (1c)$$

The centrifugal acceleration and a gradient part of the acceleration due to buoyancy have been included in the pressure gradient. The non-dimensional parameters used in (1a) above are the Ekman, Prandtl, and Rayleigh numbers:

$$\text{Ek} \equiv \frac{\nu}{\Omega d^2}, \quad \text{Pr} \equiv \frac{\nu}{\kappa}, \quad \text{Ra} = \text{Ra}_{\text{rot}} \equiv \frac{d\alpha g_o \Delta T}{\Omega \nu} \quad (2)$$

where κ is the thermal diffusivity and α is the thermal expansion coefficient. The Rayleigh number Ra used here is adapted to the rotating case and is related to the conventional thermal Rayleigh number Ra_{therm} by

$$\text{Ra}_{\text{therm}} \equiv \frac{d^3 \alpha g_o \Delta T}{\kappa \nu} = \frac{d\alpha g_o \Delta T}{\Omega \nu} \frac{\Omega d^2}{\nu} \frac{\nu}{\kappa} = \text{Ra}_{\text{rot}} \frac{\text{Pr}}{\text{Ek}} \quad (3)$$

In what follows, we will denote Ra_{rot} merely by Ra. No-slip and fixed temperature boundary conditions are applied at the inner and outer radii:

$$\mathbf{U}(r_{\text{in}}) = 0, \quad T(r_{\text{in}}) = 1 \quad (4a)$$

$$\mathbf{U}(r_{\text{out}}) = 0, \quad T(r_{\text{out}}) = 0 \quad (4b)$$

and the conductive solution is:

$$\mathbf{U} = 0, \quad T_{\text{cond}}(r) = \frac{r_{\text{out}} r_{\text{in}}}{r} - r_{\text{in}} \quad (5)$$

Following the benchmark study [27], we set the Prandtl number to $\text{Pr} = 1$ and the radii to $r_{\text{out}} = 20/13$ and $r_{\text{in}} = 7/13$ (so that $\eta = 7/20 = 0.35$) throughout the investigation and we will vary Ek and Ra.

2.2. Overview

The first states to appear at the onset of convection are rotating waves. We denote these by RW_M where M is the azimuthal wavenumber of the rotating wave. Figures 2(a) and 2(b), modeled on those in [12], display properties of some of these rotating wave solutions for $\text{Ek} = 10^{-3}$. Four traveling waves are shown, with azimuthal wavenumber from $M = 2$ to $M = 5$. At $\text{Ek} = 10^{-3}$, RW_4 is the first in Rayleigh number to bifurcate, and therefore it is the only one which is stable at onset. The drift frequency, i.e. the frequency relative to the imposed frequency Ω , decreases as Ra increases, from prograde (faster than Ω) to retrograde (slower than Ω). These results were obtained via timestepping simulations starting from initial conditions of the form $\cos(M\varphi)$.

Figure 3 shows the qualitative evolution of the rotating waves as Ek is decreased from 10^{-2} to 10^{-5} : the flow is increasingly stronger near the inner sphere and the azimuthal wavenumber increases. These two properties are related. Each half-wave tends to occupy an equal length in radius ($r_{\text{max}} - r_{\text{in}}$) and in circumference ($\sim \pi r_{\text{in}}/M$), leading to $M \approx \pi r_{\text{in}}/2(r_{\text{max}} - r_{\text{in}})$. With decreasing Ek, the radius r_{max} decreases [28] and so M increases. The RW_8 appeared naturally when performing timestepping simulations from an initial condition of the form $\cos(4\varphi)$ at $\text{Ek} = 10^{-4}$, as did the RW_{12} at $\text{Ek} = 3 \times 10^{-5}$. This result was then used as input for a path-following computation to obtain the RW_{12} data at $\text{Ek} = 10^{-5}$. The same techniques were applied to obtain the RW_4 data at $\text{Ek} = 10^{-2}$ starting from the existing results of the same rotating wave at $\text{Ek} = 10^{-3}$.

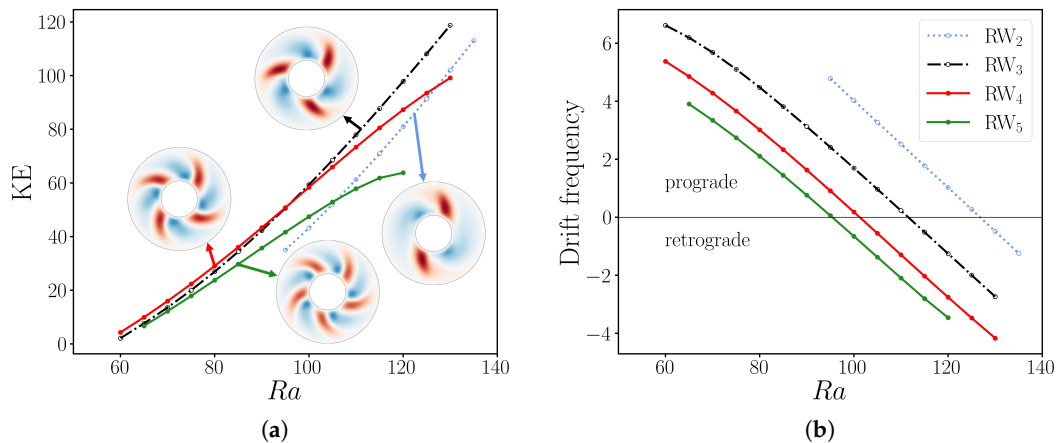


Figure 2. (a) Bifurcation diagram of convection in a rotating spherical annulus for $Ek = 10^{-3}$ using kinetic energy density. Branches RW_2 (blue), RW_3 (black), RW_4 (red) and RW_5 (green) are shown. (b) Drift frequency of rotating waves RW_2 , RW_3 , RW_4 and RW_5 for $Ek = 10^{-3}$. The drift frequencies for each rotating wave decreases from prograde (faster than imposed velocity Ω) to retrograde (slower than Ω) with increasing Ra with the same slope. The resolution used for these simulations was $(N_r, N_\theta, N_\phi \times M) = (46, 72, 128 \times 1)$.

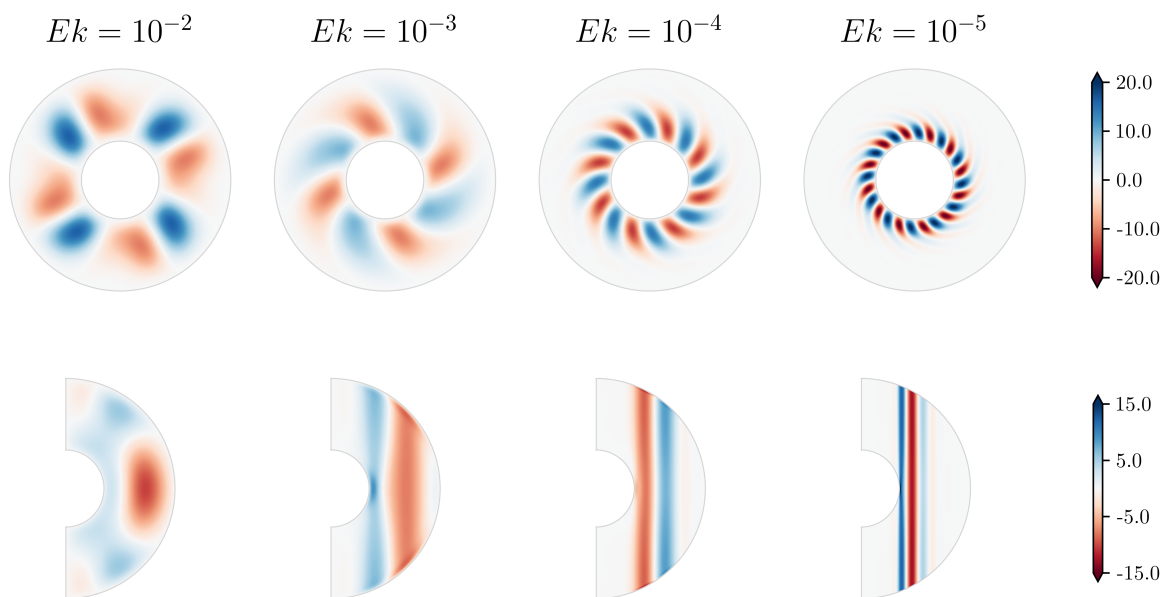


Figure 3. Velocity fields of rotating waves as the Ekman number is reduced. Shown are: RW_4 for $Ek = 10^{-2}$ and $Ra = 140$; RW_4 for $Ek = 10^{-3}$ and $Ra = 77$; RW_8 for $Ek = 10^{-4}$ and $Ra = 88$; and RW_{12} for $Ek = 10^{-5}$ and $Ra = 126$. All of these states have an approximate kinetic energy density of 25. For each field, the radial velocity u_r on the equatorial plane (top) and the azimuthal velocity u_ϕ on a meridional plane (bottom) are shown. As Ek decreases, the convective region becomes more confined to the region of the inner sphere and the flow becomes almost axially independent, according to the Taylor-Proudman theorem. The resolutions used for these simulations were: $(N_r, N_\theta, N_\phi \times M) = (46, 72, 32 \times 4)$ for RW_4 at $Ek = 10^{-2}$ and $Ek = 10^{-3}$; $(60, 100, 28 \times 8)$ for RW_8 ; and $(90, 184, 32 \times 12)$ for RW_{12} .

