JOURNAL OF COMPUTATIONAL PHYSICS 115, 302–318 (1994)

Spectrally Accurate Contour Dynamics

R. D. VAN BUSKIRK AND P. S. MARCUS

Department of Mechanical Engineering, University of California, Berkeley, California 94720

Received July 15, 1994; revised April 7, 1994

We present an exponentially accurate boundary integral method for calculating the equilibria and dynamics of piece-wise constant distributions of potential vorticity. The method represents contours of potential vorticity as a spectral sum and solves the Biot–Savart equation for the velocity by spectrally evaluating a desingularized contour integral. We use the technique in both an initial-value code and a Newton continuation method. Our methods are tested by comparing the numerical solutions with known analytic results, and it is shown that for the same amount of computational work our spectral methods are more accurate than other contour dynamics methods currently in use.

1. INTRODUCTION

Boundary integral methods are often used to solve for the motion of two-dimensional, nondissipative flows with interfaces [1–6]. When the interface represents a jump in potential vorticity, the magnitude of the discontinuity at the interface is constant in time, and the velocity of the fluid is completely specified by the location of the interface and by the boundary conditions. The motions of the vorticity interface and the method of calculation are collectively known as "contour dynamics" [7] (henceforth referred to as CD). The earliest implementations of CD approximated the locations of the interfaces with Lagrangian marker particles connected by low-order polynomials and used a variety of methods for computing the contour integral along the interface to obtain the velocity. Second- and third-order accurate methods have been reviewed by Zou *et al.* [8] and Dritschel [9]. Baker and Shelly [10, 11] developed a sixth-order method that uses Hermite polynomials for interpolation and quintic splines to calculate derivatives. Relaxation methods that converge to steady equilibria, rather than solve the initial-value problem, use modifications of these CD Methods [1, 12, 13], with an exception being the work of Pullin and Jacobs [5] and Pullin *et al.* [14], where spectral methods are used to represent a steady interface. Pullin *et al.* used their CD method to examine the stability of vortex layers, but they were only able to study perturbations of moderate amplitude due to difficulties with convergence.

In this paper we show that for the same amount of computational work needed by the second- and third-order accurate CD methods, there are spectral CD algorithms that are exponentially accurate and robust for both initial-value problems and steady-state equilibrium finders. In Section 2 we present the equations for potential vorticity dynamics and formulate the CD of piecewise uniform distributions of potential vorticity. In Section 3 a spectral representation for the locations of contours is described, and its errors are analyzed. Section 4 describes our quadrature method for the contour and discusses the accuracy of our velocity calculation. In Section 5, we test the performance of our CD initial-value code. A Newton pseudo-arclength continuation method for finding steady solutions is described in Section 6, and its rate of convergence and accuracy are analyzed. In Section 7 we discuss an inverse problem: given the boundary of a patch of uniform potential vorticity find a zonal velocity and/or bottom topography so that the flow is in equilibrium. Our conclusions and a discussion of implications for future work are presented in Section 8.

2. FORMULATION OF THE PROBLEM

In the quasi-geostrophic limit, the equations that govern a one-layer, constant density, rapidly rotating flow with a free surface can be written in terms of the potential vorticity $q(x, y, t)$:

$$\left[\frac{\partial}{\partial t} + (\mathbf{v} \cdot \nabla)\right] q \equiv \frac{Dq}{Dt} = 0. \tag{1}$$

In this limit the velocity $\mathbf{v}(x, y, t)$ is two dimensional and divergence-free,

$$v_z \equiv 0 \tag{2}$$

$$\nabla \cdot \mathbf{v} = 0. \tag{3}$$

The potential vorticity [15] in terms of the stream function ψ is written

$$q(x, y, t) \equiv \nabla^2\psi(x, y, t) - \frac{\psi(x, y, t)}{L_R^2} + \beta(x, y), \tag{4}$$

where L_R is the Rossby deformation radius (a measure of the depth of the layer) and $\beta(x, y)$ is the generalized β-plane term

0021-9991/94 $6.00

which has contributions due to both the variations in the Coriolis force and the bottom topography of the layer.[1] The stream function ψ is related to the velocity and the vorticity ω in the usual way,

$$\mathbf{v} \equiv \hat{\mathbf{z}} \times \nabla\psi \tag{5}$$

and

$$\omega \equiv \nabla^2\psi, \tag{6}$$

where $\hat{\mathbf{z}} \equiv \hat{\mathbf{x}} \times \hat{\mathbf{y}}$ is the unit vector perpendicular to the two-dimensional plane. We arbitrarily decompose the potential vorticity into two components defining $q(x, y, t) \equiv \bar{q} + q'(x, y, t)$, where we require $\bar{q}$ to be constant in space and time. We also decompose $\psi(x, y, t)$ such that $\psi(x, y, t) \equiv \bar{\psi}(x, y) + \psi'(x, y, t)$, where $\bar{\psi}$ is steady and solves the linear, inhomogeneous Helmholtz equation:

$$\bar{q} \equiv \nabla^2\bar{\psi}(x, y) - \frac{\bar{\psi}(x, y)}{L_R^2} + \beta(x, y). \tag{7}$$

Then $\psi'(x, y, t)$ is the solution to the homogeneous Helmholtz equation

$$q'(x, y, t) = \nabla^2\psi'(x, y, t) - \frac{\psi'(x, y, t)}{L_R^2} \tag{8}$$

and q' is governed by

$$\left[\frac{\partial}{\partial t} + (\mathbf{v} \cdot \nabla)\right] q' = 0. \tag{9}$$

We also define $\mathbf{v} \equiv \bar{\mathbf{v}} + \mathbf{v}'$, where $\bar{\mathbf{v}}$ and $\mathbf{v}'$ are related to their respective streamfunctions, $\bar{\psi}$ and ψ', as in (5).

Our motivation for decomposing $\mathbf{v}$, ψ, and q into steady and time-dependent components is that it allows us to use Green's functions. Because there is a linear *homogeneous* differential operator that relates ψ' to q', we can use a Green's function to calculate ψ' from q' in the standard way [16–18] and thereby use the generalized Biot–Savart law for finite L_R,

$$\psi'(x, y, t) = \iint G(|\mathbf{x} - \mathbf{x}'|)q'(x', y', t)dx'dy', \tag{10}$$

where the Green's function G is defined to be the solution to

$$\delta(\mathbf{x} - \mathbf{x}') = \left(\nabla^2 - \frac{1}{L_R^2}\right) G(|\mathbf{x} - \mathbf{x}'|) \tag{11}$$

and where G satisfies the desired boundary conditions. In this paper we shall impose the boundary condition that the gradient of G be zero at infinity so that the velocity goes to zero at infinity. By taking the gradient of (8), we obtain similar equations for the velocity:

$$\hat{\mathbf{z}} \times \nabla q = \left[\nabla^2 - \frac{1}{L_R^2}\right] \mathbf{v}'. \tag{12}$$