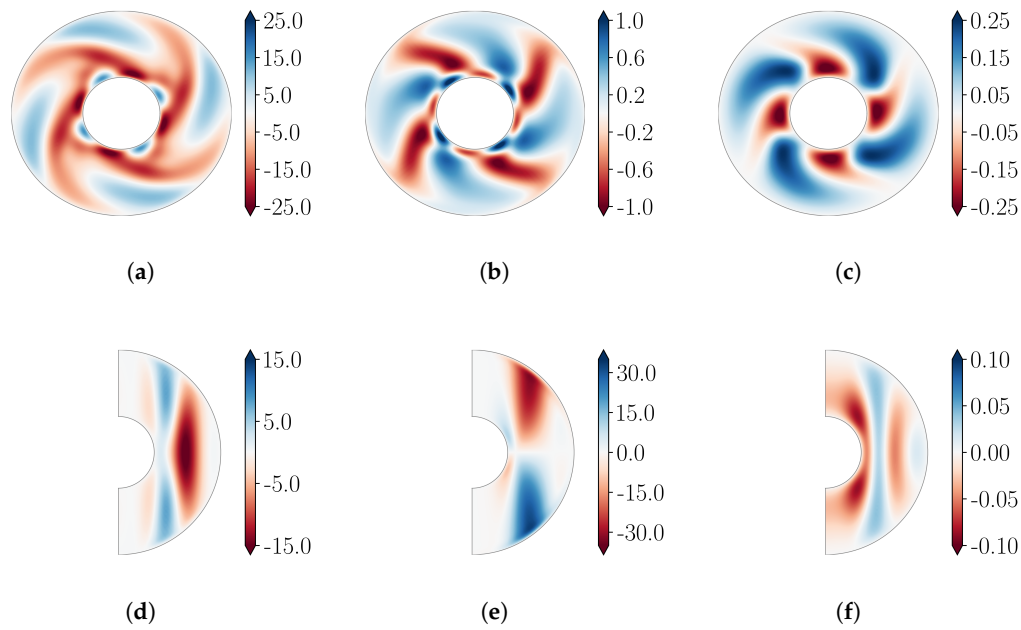


Figure 4. Flow visualizations for a RW_4 at $Ek = 10^{-3}$ and $Ra = 140$ are shown. Top, left to right: U_φ , U_θ and $T - T_{\text{cond}}$ in the equatorial plane. Bottom, left to right: U_r , U_θ and $T - T_{\text{cond}}$ in a meridional plane. The resolution used for this simulation was $(N_r, N_\theta, N_\varphi \times M) = (46, 72, 32 \times 4)$.

3. Numerical Methods

3.1. Spatial Representation

The velocity is decomposed into toroidal and poloidal potentials e and f such that

$$\mathbf{U}(r, \theta, \varphi, t) = \nabla \times e \mathbf{e}_r + \nabla \times \nabla \times f \mathbf{e}_r \quad (6)$$

These and the temperature are expanded in spherical harmonics $Y_l^m(\theta, \varphi)$ and Chebyshev polynomials T_k of the shifted radius $x \equiv (2r - (r_o + r_i))/(r_o - r_i)$.

$$f(r, \theta, \varphi, t) = \sum_{k,l,m} f_{k,l,m}(t) T_k(x) Y_l^m(\theta, \varphi) \quad (7)$$

To carry out transformations to and from the spherical harmonic space, we use the SHTns library [20]. This is a highly efficient code that uses the FFTW library to perform fast transformations to and from Fourier space, and the recursion relations presented in [29] for optimized computation of the associated Legendre polynomials. Although the library allows for multithreaded computations as well as GPU offloading, these capabilities were not used in the present work.

Dealiasing in the r direction is carried out according to the 3/2 rule, whereas in the θ and φ directions, it is performed by SHTns. To achieve this, the library determines a suitable number of physical points based on the non-linear order of the equations to be solved (two in this case) with the aim of maximizing the efficiency of the FFT algorithm. Moreover, for solutions which have azimuthal periodicity $2\pi/M$, we restrict our calculations to a segment of the sphere $0 \leq \varphi < 2\pi/M$, or, equivalently, to those which have Fourier expansions containing only multiples of the wavenumber M ; this is easily implemented using options in SHTns. Lastly, we will denote the physical points in r , θ and φ as N_r , N_θ and $N_\varphi \times M$, respectively. The resolutions used will be given in the captions of each of the figures.

The radial component of the curl and of the double curl of equation (1a) are taken, leading to:

$$\sum_{l,m} \frac{l(l+1)}{r^2} \text{Ek} \left(\frac{\partial}{\partial t} - L_l \right) e_{l,m}(r,t) Y_l^m(\theta, \varphi) = \mathbf{e}_r \cdot \nabla \times \mathbf{F} \quad (8)$$

$$-\sum_{l,m} \frac{l(l+1)}{r^2} \text{Ek} \left(\frac{\partial}{\partial t} - L_l \right) L_l f_{l,m}(r,t) Y_l^m(\theta, \varphi) = \mathbf{e}_r \cdot \nabla \times \nabla \times \mathbf{F} \quad (9)$$

where

$$L_l \equiv \frac{\partial^2}{\partial r^2} - \frac{l(l+1)}{r^2} \quad \text{and} \quad \mathbf{F} \equiv \text{Ek} (\mathbf{U} \cdot \nabla) \mathbf{U} - \nabla P + \frac{Ra}{r_{\text{out}}} T \mathbf{r} - 2\mathbf{e}_z \times \mathbf{U} \quad (10)$$

Equations (8) and (9) are decoupled in ℓ, m and in potential (e vs f), and so its discrete version involves a block-diagonal matrix with 3 N_θ blocks that are each $N_r \times N_r$, as in Figure 5(a).

The temperature field is also expanded in Chebyshev polynomials and spherical harmonics like (7). Its evolution is governed by the discretized form of (1b).

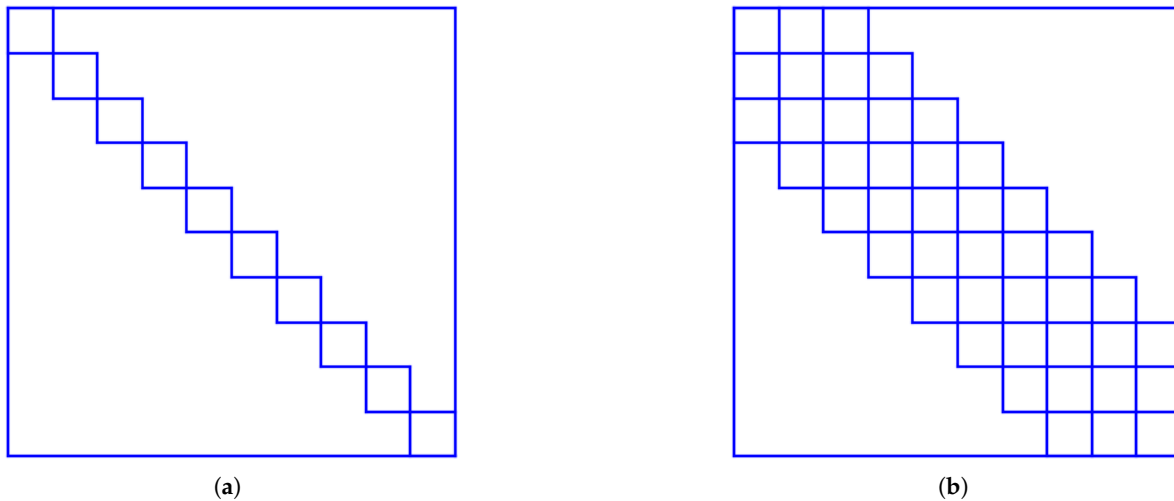


Figure 5. (a) Block-diagonal matrix. (b) Block-pentadiagonal matrix.

3.2. Implicit Coriolis Integration

To carry out implicit integration of the Coriolis term, we include the Coriolis force in the left-hand-side and remove it from \mathbf{F} , leading to:

$$\begin{aligned} & \sum_{m,l} \left[\frac{l(l+1)}{r^2} \left(\frac{\partial}{\partial t} - \text{Ek} L_l \right) - \frac{2im}{r^2} \right] e_{lm}(r,t) Y_l^m(\theta, \varphi) \\ & + \frac{2}{r^2} \left(\frac{l(l+1)}{r} - \frac{\partial}{\partial r} \right) f_{lm}(r,t) \sin \theta \frac{d}{d\theta} Y_l^m(\theta, \varphi) \\ & + 2 \frac{l(l+1)}{r^2} \left(\frac{2}{r} - \frac{\partial}{\partial r} \right) f_{lm}(r,t) \cos \theta Y_l^m(\theta, \varphi) = \mathbf{e}_r \cdot \nabla \times \mathbf{F}^{\text{implicit}} \end{aligned} \quad (11a)$$

$$\begin{aligned} & \sum_{m,l} - \left[\frac{l(l+1)}{r^2} \left(\frac{\partial}{\partial t} - \text{Ek} L_l \right) - \frac{2im}{r^2} \right] L_l f_{lm}(r,t) Y_l^m(\theta, \varphi) \\ & + \frac{2}{r^2} \left(\frac{l(l+1)}{r} - \frac{\partial}{\partial r} \right) e_{lm}(r,t) \sin \theta \frac{d}{d\theta} Y_l^{|m|}(\theta, \varphi) \\ & + 2 \frac{l(l+1)}{r^2} \left(\frac{2}{r} - \frac{\partial}{\partial r} \right) e_{lm}(r,t) \cos \theta Y_l^m(\theta, \varphi) = \mathbf{e}_r \cdot \nabla \times \nabla \times \mathbf{F}^{\text{implicit}} \end{aligned} \quad (11b)$$

where

$$\mathbf{F}^{\text{implicit}} \equiv \text{Ek}(\mathbf{U} \cdot \nabla)\mathbf{U} - \nabla P + \frac{\text{Ra}}{r_{\text{out}}} T \mathbf{r} \quad (12)$$

The recursion relations of the Legendre polynomials

$$\sin \theta \frac{d}{d\theta} P_l^{|m|} = \frac{l(l-|m|+1)}{2l+1} P_{l+1}^{|m|} - \frac{(l+|m|)(l+1)}{2l+1} P_{l-1}^{|m|} \quad (13a)$$

$$\cos \theta P_l^{|m|} = \frac{l-|m|+1}{2l+1} P_{l+1}^{|m|} + \frac{(l+|m|)}{2l+1} P_{l-1}^{|m|} \quad (13b)$$

together with

$$Y_l^m(\theta, \varphi) = N_l^m P_l^m(\cos \theta) e^{im\varphi} \quad N_l^m = \sqrt{(2 - \delta_{m0})(2l+1) \frac{(l-m)!}{(l+m)!}} \quad (14)$$

can be used to transform (11a) and (11b) into:

$$\begin{aligned} \sum_{m,l} Y_l^m(\theta, \varphi) \left\{ \left[\frac{l(l+1)}{r^2} \left(\frac{\partial}{\partial t} - \text{Ek} L_l \right) - \frac{2im}{r^2} \right] e_{lm}(r, t) \right. \\ + \frac{2}{r^2} \frac{(l-1)(l-|m|)}{2l-1} \left[\left(\frac{(l-1)l}{r} - \frac{\partial}{\partial r} \right) + l \left(\frac{2}{r} - \frac{\partial}{\partial r} \right) \right] \frac{N_{l-1}^m}{N_l^m} f_{l-1,m}(r, t) \\ \left. + \frac{2}{r^2} \frac{(l+1+|m|)(l+2)}{2l+3} \left[- \left(\frac{(l+1)(l+2)}{r} - \frac{\partial}{\partial r} \right) + (l+1) \left(\frac{2}{r} - \frac{\partial}{\partial r} \right) \right] \frac{N_{l+1}^m}{N_l^m} f_{l+1,m}(r, t) \right\} \\ = \mathbf{e}_r \cdot \nabla \times \mathbf{F}^{\text{implicit}} \quad (15a) \end{aligned}$$

$$\begin{aligned} \sum_{m,l} Y_l^m(\theta, \varphi) \left\{ - \left[\frac{l(l+1)}{r^2} \left(\frac{\partial}{\partial t} - \text{Ek} L_l \right) - \frac{2im}{r^2} \right] L_l f_{lm}(r, t) \right. \\ + \frac{2}{r^2} \frac{(l-1)(l-|m|)}{2l-1} \left[\left(\frac{(l-1)l}{r} - \frac{\partial}{\partial r} \right) + l \left(\frac{2}{r} - \frac{\partial}{\partial r} \right) \right] \frac{N_{l-1}^m}{N_l^m} e_{l-1,m}(r, t) \\ \left. + \frac{2}{r^2} \frac{(l+1+|m|)(l+2)}{2l+3} \left[- \left(\frac{(l+1)(l+2)}{r} - \frac{\partial}{\partial r} \right) + (l+1) \left(\frac{2}{r} - \frac{\partial}{\partial r} \right) \right] \frac{N_{l+1}^m}{N_l^m} e_{l+1,m}(r, t) \right\} \\ = \mathbf{e}_r \cdot \nabla \times \nabla \times \mathbf{F}^{\text{implicit}} \quad (15b) \end{aligned}$$

In contrast with (8)-(9), equations (15a)-(15b) are coupled in several ways:

- The spectral coefficients e_{lm} and f_{lm} both appear in both equations.
- While each m can be treated independently, components l , $l+1$, and $l-1$ are coupled.
- The real and imaginary parts of e_{lm} and f_{lm} are coupled via the imaginary coefficient $2im/r^2$.

However, two decoupled classes of coefficients appear, one class containing coefficients with odd l for e_l and even l for f_l and a second class with the opposite property. For example the real component of e_{lm} is coupled to its imaginary component and to the real parts of $f_{l\pm 1,m}$. The sums in (15a) and (15b) are represented by block pentadiagonal matrices, as illustrated in Figure 5(b).

3.3. Newton Method

We represent our system of governing equations schematically by

$$\frac{\partial \mathbf{U}}{\partial t} = \mathcal{N}(\mathbf{U}) + \mathcal{L}\mathbf{U} \quad (16)$$

For our problem, \mathbf{U} corresponds to the poloidal and toroidal fields e and f and the temperature T . For explicit or implicit Coriolis integration, $\mathcal{N}(\mathbf{U})$ corresponds to the right-hand-sides of (8)-(9) or of

(15a)-(15b), respectively. $\mathcal{L}\mathbf{U}$ corresponds to minus the left-hand-sides of the same equations with the time derivative omitted. (The correspondence is imperfect, since $\partial/\partial t$ does not act directly on \mathbf{U} .)

We represent our timestepping code schematically by the following implicit-explicit Euler method:

$$\begin{aligned} \mathbf{U}(t + \Delta t) &= \mathbf{U}(t) + \Delta t(\mathcal{N}(\mathbf{U}(t)) + \mathcal{L}\mathbf{U}(t + \Delta t)) \\ \implies \mathbf{U}(t + \Delta t) &= \mathcal{B}(\mathbf{U}(t)) \equiv (I - \Delta t\mathcal{L})^{-1}(I + \Delta t\mathcal{N})\mathbf{U}(t) \end{aligned} \quad (17)$$

where \mathcal{L} is linear (though \mathcal{N} need not be nonlinear). Steady states are solutions to

$$0 = \mathcal{A}(\mathbf{U}) \equiv \mathcal{N}(\mathbf{U}) + \mathcal{L}\mathbf{U} \quad (18)$$

or, alternatively

$$0 = \mathbf{U}(t + \Delta t) - \mathbf{U}(t) \quad (19)$$

where \mathbf{U} is a solution to the continuous-time differential equation (16). Surprisingly, (19) is also a criterion for stationarity when $\mathbf{U}(t + \Delta t)$ is calculated via the time-stepping scheme (17) for any value of Δt , as can be seen from

$$\begin{aligned} (\mathcal{B} - I)(\mathbf{U}(t)) &= \mathbf{U}(t + \Delta t) - \mathbf{U}(t) = (I - \Delta t\mathcal{L})^{-1}(I + \Delta t\mathcal{N})\mathbf{U}(t) - \mathbf{U}(t) \\ &= (I - \Delta t\mathcal{L})^{-1}(I + \Delta t\mathcal{N} - (I - \Delta t\mathcal{L}))\mathbf{U}(t) \\ &= \underbrace{(I - \Delta t\mathcal{L})^{-1}\Delta t}_{\mathcal{P}(\Delta t)} \underbrace{(\mathcal{N} + \mathcal{L})}_{\mathcal{A}} \mathbf{U}(t) \end{aligned} \quad (20)$$

Thus $\mathcal{B} - I = \mathcal{P}(\Delta t)\mathcal{A}$ and so roots of \mathcal{A} of (18) can be found by computing roots of $\mathcal{B} - I$, which is simply the difference between two large consecutive timesteps. Note that (20) does not use Taylor series; Δt can be of any size. Indeed, for Δt large, $\mathcal{P}(\Delta t) \rightarrow -\mathcal{L}^{-1}$ acts as a preconditioner (approximate inverse) for \mathcal{A} , whose poor conditioning is due to that of \mathcal{L} . Typically, for Newton solving we use $\Delta t = 200$.

Finding the roots of (20) via Newton's method requires the solution of the linear system

$$(\mathcal{B}_{\mathbf{U}} - I)\mathbf{u} = (\mathcal{B} - I)\mathbf{U} \quad (21)$$

where \mathbf{U} is the current estimate and $\mathcal{B}_{\mathbf{U}}$ is the Jacobian of \mathcal{B} evaluated at \mathbf{U} , with $\mathcal{N}_{\mathbf{U}}$ substituted for \mathcal{N} in (17). For example, if $\mathcal{N}(\mathbf{U})$ contains $(\mathbf{U} \cdot \nabla)\mathbf{U}$, this term becomes $(\mathbf{U} \cdot \nabla)\mathbf{u} + (\mathbf{u} \cdot \nabla)\mathbf{U}$ in $\mathcal{N}_{\mathbf{U}}\mathbf{u}$, as well as in the operators \mathcal{A} and \mathcal{B} .

We solve (21) using a Krylov method such as GMRES [30]. In such matrix-free methods, one need only provide the right-hand-side $(\mathcal{B} - I)\mathbf{U}$ for the current estimate, and a routine which carries out the *action* of $(\mathcal{B}_{\mathbf{U}} - I)$ on any vector \mathbf{u} . The computational cost of a Newton step can be measured by the number of such actions (i.e. GMRES iterations) taken by the Krylov method, since the right-hand-side remains constant throughout the step. The number of GMRES iterations required is low if $(\mathcal{B}_{\mathbf{U}} - I) = \mathcal{P}(\Delta t)\mathcal{A}$ is well-conditioned; it is for this reason that we take Δt large. Once the decrement \mathbf{u} is determined by solving (21), it is subtracted from \mathbf{U} to form an improved estimate. We accept \mathbf{u} as the decrement if the linear system (21) is solved by GMRES to relative accuracy 10^{-10} . For Newton's method, we accept \mathbf{U} as a steady state, i.e. a solution to (19), if $\|(\mathcal{B} - I)\mathbf{U}\| < 10^{-7}$.

This method [11–13,25,31,32] has been called Stokes preconditioning because the linear operator \mathcal{L} which is integrated implicitly in (17) is usually the viscous term that occurs in Stokes equation. However, it is beneficial to include as many possible other terms in \mathcal{L} , as long as they are linear and can be inverted directly. It is for the reason that we have included the Coriolis force in \mathcal{L} , as described in section 3.2. In section 5, we will compare the cost of computations using Newton's method with

implicit vs. explicit Coriolis integration for various values of the Ekman number by comparing the number of GMRES iterations necessary for each.

3.4. Traveling Waves

Newton's method can also be used to compute traveling waves in the same way. Azimuthal rotating waves satisfy $\mathbf{U}(\varphi, t) = \tilde{\mathbf{U}}(\varphi - Ct) = \tilde{\mathbf{U}}(\tilde{\varphi})$, where C is an unknown wavespeed. Thus, $\partial_t \mathbf{U} = -C \partial_{\tilde{\varphi}} \tilde{\mathbf{U}}$. Substituting into (16) and dropping the tildes leads to

$$0 = \mathcal{N}(\mathbf{U}) + C \partial_{\varphi} \mathbf{U} + \mathcal{L} \mathbf{U} \quad (22)$$

The explicit portion of the timestep is augmented to include $C \partial_{\varphi}$ as well as \mathcal{N} :

$$\mathbf{U}(t + \Delta t) = (I - \Delta t \mathcal{L})^{-1} (I + \Delta t (\mathcal{N} + C \partial_{\varphi})) \mathbf{U}(t) \quad (23)$$

If \mathbf{U} is expressed in terms of spherical harmonics in which the φ dependence is trigonometric, the action of $C \partial_{\varphi}$ on a Fourier component \mathbf{U}_m is merely $C \partial_{\varphi} \mathbf{U}_m = C i m \mathbf{U}_m$.