Note that because $\bar{q}$ is uniform, $\nabla q = \nabla q'$. The Green's function that relates q to ψ' is the same as the one that relates ∇q to the velocity:

$$\mathbf{v}'(\mathbf{x}) = \iint G(|\mathbf{x} - \mathbf{x}'|)(\hat{\mathbf{z}} \times \nabla q)dx'dy'. \tag{13}$$

The above area integral reduces to a contour integral when q' is piece-wise constant. If q' consists of n patches of piece-wise constant potential vorticity, (13) becomes

$$\mathbf{v}'(\mathbf{x}) = -\sum_{i=1}^{n} \Delta q_i \oint G(|\mathbf{x} - \mathbf{x}'(s)|)\mathbf{ds}, \tag{14}$$

where Δq_i is the jump in potential vorticity across the i^{th} interface (by definition of $\bar{q}$, $q' \equiv 0$ outside the patches), s is the arc length along the contour, and $\mathbf{ds} \equiv (d\mathbf{x}'/ds)ds$. The sign of this expression is chosen as follows: The parameterization is assumed to be counterclockwise for a closed contour, the unit normal is outward, and Δq is defined to be positive if q is higher inside a contour than it is outside.

For simplicity in this paper we shall examine one or two finite area patches of $\Delta q'$. For finite L_R, the zeroth-order modified Bessel function $-K_0(|\mathbf{x} - \mathbf{x}'|/L_R)/2\pi$ is the Green's function [16–18], while for infinite L_R, $G = \log(|\mathbf{x} - \mathbf{x}'|)/2\pi$. Note that for infinite L_R and $\nabla \times \bar{\mathbf{v}}$ equal to a constant, the equation of motion (1) reduces to Euler's equation. The application of the method to finite domains or periodic boundary conditions can be accomplished by using the appropriate Green's function. For the remainder of the paper we shall describe our methods in terms of Green's functions so that applications to other boundary conditions are clear.

3. REPRESENTATION OF THE BOUNDARY

CD methods consist of three distinct parts: (1) a method of approximating the location of the contours, (2) a quadrature for the contour integral, and (3) an algorithm for advecting or updating the contours. In this section we describe the first part: how we spectrally represent the boundary. Generally, a closed contour in two dimensions can be represented as two periodic

[1] We define g as the local acceleration of gravity, $f(x, y)$ as the local Coriolis force with f_0 as its average value, $[H_0 + h(x, y, t) - h_B(x, y)]$ as the depth of the layer with H_0 as its average value, h as the height of the free surface, and h_B as the depth of the bottom topography. Then, $\beta \equiv f(x, y) - f_0 + h_B f_0/H_0$, $L_R \equiv \sqrt{gH_0}/f_0$, and $\psi \equiv gh/f_0$.

functions of a parameterization τ that give the coordinates of the points on the contour: $[x(\tau), y(\tau)]$. Since these two functions are periodic (and often analytic) our approach is to represent them spectrally:

$$x(\tau) = \mathrm{Re}\left[\sum_{n=0}^{N/2-1} X_n e^{in\tau}\right] \tag{15}$$

and

$$y(\tau) = \mathrm{Re}\left[\sum_{n=0}^{N/2-1} Y_n e^{in\tau}\right]. \tag{16}$$

The $N/2$ complex Fourier coefficients, X_n and Y_n, are related *via* discrete Fourier transform to the N points $\mathbf{x}_i \equiv [x(\tau_i), y(\tau_i)]$, which are equally spaced in τ. We shall pick our parameterization τ to have period 2π, and we shall pick the N discrete points as

$$\tau_i \equiv 2\pi(i - 1/2)/N, \quad i = 1, 2, \ldots, N. \tag{17}$$

Note that the above definition of the boundary not only defines a curve that interpolates between the discrete points, but also is an algorithm for taking derivatives. In contrast, the second-order methods described by Zou *et al.* use piece-wise linear approximations for $x(\tau)$ and $y(\tau)$, while Dritschel uses cubic splines to approximate the boundary.

The accuracy of any representation of a boundary is dependent on how the $\{\tau_i\}$ are physically distributed on the boundary. Consider a given contour with N discrete points $\mathbf{x}_i$ placed along it. Their distribution along the boundary depends on the definition of τ. For example, for a simple contour about the origin, one might choose $\tau = \theta$, where θ is the azimuthal polar coordinate. Or, τ could be chosen to be proportional to the arc-length s, or to s modified by a weighting function of the local curvature. In any case, once τ and N are chosen, the discrete points $\mathbf{x}_i \equiv \mathbf{x}(\tau_i)$ are determined uniquely and by discrete Fourier transform so are the $N/2$ complex Fourier coefficients (X_n, Y_n) used in (15) and (16). With (15) and (16) our approximation for the location of the boundary for any continuous value of τ is uniquely determined. To quantify the error and to test for the rate of convergence of our approximation as a function of N, we define the error $\varepsilon_b(N)$ to be a measure of the difference between the approximation and exact locations of the boundary for K discrete values of τ,

$$\varepsilon_b(N) \equiv \left[\frac{\sum_{j=1}^{K} |\mathbf{x}_N(\tau_j') - \mathbf{x}(\tau_j')|^2}{\sum_{j=1}^{K} |\mathbf{x}(\tau_j')|^2}\right]^{1/2}, \tag{18}$$

where $\tau_j' \equiv 2\pi(j - 1/2)/K$, $N/2$ is the number of complex Fourier modes in (15) and (16), $\mathbf{x}_N(\tau_j')$ is the numerically interpolated boundary point at τ_j' using (15) and (16), and $\mathbf{x}(\tau_j')$ is the exact boundary position at τ_j'. Note that $K \gg N$.

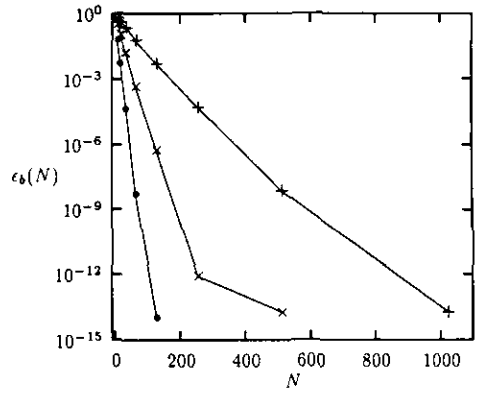

FIG. 1. The error $\varepsilon_b(N)$ as a function of the number of discrete points N on the boundary for an elliptical boundary with the boundary points equally spaced in τ, where τ is the azimuthal polar coordinate θ. The errors are plotted for (●) 2 : 1, (×) 5 : 1, and (+) 15 : 1 ellipses. The value of K used in Eq. (18) is 2048. Note that the resolution is always limited by roundoff errors of order 10^{-15}.