To use Newton's method to determine the unknown field \mathbf{U} and wavespeed C , we substitute $\mathbf{U} \rightarrow \mathbf{U} - \mathbf{u}$ and $C \rightarrow C - c$ into (22):

$$0 = \mathcal{N}(\mathbf{U} - \mathbf{u})(C - c) + \partial_{\varphi}(\mathbf{U} - \mathbf{u}) + \mathcal{L}(\mathbf{U} - \mathbf{u}) \quad (24)$$

Expanding around (\mathbf{U}, C) and truncating at first order leads to the linear system that must be solved for the decrements (\mathbf{u}, c) :

$$(\mathcal{N}_{\mathbf{U}} + C \partial_{\varphi} + \mathcal{L}) \mathbf{u} + c \partial_{\varphi} \mathbf{U} = (\mathcal{N} + C \partial_{\varphi} + \mathcal{L})(\mathbf{U}) \quad (25)$$

The preconditioner remains $\mathcal{P}(\Delta t) = (I - \Delta t \mathcal{L})^{-1} \Delta t$ with large Δt , since \mathcal{L} continues to be responsible for the large condition number of (25).

$$\mathcal{P}(\Delta t) [(\mathcal{N}_{\mathbf{U}} + C \partial_{\varphi} + \mathcal{L}) \mathbf{u} + c \partial_{\varphi} \mathbf{U}] = \mathcal{P}(\Delta t) (\mathcal{N} + C \partial_{\varphi} + \mathcal{L})(\mathbf{U}). \quad (26)$$

An additional equation must be added to the system to compensate for the additional unknown. We choose a phase condition, more specifically we require the imaginary part of a single component (temperature, toroidal or poloidal field; radial and angular and azimuthal mode) of \mathbf{u} be zero, i.e. that the corresponding *phase component* (whose index we shall call J) of \mathbf{U} remains unchanged.

This simple choice suggests a trick for retaining the size of the unknown \mathbf{u} rather than using augmented fields, since u_J is no longer an unknown. The routine which acts on (\mathbf{u}, c) is defined such that c is stored in u_J . At the beginning of an action, c is extracted, u_J is set to zero, and the explicit part of the action on \mathbf{u} is carried out. Then we multiply the stored value of $\partial_{\varphi} \mathbf{U}$ by c and add the result to the explicit part of the action. When the Krylov method converges and returns \mathbf{u} , we must again extract c from u_J , after which u_J is set to zero. Effectively, although C and c have been added to the unknowns for the Newton method and for the linear equation, u_J and u_J are no longer unknowns.

3.5. Continuation

We compute solutions along a branch via Newton's method as described above. We do not impose additional equations, such as requiring a new solution along the branch to be a certain distance from a prior solution or that the increment along the branch must be perpendicular to some direction. Therefore the only additional ingredients that must be introduced to discuss continuation are the choice of initial estimate for each solution along the branch and the parameters that are prescribed for the Newton iteration.

If the continuation is in Rayleigh number, we increment or decrement this parameter according to the number of Newton iterations required for convergence in the previous step as follows [33]:

$$Ra^{(i+1)} = Ra^{(i)} + \Delta Ra = Ra^{(i)} + \alpha(Ra^{(i)} - Ra^{(i-1)}) \quad (27)$$

where $Ra^{(i)}$, $Ra^{(i-1)}$ denote the Rayleigh numbers used in the two previous continuation steps, ΔRa is the increment or decrement, and α is defined by:

$$\alpha = \frac{N^{\text{opt}} + 1}{N^{(i)} + 1} \quad (28)$$

N^{opt} is the target number of Newton iterations. If $N^{(i)} = N^{\text{opt}}$, then ΔRa remains unchanged, whereas it ΔRa is reduced (increased) if $N^{(i)} > (<)N^{\text{opt}}$. The choice of N^{opt} is guided by two considerations. The first is the level of sampling desired along the branch. The second is economy: a smaller value of α will engender more values of Ra , but each calculation will be faster. For our computations we fixed N^{opt} to be between 4 and 6.

To choose an initial estimate \mathbf{U} for the next solution along a branch at $Ra^{(i+1)}$, first-order and even zero-th order extrapolation (i.e. using just the previously computed field) can be used to follow a smooth and monotonically varying branch. However, quadratic extrapolation is essential for going around turning points. Lagrange interpolation uses the three previous Rayleigh numbers $Ra^{(i)}$, $Ra^{(i-1)}$, $Ra^{(i-2)}$ and the new Rayleigh number $Ra^{(i+1)}$ from (27) to determine coefficients a, b, c such that

$$Ra^{(i+1)} = aRa^{(i)} + bRa^{(i-1)} + cRa^{(i-2)} \quad (29)$$

after which we set

$$\mathbf{U}^{(i+1)} = a\mathbf{U}^{(i)} + b\mathbf{U}^{(i-1)} + c\mathbf{U}^{(i-2)} \quad (30)$$

to form a new estimate of the solution. Newton iterations are then used to refine \mathbf{U} until the norm of (22) is 10^{-7} . This procedure requires saving the three previous solution vectors and Rayleigh numbers.

3.6. Turning Points

We now turn to the more complicated matter of extrapolating near saddle-node bifurcations, at which \mathbf{U} ceases to be a single-valued function of Ra . The code detects that the current step is in the vicinity of a turning point by comparing relative changes between the solution vector \mathbf{U} , and the Rayleigh number, i.e. we compare

$$\left| \frac{\Delta U_J}{U_J} \right| \equiv \left| \frac{U_J^{(i)} - U_J^{(i-1)}}{U_J^{(i)}} \right| \quad \text{with} \quad \gamma \left| \frac{\Delta Ra}{Ra} \right| \equiv \gamma \left| \frac{Ra^{(i)} - Ra^{(i-1)}}{Ra^{(i)}} \right| \quad (31)$$

where J is the index denoting the element of the solution vector $U^{(i)}$ of highest absolute value and γ is a constant factor that makes the two magnitudes comparable. For our computations we fixed this constant to be between 10 and 100.

Since the normal form of a saddle-node bifurcation at $(\mu, x) = (0, 0)$ is $\dot{x} = \mu - x^2$, dependent variables (x, \mathbf{U}) vary like the square root of the control parameter $(\mu, Ra - Ra_c)$ near a turning point. Consequently, as the current solution approaches a turning point, the relative difference $|\Delta U_J / U_J|$ increases until it is greater than that of $\gamma |\Delta Ra / Ra|$. It is at this stage that the code switches to extrapolating quadratically, using the component U_J as the independent variable instead of the Rayleigh number. This allows ΔRa to change sign and the continuation to go around the turning point.

We replace (27) and (29) by analogous equations in \mathbf{U}_j to determine $\mathbf{U}_j^{(i+1)}$ and a, b, c , and then replace (29) and (30) by

$$\mathbf{U}_j^{(i+1)} = a\mathbf{U}_j^{(i)} + b\mathbf{U}_j^{(i-1)} + c\mathbf{U}_j^{(i-2)} \text{ for } j \neq J, \text{ and} \quad (32)$$

$$Ra^{(i+1)} = aRa^{(i)} + bRa^{(i-1)} + cRa^{(i-2)} \quad (33)$$

to set the estimate of the new solution. Note that we have only changed the method of determining the initial guess. Ra remains fixed to $Ra^{(i+1)}$ during the Newton step, while all the elements of \mathbf{U} are allowed to vary.

Continuation follows in this manner until eventually $|\Delta\mathbf{U}_j/\mathbf{U}_j|$ exceeds $\gamma|\Delta Ra/Ra|$, at which point the code switches back to using the Rayleigh number as the independent variable. The parameter γ can be obtained by analyzing the behaviour of the continuation near a turning point. Indeed, if γ is too large, then the code will continue to use Ra as the independent variable past the point at which there is no solution for the next continuation step, and Newton's method will not converge. Should this situation arise, the continuation can be restarted at the last converged solution using a suitably reduced value of γ .

4. Branch Following

4.1. Continuation in Rayleigh Number

We used the continuation code to compute branches of rotating waves. As previously explained, to search for solutions having an M azimuthal periodicity we perform our computations to a fraction of the domain given by $2\pi/M$. The solutions are computed independently of their stability; most of the solutions we compute are unstable, either within the restricted domain $0 \leq \varphi < 2\pi/M$, or to instabilities which break this restriction such as rotating waves with a different M .

Fixing the Ekman number, we carried out continuation in Rayleigh number Ra . Most of the branches that we computed showed no unusual features, varying smoothly and monotonically down to their threshold at a supercritical Hopf bifurcation. However, a few branches presented some interesting non-monotonic behavior, which we show in order to display the capacities of our code. Figure 6 tracks the Rayleigh-number dependence of the RW_4 branch for $Ek = 3.53 \times 10^{-5}$. This branch consists of three smooth regions that are separated from one another by short intervals which each contain two saddle-node bifurcations and rapid changes in drift frequency. In order to focus in on these intervals, we have computed branches at slightly higher and lower Ekman numbers in Figures 7 and 8. Each of these branches contains a single interval of rapid change. We also computed an RW_8 branch for $Ek = 1.26 \times 10^{-5}$, shown in Figure 9. Indeed, for this value of Ek , rotating waves with azimuthal wavenumber 8 are more appropriate, i.e. more likely to be stable. This branch also contains a plateau in drift frequency adjacent to a short interval at $Ra \approx 159$ of rapid change delimited by two saddle-node bifurcations. These rapid rises and plateaus in drift frequency should have a physical or phenomenological explanation, but we have not been able to find one.

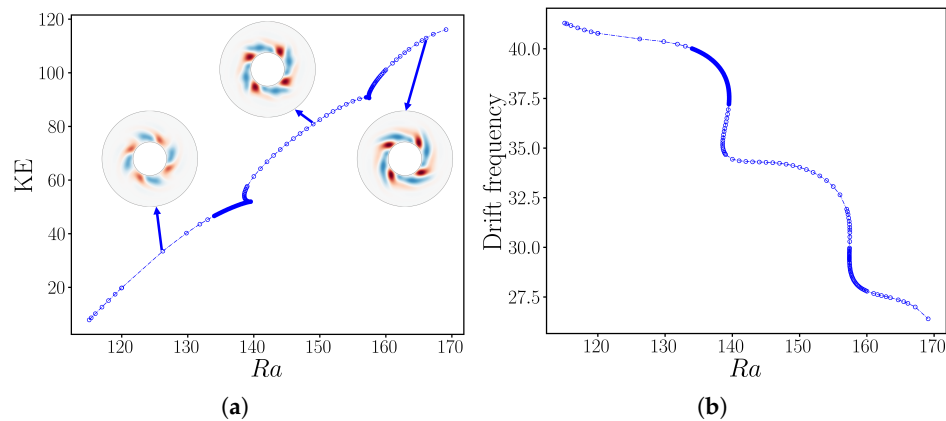


Figure 6. Bifurcation diagram of rotating wave RW_4 for $Ek = 3.53 \times 10^{-5}$ as a function of Ra . (a) Kinetic energy. The branch contains three smooth regions separated by two short intervals (at $Ra \approx 140$ and $Ra \approx 158$) of rapid change containing saddle-node bifurcations. (b) Drift frequency. Three long intervals of almost constant drift frequency are separated by two short intervals of rapid change. The resolution used for this computation was $(N_r, N_\theta, N_\varphi \times M) = (60, 80, 40 \times 4)$.

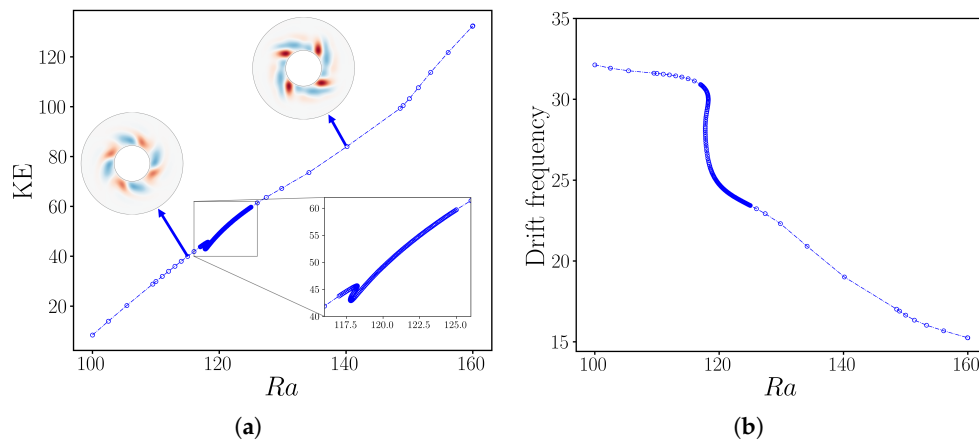


Figure 7. Bifurcation diagram of rotating wave RW_4 for $Ek = 5.6 \times 10^{-5}$ as a function of Ra . (a) Kinetic energy. (b) Drift frequency. The branch contains a single interval of rapid change at $Ra \approx 118$. The resolution used for this computation was $(N_r, N_\theta, N_\varphi \times M) = (46, 72, 32 \times 4)$.

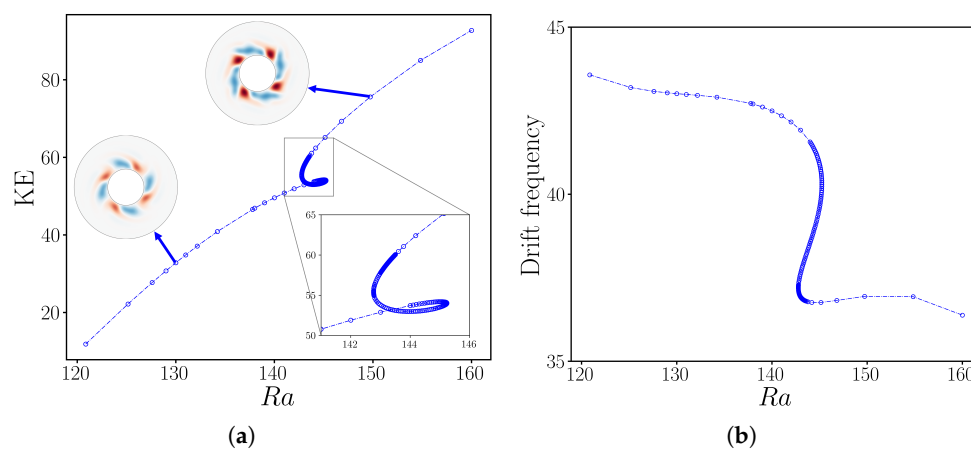


Figure 8. Bifurcation diagram of rotating wave RW_4 for $Ek = 3.16 \times 10^{-5}$ as a function of Ra . (a) Kinetic energy. (b) Drift frequency. The branch contains a single interval of rapid change at $Ra \approx 144$. The resolution used for this computation was $(N_r, N_\theta, N_\varphi \times M) = (60, 80, 40 \times 4)$.

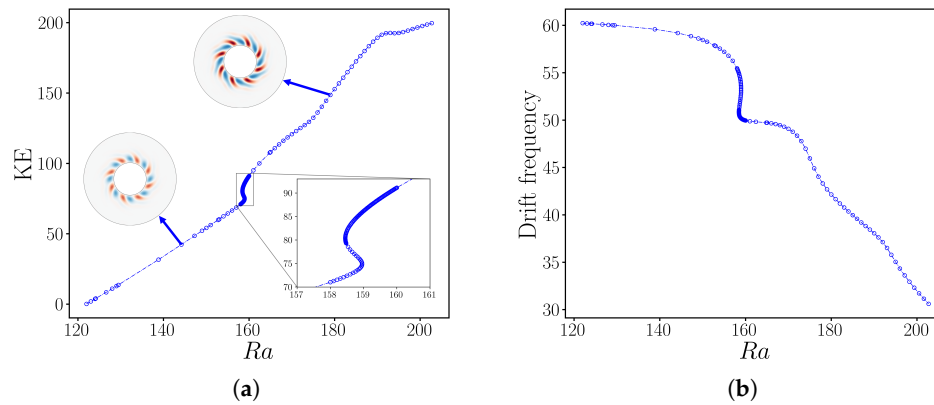


Figure 9. Bifurcation diagram of rotating wave RW_8 for $Ek = 1.26 \times 10^{-5}$ as a function of Ra . (a) Kinetic energy. (b) Drift frequency. The branch contains a single interval of rapid change at $Ra \approx 159$. The resolution used for this computation was $(N_r, N_\theta, N_\varphi \times M) = (68, 100, 28 \times 8)$.

4.2. Continuation in Ekman Number and in Resolution

We adapted our method to follow branches in Ekman number, varying Ek on a logarithmic scale. We replace Ra with $\log_{10}(Ek)$ in (27) as follows:

$$\begin{aligned} \log_{10}(Ek^{(i+1)}) &= \log_{10}(Ek^{(i)}) + \Delta[\log_{10}(Ek)] \\ &= \log_{10}(Ek^{(i)}) + \alpha(\log_{10}(Ek^{(i)}) - \log_{10}(Ek^{(i-1)})) \end{aligned} \quad (34)$$

where the computation of α remains as in equation (28). In this case, we used $N^{\text{opt}} = 3$ for all of the simulations. Furthermore, in order to go around turning points, we now compare $|\Delta \mathbf{U}_j / \mathbf{U}_j|$ and $\gamma |\Delta[\log_{10}(Ek)] / \log_{10}(Ek)|$, where we chose γ between 400 and 500.

Apart from these minor changes, there are several major differences between performing continuation in Ekman and Rayleigh numbers. The first is that the matrices that represent the left-hand-sides of (8)-(9) or (11a)-(11b) must be recomputed every time a new value of Ek is chosen. Indeed, while the Rayleigh number only appears in the right hand side of these equations, Ek is also present in the diffusion terms, which are treated implicitly regardless of how the Coriolis term is treated. This means that extra work must be done at the beginning of every continuation step.

Secondly, as the Ekman number is decreased, the critical Rayleigh number for onset of convection increases. That is, rotation stabilizes the configuration against convection. A classic result is that the critical value of the usual thermal Rayleigh number Ra_{therm} varies like $Ek^{-4/3}$, so that the critical value of our Rayleigh number $Ra_{\text{rot}} = Ra_{\text{therm}} Ek / Pr$ varies like $Ek^{-1/3} / Pr$. Therefore, instead of keeping Ra fixed as we vary Ek , we have kept $Ra Ek^{1/3}$ fixed by increasing Ra every time we decrease Ek .

Third, as the simulation ventures towards lower Ekman numbers, the fields require a higher resolution. This is clearly seen from the visualizations in Figure 3, which show that the radial extent and azimuthal wavelength decrease with Ek . To achieve this, grid refinement is introduced in the code such that the spectral resolution is increased by 20% whenever under-resolution is detected in the Chebyshev or the spherical harmonic modes. This is done by introducing a threshold for the amplitude ratio between the mode of highest absolute value and of highest wavenumber. When this ratio exceeds the threshold, the code calls a grid refinement subroutine which first deallocates the previous grid, then creates a new one using the new resolution and finally represents the fields from the last continuation step on this new grid by filling in the new modes with zero. This last step is then recomputed with the new resolution and path following continues as intended. In our computations we chose the under-resolution threshold to be between 10^{-6} and 10^{-7} .

Figure 10 shows the result of continuation in Ek . Because the convective threshold obeys a scaling law $Ra_{\text{therm}}^{\text{crit}} \sim Ek^{-4/3}$ [2], we fixed Ra to be proportional to $Ek^{-1/3}$ so as to remain at approximately

the same distance from the convective threshold. This branch showed three coexisting solutions over the range $4 \times 10^{-4} \lesssim Ek \lesssim 6 \times 10^{-5}$. Our grid refinement algorithm increased the resolution in r by about 20% and by 50% in θ and φ . We emphasize that most continuations showed monotonic behavior and we have chosen deliberately to present those that did not.