We illustrate our approximation and its error for an elliptical boundary and for a "cat's-eye" shaped boundary for various choices of τ. We choose an ellipse as an example because the boundary is analytic and because ellipses are exact solutions (both steady and time-periodic) to Euler's equation with appropriate $\bar{\mathbf{v}}$. The "cat's-eye" shaped boundary is of interest because it approximates the shape of steady vortices with discontinuous tangent vectors (see, for example, Fig. 14).

Note that for an ellipse, a truncated spectral representation with $N = 2$ could be exact; if we define τ such that

$$(x, y) = [a \cos(\tau), b \sin(\tau)], \tag{19}$$

then only the first two Fourier coefficients in (15) and (16) are needed, so with proper placement only four points would be needed to represent the boundary *exactly*. This is a rather special circumstance, and in general we will not be able to know the ideal point distribution a priori.

Figure 1 is a plot of $\varepsilon_b(N)$ as a function of N for elliptical boundaries centered at the origin with three different values of the ellipticity and for the parameterization $\tau \equiv \theta$, where θ is the azimuthal polar coordinate. The figure shows the exponential convergence of the boundary representation as a function of N. It also shows that the representation is less accurate for more elongated ellipses. Figure 2 shows $\varepsilon_b(N)$ for a distribution of $\mathbf{x}_j$ that is concentrated at the two points of maximum curvature. Specifically, the points are equally spaced in τ, where τ is defined $\theta \equiv \tau - \sin(2\tau)/2$ and where the major axes of the ellipses are aligned along the x-axis. Figure 2 shows the obvious advantage of having the points $\mathbf{x}_i$ clustered in regions of high curvature. Note that a 50:1 ellipse with only 128 points has an ε_b less than 10^{-5} for this parameterization. By increasing N to

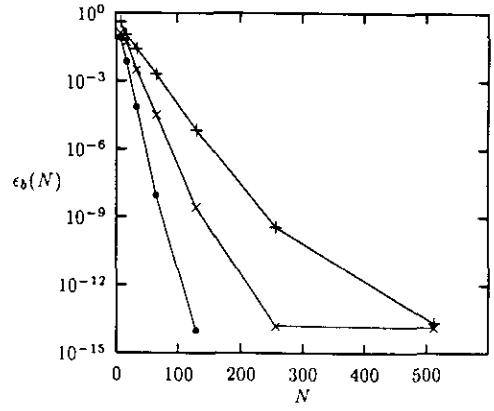

FIG. 2. Same as Fig. 1 but with τ chosen so that $\theta(\tau) \equiv \tau - \sin(2\tau)/2$. We plot errors for (●) 2:1, (×) 10:1, and (+) 50:1 ellipses.

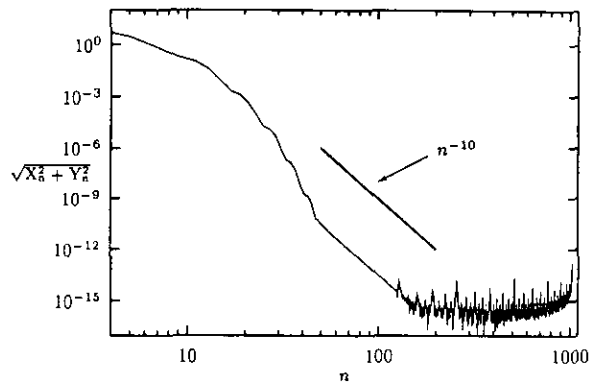

FIG. 3. The magnitude of the Fourier coefficients $(X_n^2 + Y_n^2)^{1/2}$ as a function of n for the $\tau^{(3)}$ parameterization with $N = 1024$ for the "cat's-eye" shaped boundary. A line with a slope of n^{-10} is also plotted to compare with the asymptotic behavior at $50 \leq n \leq 100$. For $n > 100$, roundoff errors of 10^{-15} dominate.

512 even this highly elongated an ellipse can be represented to within the accuracy of our 10^{-15} roundoff error.

We have also examined the errors in representing a boundary that has a discontinuous tangent vector (i.e., the local radius of curvature of the boundary goes to zero). This is of interest because many of our families of calculated steady solutions [18, 19, 20] have limits in which the tangent vectors are discontinuous, (cf. Fig. 14). The specific boundary that we examine here is given in polar coordinates by: $r \equiv 1/[1 + \sin(\theta)]$, for $0 < \theta < \pi$ and $r \equiv 1/[1 - \sin(\theta)]$, for $\pi < \theta < 2\pi$. This boundary is simply two sections of a parabola connected together. We use three distributions of points $\mathbf{x}_i$ which are defined by their respective parameterizations $\tau^{(l)}$, $l = 1, 2, 3$, where $\theta \equiv \tau^{(1)}$, $\theta \equiv \tau^{(2)} - \sin(2\tau^{(2)})/2$, and $\theta \equiv \tau^{(3)} - \sin(2\tau^{(3)})/2 - \sin[2\tau^{(3)} - \sin(2\tau^{(3)})]/2$. All three distributions have equally spaced points in τ, but they are concentrated at $\tau_0^{(l)}$, the location of the discontinuous tangent vector (i.e., at $\theta = 0$ or $\theta = \pi$), to differing degrees. Near the discontinuities, $\theta = O(\tau^{(1)} - \tau_0^{(1)})$, $\theta = O(\tau^{(2)} - \tau_0^{(2)})^3$, and $\theta = O(\tau^{(3)} - \tau_0^{(3)})^9$, respectively. Writing $d\mathbf{x}/d\tau \equiv (d\mathbf{x}/d\theta)\,(d\theta/d\tau)$, we see that $\mathbf{x}(\tau^{(1)})$ is C^0, $\mathbf{x}(\tau^{(2)})$ is C^2, and $\mathbf{x}(\tau^{(3)})$ is C^8. If a periodic function is C^k and if its $(k+1)^{\text{th}}$ derivative is piece-wise continuous and differentiable, then the Fourier coefficients asymptotically decrease as $n^{-(k+2)}$. Therefore, the Fourier coefficients X_n and Y_n in (15)–(16) should asymptotically decrease differently for each of the three parameterizations. Specifically, the highest Fourier coefficients should be proportional to n^{-2} for $\tau^{(1)}$, n^{-4} for $\tau^{(2)}$, and n^{-10} for $\tau^{(3)}$. For $\tau^{(1)}$ and $\tau^{(2)}$ we have numerically found that this asymptotic behavior holds, but for $\tau^{(3)}$ the numerical results are somewhat surprising. A plot of the Fourier coefficients for $\tau^{(3)}$ with $N = 1024$ is shown in Fig. 3 (only the even coefficients are plotted because the symmetry makes the odd ones zero.) The Fourier coefficients fall off *faster* than n^{-10} (almost exponentially) for $n < 50$. Only for $n > 50$ is the asymptotic behavior present. For $n > 100$ the asymptotic behavior is overwhelmed by roundoff error. What is surprising (based on previous experience with spectral methods) is that the Fourier coefficients with low n usually decrease *slower* than the asymptotic limit. We have no explanation for (but are certainly pleased with) the initial rapid (nearly exponential) decrease of the coefficients.

Figure 4 illustrates the error $\varepsilon_b(N)$ of these three representations as a function of N. In this case and in general, we find that $\varepsilon_b(N)$ correlates well with $\sqrt{X_N^2 + Y_N^2}$. In particular, for the parameterization $\tau^{(1)}$, ε_b decreases as N^{-2}; for $\tau^{(2)}$ as N^{-4}; while for $\tau^{(3)}$ the error decreases exponentially with N until roundoff error dominates at 10^{-15}.

4. VELOCITY CALCULATION

Our quadrature method for calculating the contour integral in (14) to obtain the velocity (and all contour integrals presented in this paper) is based on the fact that the integral of a periodic function is proportional to the zeroth Fourier mode of the integrand. Because the Green's function in the integrand has a loga-

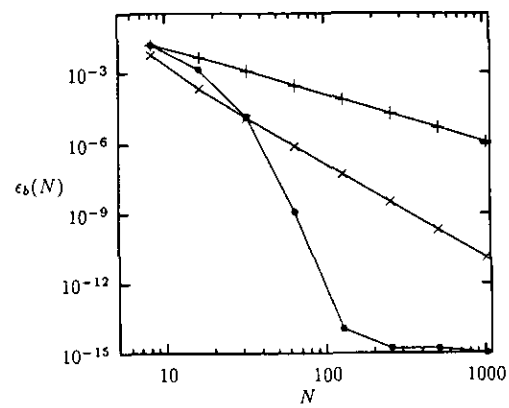

FIG. 4. The error $\varepsilon_b(N)$ of the three approximations of the "cat's-eye" shaped boundary as a function of the number of points on the boundary N. We plot results for (+) $\tau^{(1)}$, (×) $\tau^{(2)}$, and (●) $\tau^{(3)}$, respectively.

rithmic singularity, its Fourier expansion has slow convergence, and approximating its integral with a sum over collocation points results in large aliasing errors of the zeroth Fourier mode. Following Baker *et al.* [21] we desingularize the integral by using an integration by parts,

$$\begin{aligned}\mathbf{v}'(\mathbf{x}) &= -\Delta q \oint G[|\mathbf{x}-\mathbf{x}'(\tau)|]\frac{d\mathbf{x}'}{d\tau}\,d\tau \\ &= \Delta q \oint [\mathbf{x}-\mathbf{x}'(\tau)]\dot{G}[|\mathbf{x}-\mathbf{x}'(\tau)|]\frac{d[|\mathbf{x}-\mathbf{x}'(\tau)|]}{d\tau}\,d\tau,\end{aligned} \tag{20}$$

where $\dot{G}(r) \equiv dG(r)/dr$. The surface term vanishes because the contour is closed. If the singularity of G is logarithmic, which is the case for two-dimensional potential vorticity CD formulations, then the new form of the integral is nonsingular. Specifically for $L_R \to \infty$, we have $G(r) = \log(r)/2\pi$ and

$$\mathbf{v}'(\mathbf{x}) = \frac{\Delta q}{2\pi}\oint \frac{[\mathbf{x}-\mathbf{x}'(\tau)]}{r^2}\frac{d(r^2)}{d\tau}\,d\tau, \tag{21}$$

where $r^2 = |\mathbf{x}-\mathbf{x}'(\tau)|^2$. We compute the integrals in (20) and (21) numerically by evaluating the integrand at M equally spaced points in τ, summing the evaluation, and multiplying by the τ-spacing. This is a spectral calculation of the contour integral.

Note that the parameterization of τ and value of M (the number of collocation points), used in computing the contour integral in (21), need not be the same as the τ and N used in representing the location of the boundary in (15) and (16). However, for simplicity throughout the remainder of this paper we shall choose the parameterization of the two τ's to be the same, although in general $M \neq N$.

From (20) our approximation of the velocity at $\mathbf{x}$ with M collocation points is

$$\begin{aligned}\mathbf{v}'_M(\mathbf{x}) = \frac{2\pi\Delta q}{M}\sum_{m=1}^{M}\frac{\Delta x_m}{|\Delta\mathbf{x}_m|}\dot{G}(|\Delta\mathbf{x}_m|)\Bigg\{&[x'(\hat{\tau}_m)-x]\frac{dx'(\hat{\tau}_m)}{d\tau} \\ &+[y'(\hat{\tau}_m)-y]\frac{dy'(\hat{\tau}_m)}{d\tau}\Bigg\},\end{aligned} \tag{22}$$

where the M collocation points $\mathbf{x}'(\hat{\tau}_m)$ on the boundary are defined as

$$\hat{\tau}_m \equiv 2\pi m/M, \quad m = 0, 1, \cdots, M-1, \tag{23}$$

where derivatives in (22) are evaluated spectrally and where $\Delta\mathbf{x}_m \equiv \mathbf{x} - \mathbf{x}'(\hat{\tau}_m)$.

In solving the steady-state and initial-value problems, we shall need to compute the velocity of marker particles on the boundary and therefore we need to compute $\mathbf{v}'(\mathbf{x})$ on the boundary. Thus, a fair test of our velocity-finding algorithm requires

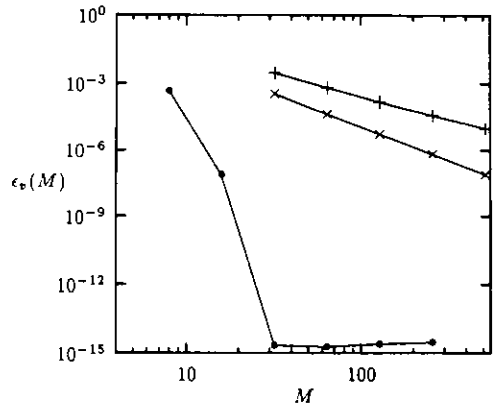

FIG. 5. The velocity error $\varepsilon_v(M)$ of our spectral method (●) as a function of the number of collocation points M. For this test we have set P in (24) and (25) equal to M. We also plot the ε_v errors of the second-order R2 method (+) and Dritschel's third-order method (×) as described by Zou *et al.* [8].

examining the error when $\mathbf{v}'(\mathbf{x})$ is evaluated on a set of P test points $\mathbf{x}_p$ on the boundary defined by

$$\mathbf{x}_p \equiv \mathbf{x}(\tau_p): \tau_p = 2\pi(p-\tfrac{1}{2})/P, \quad p = 1, 2, \cdots, P. \tag{24}$$

Figure 5 shows the results of a test where the boundary is a 2:1 ellipse, so $\mathbf{v}'(\mathbf{x})$ is known analytically. We define an L_2 error $\varepsilon_v(M)$,

$$\varepsilon_v(M) \equiv \left[\frac{\sum_{p=1}^{P}|\mathbf{v}'_M(\mathbf{x}_p)-\mathbf{v}'(\mathbf{x}_p)|^2}{\sum_{p=1}^{P}|\mathbf{v}'(\mathbf{x}_p)|^2}\right]^{1/2}, \tag{25}$$

where $\mathbf{v}'_M$ is our approximate velocity given by (22) and where $\mathbf{v}'$ is the exact analytically known velocity. As noted earlier, the boundary of an ellipse with aspect ratio λ can be represented exactly by the parameterization in (19). We have used this choice of parameterization in evaluating (25), so that all of the errors in $\varepsilon_v(M)$ are due to quadrature and not to the boundary representation. Figure 5 shows that the error decreases approximately exponentially with M until the roundoff error is reached at $M = 32$.