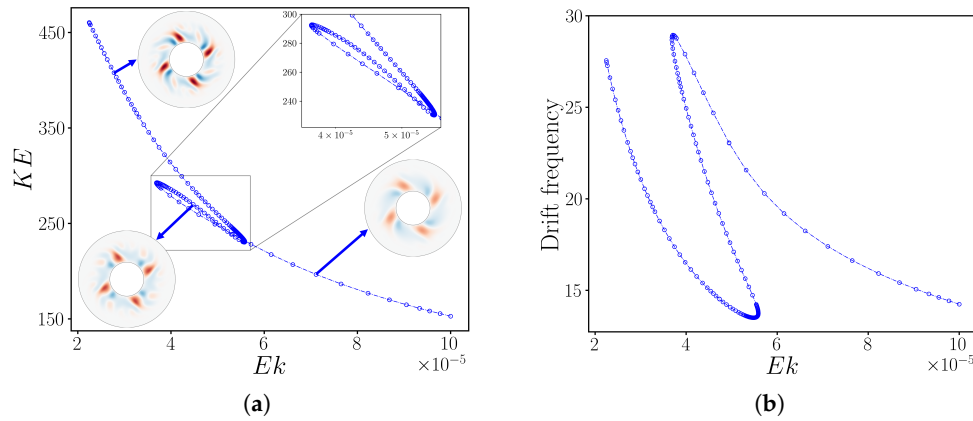


Figure 10. Bifurcation diagram of rotating wave RW_4 as a function of Ek , while fixing $Ra = 6.5 \times Ek^{-1/3}$. (a) Kinetic energy. (b) Drift frequency. Three solution branches exist over the range $4 \times 10^{-5} \lesssim Ek \lesssim 6 \times 10^{-5}$. The resolutions used for this computation ranged from $(N_r, N_\theta, N_\varphi \times M) = (68, 92, 48 \times 4)$ to $(80, 136, 70 \times 4)$.

5. Timing Comparisons

Newton's method is very fast, typically requiring 3-10 iterations. The bottleneck in applying Newton's method to systems of partial differential equations is always the solution of the linear systems. In our case, this is quantified by the total number of GMRES iterations necessary to compute a new state along the branch. We find that the number of iterations depends on the Ekman number, but is fairly independent of the Rayleigh number. We therefore average the number of GMRES iterations over all the points computed in a branch for a fixed Ekman number. We do this for the Newton method with both implicit and explicit Coriolis to produce the curves in Figures 11, 12, and 13.

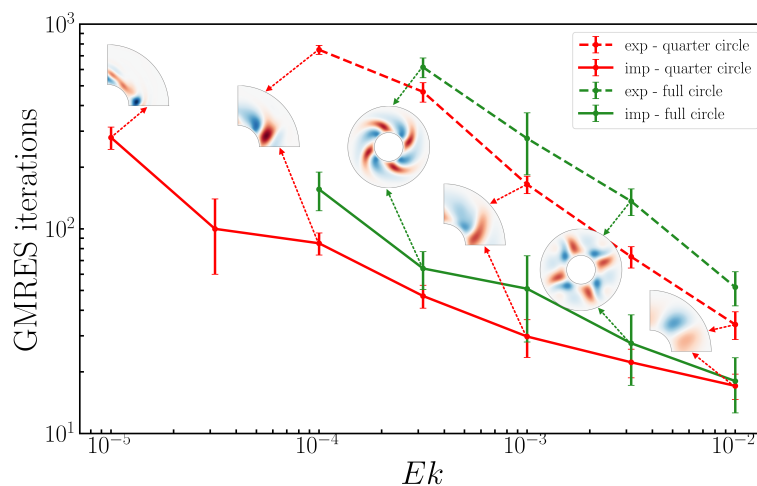


Figure 11. Total number of matrix-vector actions required by nested Newton-GMRES algorithm to compute RW_4 as a function of Ek with explicit (dashed) and implicit (solid) implementation of Coriolis force. An average is taken over a branch of Ra values. The number of actions required by the explicit algorithm is always greater than that required by the implicit algorithm, with the ratio between them increasing from approximately 2 at $Ek = 10^{-2}$ to approximately 9 at $Ek = 10^{-4}$. Using the full domain $0 \leq \varphi < 2\pi$ (green) instead of the reduced domain $0 \leq \varphi < \pi/2$ (red) increases the number of GMRES iterations, as well as the cost of each iteration. In average, the full domain requires 1.5 times more actions than the reduced domain counterpart. The explicit algorithm is unsuccessful for $Ek < 10^{-4}$. The resolutions used in this case were $(N_r, N_\theta, N_\varphi \times M) = (46, 72, 128 \times 1)$ for the full circle computations and from $(46, 72, 32 \times 4)$ to $(76, 80, 40 \times 4)$ for the quarter circle ones.

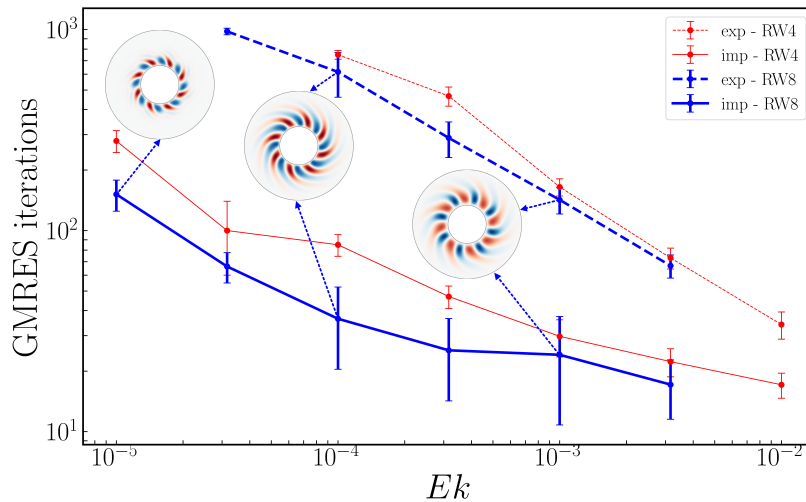


Figure 12. Total number of matrix-vector actions required by nested Newton-GMRES algorithm to compute RW₄ (red) and RW₈ (blue) as a function of Ek with explicit (dashed) and implicit (solid) implementation of Coriolis force. For clarity, the insets show the full domain, although the computations are carried out in reduced domains $0 \leq \varphi < 2\pi/M$. An average is taken over many values of Ra . The implicit algorithm is always faster than the explicit algorithm, with the ratio between them of GMRES iterations required increasing from 4 at $Ek = 3 \times 10^{-3}$ to 15 at $Ek = 3 \times 10^{-5}$, below which the explicit algorithm fails. The resolutions used for the RW₈ computations ranged from $(N_r, N_\theta, N_\varphi \times M) = (64, 100, 28 \times 8)$ to $(76, 124, 32 \times 8)$.

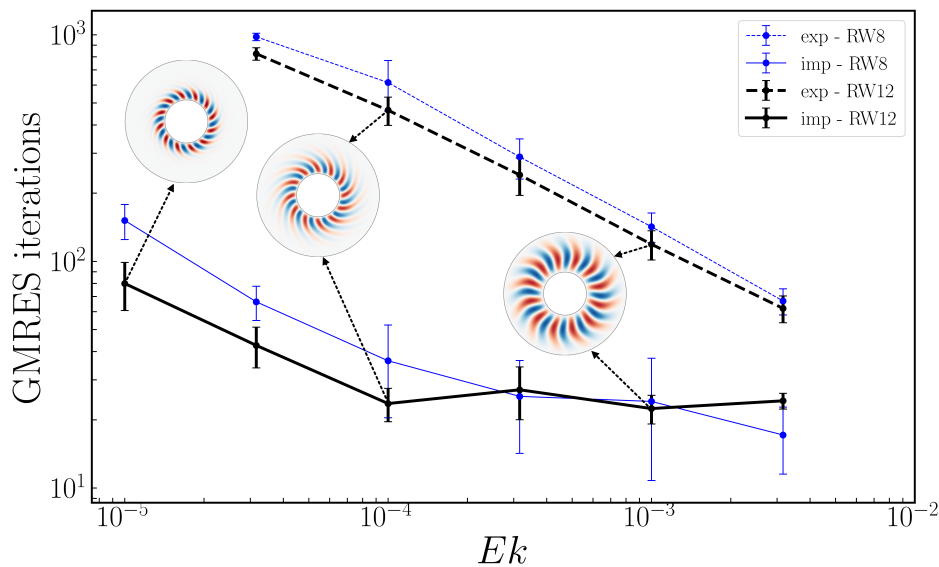


Figure 13. Total number of matrix-vector actions required by nested Newton-GMRES algorithm to compute RW₁₂ (black) and RW₈ (blue) as a function of Ek with explicit (dashed) and implicit (solid) implementation of Coriolis force. For clarity, the insets show the full domain, although the computations are carried out in reduced domains $0 \leq \varphi < 2\pi/M$. An average is taken over a branch of Ra values. The implicit algorithm is always faster than the explicit algorithm for RW₁₂, increasing from a factor of 2.5 for $Ek = 3 \times 10^{-3}$ to 20 for $Ek = 3 \times 10^{-5}$, below which the explicit algorithm fails. The resolutions used for the RW₈ computations ranged from $(N_r, N_\theta, N_\varphi \times M) = (60, 112, 20 \times 12)$ to $(90, 184, 32 \times 12)$.