We now compare our velocity error with the algorithms discussed by Zou *et al.* and Dritschel [9]. Zou *et al.* describe three second-order methods (all of which approximate the vortex boundary by an N-polygon); one method calculates the contour integral by using the exact formula for integration along line segments, and the other two use a midpoint integration for two different desingularizations of the integrand. One of the two latter methods (which they referred to as R2) uses the desingularized contour integral in (21) and is the fastest of the three methods for a given M. We can compare our spectral methods with these methods because Zou *et al.* use the same distribution of test points $\mathbf{x}_p$ as we do, the same (2:1) elliptical boundary, and the same definition of $\varepsilon_v(M)$ as in (25). Figure

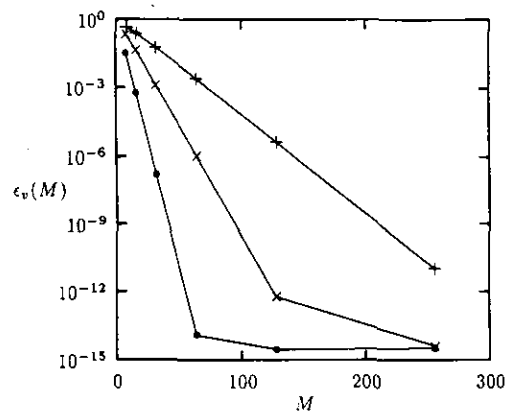

FIG. 6. The velocity error $\varepsilon_v(M)$ of our spectral method for elongated ellipses with three aspect ratios: (●) 4 : 1, (×) 9 : 1, and (+) 20 : 1.

5 clearly illustrates the differences between the second- and third-order methods and our spectral method. Even for the lowest resolution M presented in Zou *et al.*, our method calculates the velocity to an accuracy limited only by roundoff error. In our method only 16 points are needed to calculate the velocity to an accuracy of 10^{-7}. Our spectral method has precisely the same number of computer operations for the quadrature as the R2 method which, according to Zou *et al.*, is the fastest of the second-order methods and is six times faster than the third-order method of Dritschel. Hence our method, though spectrally accurate, should be as fast as the fastest of these other CD algorithms.

In Fig. 6 we show the velocity error $\varepsilon_v(M)$ of our spectral method for three different aspect ratios of highly elongated ellipses. This is a more severe test of the velocity algorithm. For the largest aspect ratios our method loses some accuracy, but still only 128 points are needed to calculate the boundary velocity of a 20 : 1 ellipse to an accuracy of 10^{-5}.

5. APPLICATION TO INITIAL-VALUE CODES

Our spectrally accurate algorithm for computing the velocity can be used in an initial-value solver by using the velocity to advect the boundary points $[x(\tau_i), y(\tau_i)]$ or, equivalently, to evolve the Fourier coefficients X_n and Y_n of the boundary representation. In the previous section we described how to calculate the velocity at P boundary points of a potential vortex. A simple method for solving the initial-value problem is to advect these same boundary points according to $d\mathbf{x}_i/dt = \mathbf{v}(\mathbf{x}_i)$, using any explicit time integration algorithm.

In general three classes of initial-value problems can be solved with CD. In one class vortices never shed filaments and their boundaries remain regular for all time. The lengths of the contours and the radii of curvature of the vortices remain finite, and CD algorithms never break down. Geophysical potential vortices with finite Rossby deformation radius L_r and vortices embedded in shearing zonal flows in which the shear and the potential vorticity have the same sign are often in this class. The algorithm for the temporal evolution of the boundary points can be decoupled from the algorithm that computes the fluid velocity $\mathbf{v}$ as a function of the locations of the points. Thus the exponential convergence of our spectral method for computing $\mathbf{v}$ carries over directly to this class. A second class of flows consists of potential vortices that form one or two large-scale tails that are advected to infinity by $\bar{\mathbf{v}}$ but form no small-scale filaments, cf. Fig. 12. A CD method will eventually break down at (and only at) the tails when they become thinner than the numerical resolution. However, the break down is benign in the sense that it can be circumvented by removing sections of the tail that are thin, far from the main bodies of the vortices, and that would be swept rapidly to infinity by $\bar{\mathbf{v}}$ had they been kept. Their removal can be carried out in a physically motivated way, and the late-time flows have been shown to be insensitive to the details of how the removal is carried out [19, 20]. This class of flows is relevant to geophysical flows such as Jupiter's Great Red Spot [18]. In this section we provide examples of the exponential convergence of our spectral method for these flows.

There is a third class of flows (and the one most frequently discussed in the literature) in which the vortices continuously shed small-scale filaments all along their boundaries. Due to the rapid stretching of the contours and exponentially fast decrease of the radii of curvature, CD methods break down quickly and at many locations. Thus, a spectral method's factor of 10 or 100 in increased spatial resolution over a finite-difference method is not that great an advantage. Contour surgery can be used as a fix-up, but in general it does not cure the break down, it only postpones it [9]. Although we have developed a contour surgery for spectral CD [19], we do not use it in any of the calculations reported in this paper. (Another, perhaps more useful, form of smoothing that keeps a spectral CD from breaking down is the damping of the highest Fourier coefficients in (15) and (16) [19].) We shall not make a direct comparison here between rates of convergence for finite-difference and spectral CD methods with contour surgery or contour smoothing because there are no analytic test cases. Moreover, because there is no proof that any of the dissipative CD algorithms converge to solutions of the nearly inviscid limit of the Navier–Stokes equation, comparisons among the different methods might not be meaningful. Thus we limit the discussion here to the two classes of geophysically interesting flows in which the CD method never breaks down or breaks down benignly.

Our specific procedure for solving the initial-value problem is to first define a τ and the functions of $x(\tau)$ and $y(\tau)$ for the initial boundary shape. For example, in this section many of the initial conditions are ellipses, so we choose (19) for the *initial* relation between τ and (x, y). Once the initial τ is chosen, the initial N locations of the Lagrangian boundary points are defined $\mathbf{x}_i \equiv \mathbf{x}(\tau_i)$ with τ_i given by (17), so the initial $N/2$ Fourier coefficients in (15) and (16) are determined. The M collocation points $\mathbf{x}_m \equiv \mathbf{x}(\hat{\tau}_m)$ with $\hat{\tau}_m$ given by (23) are then determined and used to compute the velocity $\mathbf{v}'_M$, using (22) at

each of the N Lagrangian boundary points $\mathbf{x}_i$. Each Lagrangian boundary point is then evolved forward in time Δt to its new position by solving $d\mathbf{x}_i/dt = \mathbf{v}'(\mathbf{x}_i) + \bar{\mathbf{v}}(\mathbf{x}_i)$. The procedure then repeats. Note that the relation $\mathbf{x}(\tau)$ is a function of time, but in this paper it is for all time $\mathbf{x}_i \equiv \mathbf{x}(\tau_i)$ with τ_i given by (17). In all but the last example of this section N is held fixed. Changing its value to N' is equivalent to adding or subtracting Lagrangian points on the boundary with a Fourier interpolation: the values of the N Fourier coefficients in Eqs. (15) and (16) are fixed while N is replaced by N' in Eq. (17). Changing the definition of τ_i in time while keeping N fixed is equivalent to rezoning the boundary points along the boundary. Only the last example of this section uses rezoning. Van Buskirk [19] has a more general discussion of rezoning algorithms with spectral methods. He found that rezoning with equal arc-lengths between boundary markers worked very well and was robust. Refinements, such as weighting the spacings by the local radii of curvature, did not significantly increase the accuracy of the calculations.

The time-stepping in solving $d\mathbf{x}_i/dt = \mathbf{v}(\mathbf{x}_i, t)$ is carried out either with fourth-order Runge–Kutta or Bulirsch–Stoer. In the numerical solution of the initial-value problem there are three distinct sources of error: errors in boundary representation, errors in calculating the velocity contour integral, and errors in the time integration. In this paper we are not interested in the time integration errors. Thus we will pick a small time step in Runge–Kutta or error tolerance in Bulirsch–Stoer so that the other two sources of error dominate. We do not explicitly enforce the conservation of any quantities (e.g., circulation, energy, and momentum) but use the changes in these conserved quantities as tests of an algorithm's accuracy.

5.1. *Test of Rotating Kirchoff Ellipse*

Analytical examples of the first class of flows are the rotating Kirchoff and oscillating Kida ellipses. Kida [22] first analyzed the evolution of an elliptical patch of potential vorticity q' with a $\bar{\mathbf{v}}$ that is linear in x and y. For a $\bar{\mathbf{v}}$ of the form

$$\bar{\mathbf{v}}(\mathbf{x}) = -(\Omega + \gamma)y\hat{\mathbf{x}} + (\Omega - \gamma)x\hat{\mathbf{y}}, \tag{26}$$

an initially elliptical patch of potential vorticity will remain elliptical for all time. The angle $\phi(t)$ that the major axis of the ellipse makes with the x-axis and the ellipse's aspect ratio $\lambda(t)$, obey

$$\frac{d\lambda}{dt} = 2\lambda\gamma \sin(2\phi) \tag{27}$$

$$\frac{d\phi}{dt} = \Omega + \frac{\Delta q \lambda}{(\lambda + 1)^2} + \lambda \cos(2\phi)\frac{(\lambda^2 + 1)}{(\lambda^2 - 1)}. \tag{28}$$

There is a scale invariance in the problem, and the values of $\lambda(t)$ and $\phi(t)$ do not depend on the area of the ellipse. Kirchhoff's solution is a special case of (27) and (28) with $\bar{\mathbf{v}} = 0$.

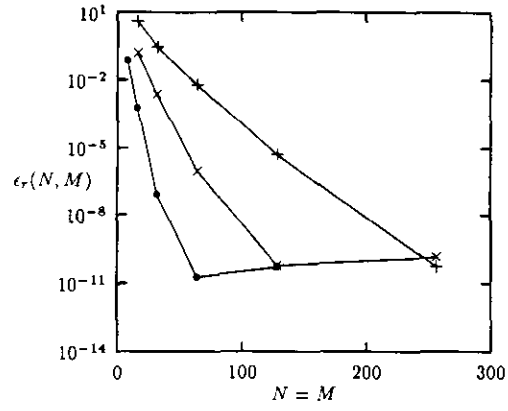

FIG. 7. The error $\varepsilon_r(N, M)$ in the time evolution of a Kirchhoff ellipse as a function of $N = M$ for three aspect ratios: (●) 4 : 1, (×) 9 : 1, and (+) 20 : 1.

A Kirchoff ellipse rotates with the constant angular velocity $\Omega_0 = \Delta q\lambda/(\lambda + 1)^2$ and has a turnaround time or rotational period $T = 2\pi/\Omega_0$. For $\lambda(t) > 1$ the semi-major and semi-minor axes are $a(t) = (A\lambda(t)/\pi)^{1/2}$ and $b(t) = (A/\lambda(t)\pi)^{1/2}$, respectively, where A is the area of the ellipse.

When we compare numerical results to the analytic elliptical solution, we are not interested in the precise distribution of Lagrangian boundary points $\mathbf{x}_i$ along the boundary, but in whether or not they lie on the ellipse given by the analytical solution. Hence we define an error that is independent of the distribution. When the exact analytically known solution is an ellipse with semi-major axis $a(t)$, semi-minor axis $b(t)$, and angle $\phi(t)$, we define the boundary error ε_r,

$$\varepsilon_r(N, M) \equiv \left[\frac{1}{N}\sum_{i=1}^{N}\left(\frac{\bar{x}_i^2}{a(t)^2} + \frac{\bar{y}_i^2}{b(t)^2} - 1\right)^2\right]^{1/2}, \tag{29}$$

where $(\bar{x}_i, \bar{y}_i)$ is the location of the numerical boundary point $\mathbf{x}_i$ rotated by $-\phi$. Figure 7 shows the error ε_r of our initial-value calculations of three Kirchhoff ellipses with different λ and $A = 1$ and $\Delta q = 1$. Here we have chosen N, the number of Lagrangian boundary points, equal to M the number of collocation points used in computing $\mathbf{v}_M(\mathbf{x}_i)$ and τ as in (19). The error is shown after integrating with the Runge–Kutta method for $\frac{1}{8}$ turnaround time with 2000 time steps. Because the boundary parameterization τ in this case is exact, the errors of this initial-value calculation closely follow the quadrature errors ε_v in Fig. 6. In Both Figs. 6 and 7, the errors decrease exponentially with M at approximately the same rate until they reach roundoff error. The roundoff error in ε_v is approximately 2000 times smaller than roundoff error in ε_r which accumulates over 2000 time steps.