Measuring the economy realized in timestepping is more problematic. Because explicit integration effectively approximates a decaying exponential by a polynomial, it displays artificial temporal divergence if the timestep is too large. Because implicit integration instead approximates the exponential by a decaying rational function, the timestep is not constrained by stability. The timestep is, however, still constrained by accuracy. A particularly demanding criterion for judging accuracy is the wavespeed [34]. We compute the relative errors in wavespeeds $C_{\text{error}} \equiv (C_{\Delta t} - C_{\text{exact}})/C_{\text{exact}}$, where C_{exact} is

obtained from Newton's method and hence has no timestepping error. Figure 14 presents the relative errors as a function of Ek and Δt , with the other parameters set to the values below the figure. The errors obtained from the explicit and implicit methods are the same for the same value of Δt . For each Ek , there is a minimum Δt above which the explicit simulation diverges in time, but the implicit simulation still yields a solution with an error that is less than 5%. This value of Δt is indicated in Figure 14 by the left endpoint of an arrow for each Ek number. But eventually, Δt is so large that the result of the implicit method can even be qualitatively incorrect. As an upper limit for the implicit method, we choose that the relative error on C_{error} must be less than 5%. The limiting values of large Δt are indicated as the right endpoints of the arrows in Figure 14 and in Table 1. Hence, for each Ek , there is a range of Δt -values that can be used only for implicit timestepping and whose error does not exceed 5%. These ranges are given in table 1. Although these results are specific to the parameter values and flows that we have simulated, they suggest that implicit Coriolis integration can be orders of magnitude faster, enabling the simulation of convection in a spherical annulus at low Ekman numbers.

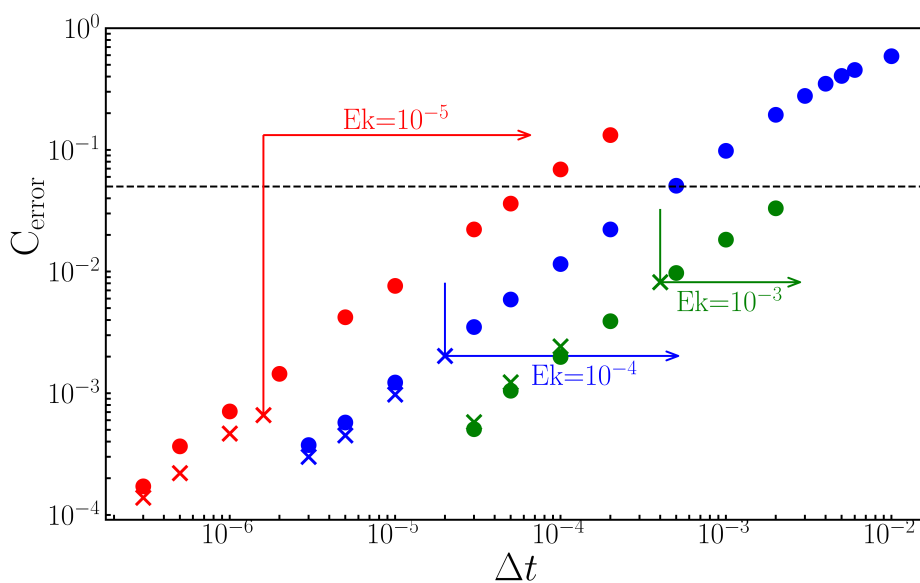


Figure 14. Accuracy of time-dependent calculation of rotating waves RW_M as a function of timestep Δt using implicit (circles) and explicit (crosses) time-stepping of the Coriolis term. Shown is the relative error $C_{error} \equiv |(C_{\Delta t} - C_{exact})/C_{exact}|$ of the drift velocity as a function of the timestep for various sets of parameter values (see table) where C_{exact} is calculated via Newton's method. Both the explicit and implicit methods are second order in time, with $C_{error} \sim \Delta t^2$. Vertical lines indicate the limiting Δt above which explicit time-stepping diverges and cannot be used. The horizontal dashed line indicates $C_{error} = 5\%$. The arrows designate the range of Δt for which implicit timestepping is advantageous: it does not diverge and C_{error} remains less than 5%. For $Ek = 10^{-3}$ and 10^{-5} , integration using Δt beyond the last point plotted yields qualitatively incorrect states. See also Table 1

Table 1. Parameters used for testing and useful range of Δt for implicit time-stepping

Ek	Ra	M	$(N_r, N_\theta, N_\phi \times M)$	C_{exact}	Δt		
					explicit diverges	5% error	ratio
10^{-3}	120	4	(46, 72, 32×4)	-2.7647	4.0×10^{-4}	2.8×10^{-3}	70
10^{-4}	130	8	(60, 80, 40×4)	3.9998	2.0×10^{-5}	5.1×10^{-4}	255
10^{-5}	140	12	(76, 136, 24×4)	51.2817	1.6×10^{-6}	6.5×10^{-5}	406

6. Discussion

We have developed a pair of codes for simulating thermal convection in a rotating spherical fluid shell that rely on an implicit treatment of the Coriolis force. The numerical cost of this improvement is quite manageable: the block diagonal matrix systems which arise from implicit treatment of the diffusive terms must be replaced by block pentadiagonal matrix systems, which can still be solved

by block banded LU decomposition and backsolving. Once implemented in a timestepping code, implicit integration with a very large pseudo-timestep ($\Delta t = 200$) can be leveraged to precondition the large linear systems that are the core of Newton's method. When only the diffusive terms are treated implicitly, this is known as Stokes preconditioning [11–13,25,31,32]; we could call the method developed here Stokes-and-Coriolis preconditioning.

We have demonstrated this method's capabilities by carrying out continuation in Rayleigh number at various values of the Ekman number on the order of 10^{-5} for rotating waves with azimuthal wavenumbers 4, 8, and 12. We have found several intriguing examples of such branches contain plateaus in drift frequency, separated by intervals of rapid change delimited by pairs of saddle-node bifurcations. The physical, or at least phenomenological, reasons for these properties remain to be discovered. We have also implemented continuation in Ekman number, spaced logarithmically, in which we automatically measure and increase the resolution as needed. We are unaware of any previous examples of continuation in Ekman number or with automatic resolution adjustment in the literature.

We have measured the economy that is realized by implicit treatment of the Coriolis force. For Newton's method, the advantage of the implicit over the explicit treatment is dramatic. For example, the explicit algorithm takes 20 times as many GMRES iterations (and hence 20 times as much CPU time) as the implicit method to compute states on the RW_{12} branch at $Ek = 3 \times 10^{-5}$. For lower Ekman numbers, the explicit method takes an unmanageable number of GMRES iterations, making it impossible to use in practice.

For timestepping, the advantage of the implicit over the explicit treatment is equally spectacular. For computing rotating waves at $Ek = 10^{-5}$, the implicit algorithm can reasonably use timesteps that are 200 times past the temporal stability limit, meaning that simulations in this regime can be 200 times faster than if an explicit method were used.

We now describe the differences between Stokes preconditioning and the method used in [21–24] for convection in rotating spheres, as well as by [35–38] for other hydrodynamic problems. These authors seek steady states or traveling waves as roots of $\mathbf{U}(t+T) - \mathbf{U}(t)$, where $\mathbf{U}(t+T)$ is computed from $\mathbf{U}(t)$ in the usual manner by carrying out many timesteps, each with a small Δt ; we call this the integration method. In contrast, although equations (19)-(20) would seem to imply that we too seek roots of $\mathbf{U}(t+T) - \mathbf{U}(t)$, this is not at all the case. In the Stokes preconditioning method, we compute $\mathbf{U}(t+T)$ from $\mathbf{U}(t)$ by carrying out a single very large implicit-explicit Euler timestep (17), with a timestep $\Delta t = T$ that is so large that the difference $\mathbf{U}(t+\Delta t) - \mathbf{U}(t)$ no longer approximates the time derivative. Via equation (20), it turns out that this difference is a preconditioned version of the operator whose roots are sought by our method, namely $\mathcal{L} + \mathcal{N}$, preconditioned by \mathcal{L} . Both methods solve the linear systems that are the core of Newton's method via GMRES, which relies on repeated actions of the Jacobian. For the integration method, the action of the Jacobian consists of integrating the linearized equations via many small timesteps, while for us it consists taking a single large linearized timestep. Here, there is a trade-off. Despite the preconditioning displayed by (20), the Jacobian resulting from Stokes preconditioning is less well conditioned than that which results from taking many small linearized timesteps, and so more Jacobian actions are required for GMRES to converge to a solution of the linear system (21). However, each action costs less, since it consists of only a single timestep. This trade-off – number of actions vs. timesteps per action – can be quantified via the total number of timesteps required to compute a steady state or travelling wave. For wall-bounded shear flows in the transitional regime, we have found that the Stokes preconditioning method is faster than the integration method by a factor of 11 for plane Couette flow and by a factor of 35 for pipe flow; see [25] for details. We have not carried out a timing comparison of the integration and the Stokes preconditioning method for convection in rotating spheres.

Although the Stokes preconditioning method is faster the integration method is more general. The Stokes preconditioning method is only capable of computing steady states or traveling waves, while

the integration method can compute periodic orbits of all kinds, including, for example modulated rotating waves or pulsing states.

A specific difference between our implementation of implicit Coriolis integration and that of [21–24,28,39] is that we solve the linear systems arising from implicit treatment of the diffusive and Coriolis terms directly via block pentadiagonal LU decomposition and backsolving, whereas [21–24,28,39] use Krylov methods to solve these systems. Time-integration using explicit, implicit and semi-implicit treatment of the Coriolis term is compared by [39]. Our strategy is to solve linear systems directly as much as possible and to resort to Krylov only to invert the full Jacobian.

One might wonder about the applicability of this achievement to geophysical and astrophysical research. Small Ekman numbers, like large Reynolds numbers, are associated with chaos and turbulence, not with the regularity and periodicity of traveling waves, which moreover are almost invariably unstable. The Boussinesq and Navier-Stokes equations generally contain a very large number of solution branches, most of them partly or completely unstable [e.g., 33,36–38,40–42]. This profusion of states, sometimes called Exact Coherent Structures, has become the focus of extensive research, motivated by the idea [43,44] that turbulence could be viewed from a dynamical systems perspective as a collection of trajectories ricocheting between periodic orbits along their unstable directions, a kind of deterministic analogue of ergodic theory.

We now turn to the relevance of implicit Coriolis timestepping for geophysics. Implicit Coriolis timestepping not only speeds up calculations of stable and realistic flows, it also allows us to take much larger timesteps that yield inaccurate results. Constraints on timesteps (whether for stability or accuracy) arise from different physical forces. In rapidly rotating fluids, the Coriolis force generates inertial waves, which have a large range of frequencies and are continually generated and damped. Using a timestep that greatly exceeds the constraint for resolving inertial waves can be considered analogous to using the incompressible Navier-Stokes equations, which deliberately neglect the high-frequency sound waves [14,15].