We compare our initial-value results with Dritschel's third-order method. Dritschel integrated the equations of motion for a Kirchhoff ellipse with $A = \Delta q = 1$ and calculated the error in the position of the Lagrangian boundary points for different

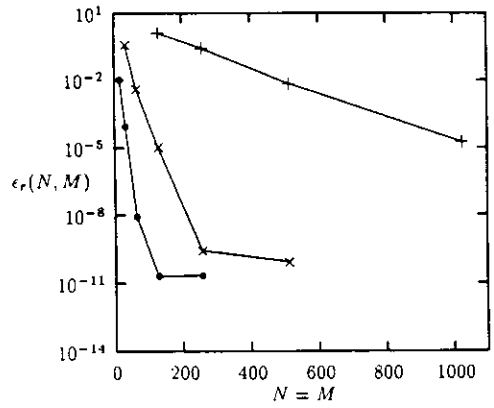

FIG. 8. Same as Fig. 7, but the initial τ equals θ the azimuthal polar coordinate, and we plot errors for (●) 2 : 1, (×) 5 : 1, and (+) 15 : 1 ellipses. The large value of ε_r for the 15 : 1 ellipse is not due to numerical error, but to the fact that the initial-value code follows the unstable solution rather than the steadily rotating Kirchhoff solution.

aspect ratios. Generally, for 256 Lagrangian boundary points, his error is less than 10^{-6} for aspect ratios between 2 and 10. (His measure of error is slightly different from ours, but if his is multiplied by $2\lambda^{1/2}$ they are equivalent to first order in the error). We repeated his calculations with our spectral method for a 10 : 1 Kirchhoff ellipse and found that for $N = M = 64$ our error is below 10^{-7}. Since the computational work of an initial-value code is proportional to NM and because the spectral method has one-sixth the computational work as Dritschel's (see Section 4), the spectral method for a 10 : 1 Kirchoff ellipse requires one-hundredth the computational work per time step as Dritschel's for the same accuracy.

We have repeated our spectral calculations of the Kirchhoff ellipse for a different initial τ (i.e., we rezoned the initial condition). Instead of defining τ as in (19) (which represents the initial boundary exactly), we set $\tau = \theta$, the azimuthal polar coordinate. In this case, the results are more complicated because the boundary is no longer represented exactly. Figure 8 shows the error ε_r as a function of resolution for the three ellipses whose initial boundary representation error ε_b are illustrated in Fig. 1. The error ε_r for the 2 : 1 Kirchhoff ellipse follows the boundary representation error ε_b very closely because the quadrature error ε_v is much smaller than ε_b. For the 5 : 1 ellipse, the error ε_r in the initial-value calculation is slightly larger than the error ε_b in Fig. 1, and the slope of ε_b is about 15% steeper than the slope of ε_r. For the 15 : 1 ellipse the error ε_r is much greater than ε_b due to the initial discretization of the boundary. Although some of this is due to quadrature, most is due to the fact that the 15 : 1 ellipse is linearly unstable and our initial-value code follows the instability rather than the steadily rotating Kirchhoff solution. The instability is more pronounced in the 15 : 1 ellipse of Fig. 8 than it is in the 20 : 1 ellipse of Fig. 7 because the initial amplitude of the unstable mode is much larger in the former due to the fact that the latter has no initial boundary discretization error (except for roundoff) and therefore no initial amplitude for the unstable eigenmode. Another way in which the error of the initial-value is measured is by examining the constancy of the conserved quantities. For example, we have computed the area A as a contour integral using

$$A(t) \equiv \iint dxdy = \oint x \frac{dy}{d\tau} d\tau. \tag{30}$$

The fractional errors in A, $[A(t) - A(0)]/A(0)$, for the 2 : 1 and 5 : 1 ellipses in Fig. 8 are always much smaller than their corresponding $\varepsilon_r(N, M)$. For the 2 : 1 ellipse the fractional error in A is three orders of magnitude smaller than $\varepsilon_r(N, M)$ for $N = M = 32$ and for $N = M = 64$.

5.2. *Test of Oscillating Kida Ellipse*

We also tested our code with an oscillating Kida ellipse with $A = 10$, $\Delta q = 1$, and $\Omega = \gamma = \frac{1}{2}$. At $t = 0$, the ellipse is aligned with the x-axis and has $\lambda = 2$. The initial positions of the marker points are given by (19). Table I shows the error ε_r defined as in (29), where $M = N$. The period T of the oscillation is 6.956. The CD does not show any sign of breaking down after three million iterations with $\Delta t = 0.002$. Table I shows that the error is small, decreases exponentially with increasing N (until it reaches roundoff), and mimics the velocity error shown in Fig. 5. The area of the Kida ellipse is conserved in time, and its fractional change in our numerical integration is given in Table II which shows that it is similar in magnitude and behavior to ε_r in Table I. The fractional changes in the energies of the Kida ellipse are approximately the square root of the fractional changes in areas given in Table II, and the fractional change in momentum $\iint y\bar{q}\, dxdy$ is always of the order of 10^{-15} machine roundoff.

5.3. *Test of 2-Contour Vortices with Finite L_r*

For finite L_r we first tested our code by examining the analytically obtainable eigenmodes of an equilibrium consisting of two concentric, circular contours with $\bar{\mathbf{v}} = 0$. The two contours

TABLE I

Error ε_r as Defined in Eq. (29) for the Kida Ellipse Described in the Text, Where the Oscillation Period is 6.956

N	$t = 5$	$t = 50$	$t = 500$
8	3.1×10^{-4}	3.0×10^{-4}	3.0×10^{-3}
16	9.4×10^{-7}	7.8×10^{-7}	2.9×10^{-7}
32	4.1×10^{-12}	3.1×10^{-12}	1.9×10^{-12}
64	3.3×10^{-13}	5.8×10^{-13}	2.1×10^{-12}

TABLE II

Fractional Change $(A(t) - A(0))/A(0)$ for the Kida Ellipse in Table I; for $N > 64$ Roundoff Dominates the Error

N	$t = 5$	$t = 50$	$t = 500$
8	1.9×10^{-5}	8.5×10^{-5}	3.5×10^{-5}
16	1.8×10^{-7}	3.8×10^{-7}	2.8×10^{-7}
32	4.7×10^{-12}	6.4×10^{-12}	6.7×10^{-11}
64	5.7×10^{-13}	6.6×10^{-13}	7.0×10^{-11}

have areas $A_1 = 25$, $A_2 = 4$, with jumps in potential vorticity $\Delta q_1 = 15$ and $\Delta q_2 = -2$, and $L_r = 1$. The perturbation eigenfunction of the two contours is

$$r_1(\phi, t) = \varepsilon \,\text{Real}[e^{im(\phi - st)}] \tag{31}$$

$$r_2(\phi, t) = \varepsilon \,\text{Real}[\delta\, e^{im(\phi - st)}]. \tag{32}$$

For $m = 4$, $\delta = 0.5387891$ with phase speed $s = 0.961066081$. The code was initialized with the $m = 4$ neutrally stable eigenmode with $\varepsilon = 10^{-5}$ and equal azimuthal spacing in the boundary points. The eigenmode has period $T = |2\pi/ms| \simeq 6.54$. The error ε_R, defined as the L_2 norm between the numerically computed locations of the boundary points and their values from Eqs. (31)–(32) is similar in magnitude to the values of ε_r in Table I for the Kida ellipse. It decreases exponentially with N and increases linearly with time.