Acknowledgments: We thank Adrian van Kan and Benjamin Favier for helpful discussions.

Author Contributions: The results were obtained by JCGS. FF provided discussions and additional results. The program was written successively by AR, CR and JCGS. The idea was suggested by LST and the paper was written by LST and JCGS.

Funding: The calculations for this work were performed using high performance computing resources provided by the Grand Equipement National de Calcul Intensif (GENCI) at the Institut du Développement et des Ressources en Informatique Scientifique (IDRIS, CNRS) through Grant No. A0162A01119.

References

1. Chandrasekhar, S. *Hydrodynamic and hydromagnetic stability*; Oxford Univ. Press, 1961.
2. Busse, F. Thermal instabilities in rapidly rotating systems. *J. Fluid Mech.* **1970**, *44*, 441–460.
3. Zhang, K.K.; Busse, F.H. On the onset of convection in rotating spherical shells. *Geophys. Astrophys. Fluid Dyn.* **1987**, *39*, 119–147.
4. Dormy, E.; Soward, A.; Jones, C.; Jault, D.; Cardin, P. The onset of thermal convection in rotating spherical shells. *J. Fluid Mech.* **2004**, *501*, 43–70.
5. Geiger, G.; Busse, F. On the onset of thermal convection in slowly rotating fluid shells. *Geophys. Astrophys. Fluid Dyn.* **1981**, *18*, 147–156. <https://doi.org/10.1080/03091928108208777>.
6. Ardes, M.; Busse, F.; Wicht, J. Thermal convection in rotating spherical shells. *Phys. Earth Planet. Inter.* **1997**, *99*, 55–67.
7. Kitauchi, H.; Araki, K.; Kida, S. Flow structure of thermal convection in a rotating spherical shell. *Nonlinearity* **1997**, *10*, 885–904. <https://doi.org/10.1088/0951-7715/10/4/005>.
8. Simitev, R.; Busse, F. Patterns of convection in rotating spherical shells. *New J. Phys.* **2003**, *5*, 97. <https://doi.org/10.1088/1367-2630/5/1/397>.
9. Al-Shamali, F.; Heimpel, M.; Aurnou, J. Varying the spherical shell geometry in rotating thermal convection. *Geophys. Astrophys. Fluid Dyn.* **2004**, *98*, 153–169. <https://doi.org/10.1080/03091920410001659281>.

10. Kimura, K.; Takehiro, S.I.; Yamada, M. Stability and bifurcation diagram of Boussinesq thermal convection in a moderately rotating spherical shell. *Phys. Fluids* **2011**, *23*, 074101. <https://doi.org/10.1063/1.3602917>.
11. Feudel, F.; Bergemann, K.; Tuckerman, L.S.; Egbers, C.; Futterer, B.; Gellert, M.; Hollerbach, R. Convection patterns in a spherical fluid shell. *Phys. Rev. E* **2011**, *83*, 046304.
12. Feudel, F.; Seehafer, N.; Tuckerman, L.S.; Gellert, M. Multistability in rotating spherical shell convection. *Phys. Rev. E* **2013**, *87*, 023021.
13. Feudel, F.; Tuckerman, L.S.; Gellert, M.; Seehafer, N. Bifurcations of rotating waves in rotating spherical shell convection. *Phys. Rev. E* **2015**, *92*, 053015.
14. van Kan, A.; Julien, K.; Miquel, B.; Knobloch, E. Bridging the Rossby number gap in rapidly rotating thermal convection. *arXiv preprint arXiv:2409.08536* **2024**.
15. Julien, K.; van Kan, A.; Miquel, B.; Knobloch, E.; Vasil, G. Rescaled Equations of Fluid Motion for Well-Conditioned Direct Numerical Simulations of Rapidly Rotating Convection. *arXiv preprint arXiv:2410.02702* **2024**.
16. Ruelle, D. Bifurcations in the presence of a symmetry group. *Arch. Rat. Mech. Anal.* **1973**, *51*, 136–152.
17. Ecke, R.; Zhong, F.; Knobloch, E. Hopf bifurcation with broken reflection symmetry in rotating Rayleigh-Bénard convection. *Europhysics Letters* **1992**, *19*, 177.
18. Chossat, P.; Lauterbach, R. *Methods in Equivariant Bifurcations and Dynamical Systems*; World Scientific: Singapore, 2000.
19. Hollerbach, R. A spectral solution of the magneto-convection equations in spherical geometry. *Int. J. Numer. Methods Fluids* **2000**, *32*, 773–797.
20. Schaeffer, N. Efficient spherical harmonic transforms aimed at pseudospectral numerical simulations. *Geochem. Geophys. Geosyst.* **2013**, *14*, 751–758.
21. Sánchez, J.; Net, M.; Garcia-Archilla, B.; Simó, C. Newton–Krylov continuation of periodic orbits for Navier–Stokes flows. *J. Comput. Phys.* **2004**, *201*, 13–33.
22. Garcia, F.; Sánchez, J.; Net, M. Antisymmetric Polar Modes of Thermal Convection in Rotating Spherical Fluid Shells at High Taylor Numbers. *Phys. Rev. Lett.* **2008**, *101*, 194501.
23. Sánchez, J.; Garcia, F.; Net, M. Computation of azimuthal waves and their stability in thermal convection in rotating spherical shells with application to the study of a double-Hopf bifurcation. *Phys. Rev. E* **2013**, *87*, 033014.
24. Garcia, F.; Net, M.; Sánchez, J. Continuation and stability of convective modulated rotating waves in spherical shells. *Phys. Rev. E* **2016**, *93*, 013119.
25. Tuckerman, L.S.; Langham, J.; Willis, A. Order-of-magnitude speedup for steady states and traveling waves via Stokes preconditioning in Channelflow and Openpipeflow. In *Computational Modelling of Bifurcations and Instabilities in Fluid Dynamics*; Springer, 2018; pp. 3–31.
26. Marti, P.; Calkins, M.; Julien, K. A computationally efficient spectral method for modeling core dynamics. *Geochemistry, Geophysics, Geosystems* **2016**, *17*, 3031–3053.
27. Cardin, P.; Dormy, E.; Gibbons, S.; Glatzmaier, G.; Honkura, Y.; Jones, C.; Kono, M.; Matsushima, M. A numerical dynamo benchmark. *Phys. Earth Planet. Inter.* **2001**, *128*, 25–34.
28. Garcia, F.; Sánchez, J.; Dormy, E.; Net, M. Oscillatory Convection in Rotating Spherical Shells: Low Prandtl Number and Non-Slip Boundary Conditions. *SIAM Journal on Applied Dynamical Systems* **2015**, *14*, 1787–1807, [<https://doi.org/10.1137/15M100729X>]. <https://doi.org/10.1137/15M100729X>.
29. Ishioka, K. A New Recurrence Formula for Efficient Computation of Spherical Harmonic Transform. *Journal of the Meteorological Society of Japan* **2018**, *96*, 241–249. <https://doi.org/10.2151/jmsj.2018-019>.
30. Saad, Y.; Schultz, M.H. GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM J. Sci. and Stat. Comput.* **1986**, *7*, 856–869.
31. Tuckerman, L.S. Steady-state solving via Stokes preconditioning; recursion relations for elliptic operators. In Proceedings of the 11th Int. Conf. Num. Meth. Fluid Dyn. Springer, 1989, pp. 573–577.
32. Mercader, I.; Batiste, O.; Alonso, A. Continuation of travelling-wave solutions of the Navier–Stokes equations. *Int. J. Num. Meth. Fluids* **2006**, *52*, 707–721.
33. Borońska, K.; Tuckerman, L.S. Extreme multiplicity in cylindrical Rayleigh-Bénard convection. II. Bifurcation diagram and symmetry classification. *Phys. Rev. E* **2010**, *81*, 036321.
34. Marcus, P.S. Simulation of Taylor-Couette flow. Part 1. Numerical methods and comparison with experiment. *Journal of Fluid Mechanics* **1984**, *146*, 45–64.
35. Duguet, Y.; Pringle, C.C.; Kerswell, R.R. Relative periodic orbits in transitional pipe flow. *Phys. Fluids* **2008**, *20*.

36. Gibson, J.F.; Halcrow, J.; Cvitanović, P. Equilibrium and travelling-wave solutions of plane Couette flow. *J. Fluid Mech.* **2009**, *638*, 243–266.
37. Zheng, Z.; Tuckerman, L.S.; Schneider, T.M. Natural convection in a vertical channel. Part 1. Wavenumber interaction and Eckhaus instability in a narrow domain. *J. Fluid Mech.* **2024**, *1000*, A28.
38. Zheng, Z.; Tuckerman, L.S.; Schneider, T.M. Natural convection in a vertical channel. Part 2. Oblique solutions and global bifurcations in a spanwise-extended domain. *J. Fluid Mech.* **2024**, *1000*, A29.
39. García González, F.; Net Marcé, M.; Garcia-Archilla, B.; Sánchez Umbría, J. A comparison of high-order time integrators for thermal convection in rotating spherical shells. *J. Comput. Phys.* **2010**, *229*, 7997–8010.
40. Schmiegel, A. Transition to turbulence in linearly stable shear flows. PhD thesis, Philipps-Universität Marburg, 1999. <http://archiv.ub.uni-marburg.de/diss/z2000/0062/>.
41. Pringle, C.C.; Duguet, Y.; Kerswell, R.R. Highly symmetric travelling waves in pipe flow. *Phil. Trans. R. Soc. Lond. A* **2009**, *367*, 457–472.
42. Borońska, K.; Tuckerman, L.S. Extreme multiplicity in cylindrical Rayleigh-Bénard convection. I. Time dependence and oscillations. *Phys. Rev. E* **2010**, *81*, 036320.
43. Cvitanovic, P.; Eckhardt, B. Periodic orbit expansions for classical smooth flows. *J. Phys. A-Math.* **1991**, *24*, L237.
44. Kawahara, G.; Uhlmann, M.; Van Veen, L. The significance of simple invariant solutions in turbulent flows. *Annu. Rev. Fluid Mech.* **2012**, *44*, 203–225.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.