To test the code with a nonlinear flow whose evolution cannot be obtained in closed form we examined the two-contour vortex shown in Fig. 9 which has $\bar{\mathbf{v}} = -y\hat{\mathbf{x}}$, $A_1 = 25$, $A_2 = 4$, $\Delta q_1 = 15$, $\Delta q_2 = 2$, and $L_r = 1$. The initial boundaries are 2 : 1 ellipses (which are not an equilibrium) plus large noise in azimuthal modes with $m < 10$. The flow remained computable without need for contour surgery for as long as we integrated in time (over three million iterations.) The areas circumscribed by the two contours A_1 and A_2 should each be conserved in time. Table III shows their fractional errors and demonstrates that they are small and decrease exponentially with N.

5.4. *Test of the Unstable Eigenmodes of a Kirchoff Ellipse*

Our initial-value code also captures the unstable eigenmodes of a 4 : 1 Kirchhoff ellipse with infinite L_r. Love [23] was the first to calculate the linear stability of rotating Kirchhoff ellipses. Define $(\tilde{x}, \tilde{y})$ as the Cartesian coordinates in the frame rotating with the same angular velocity as the ellipse (with the major axis along $\tilde{x}$), a and b as the semi-major and semi-minor axes, and (ξ, η) as the rotating elliptical coordinates

$$(\tilde{x}, \tilde{y}) \equiv c(\cosh(\xi)\cos(\eta), \sinh(\xi)\sin(\eta)) \tag{33}$$

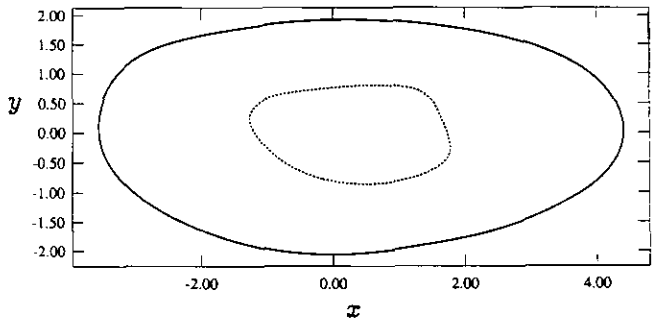

FIG. 9. The nonlinear evolution of an equilibrium consisting of two concentric circular contours with $L_r = 1$, areas $A_1 = 25$, $A_2 = 4$, $\Delta q_1 = 15$, $\Delta q_2 = 2$, and $N = M = 128$ at $t = 40$. The CD does not break down.

with $c^2 \equiv a^2 - b^2$. The location of the perturbed boundary of the Kirchoff ellipse is given by

$$\xi(\eta, t) = \xi_0 + \frac{\delta}{h_0^2}\exp(\alpha t)\,[\cos(m\eta) - \sqrt{d_m/c_m}\sin(m\eta)] + O(\delta^2), \tag{34}$$

where

$$c_m \equiv \frac{\Delta q}{2}\left[\frac{2m\lambda}{(\lambda+1)^2} - 1 + \left(\frac{\lambda-1}{\lambda+1}\right)^m\right] \tag{35}$$

$$d_m \equiv \frac{\Delta q}{2}\left[\frac{2m\lambda}{(\lambda+1)^2} - 1 - \left(\frac{\lambda-1}{\lambda+1}\right)^m\right] \tag{36}$$

$$h_0^2 = [a^2\sin^2(\eta) + b^2\cos^2(\eta)]/c^2, \tag{37}$$

where $\xi_0 \equiv \ln[(a+b)/(a-b)]/2$ is the location of the unperturbed boundary, δ is the initial amplitude of the perturbation, $\alpha \equiv \sqrt{-c_m d_m}$, and m is an integer. For our test of a 4 : 1 ellipse with $\Delta q = 1$, we pick the initial parameterization to be $\tau = \eta$ and the initial perturbation of the boundary to be the unstable $m = 3$ eigenmode given by (34) with $\delta = 3 \times 10^{-7}$. Using Runge–Kutta, we evolve the boundary to $t = 10$, taking 200 time steps. The growth rate α is 0.106. At $t = 10$ we define the error $\varepsilon_\xi(N, M)$ as

TABLE III

Fractional Change in Area of the Outer Boundary of the Two Contours Shown in Fig. 9 $|A_1(t) - A_1(0)|/A_1(0)$

N	$t = 0.1$	$t = 1.0$	$t = 10.$
16	6.9×10^{-4}	7.6×10^{-3}	6.9×10^{-2}
32	2.0×10^{-6}	4.7×10^{-5}	3.7×10^{-4}
64	5.4×10^{-10}	2.0×10^{-8}	4.2×10^{-7}
128	1.4×10^{-11}	1.8×10^{-10}	1.8×10^{-9}

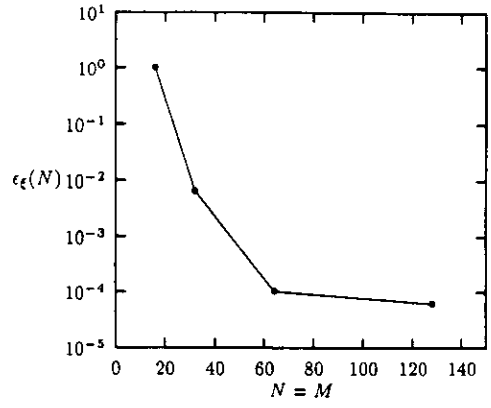

FIG. 10. The error $\varepsilon_\xi(N, M)$ which measures the difference between the analytically known unstable, $m = 3$, linear perturbation of the boundary of the Kirchhoff 4 : 1 ellipse at $t = 10$ and its numerically computed, nonlinear value. Here, $N = M$. Nonlinear effects that invalidate the linear theory dominate the numerical error when $\varepsilon_\xi \geq 10^{-4}$.

$$\varepsilon_\xi(N, M) = \frac{\{[\sum_{j=1}^{N}(\xi_N(\tau_j) - \xi(\tau_j))^2]/N\}^{1/2}}{\{[\sum_{j=1}^{N}(\xi(\tau_j) - \xi_0)^2]/N\}^{1/2}}, \tag{38}$$

where ξ_N is the numerical solution and ξ is the analytic linear solution. Figure 10 shows the error as a function of N. It illustrates that with $N = M = 64$ we resolve the linear dynamics. Increasing N to 128 points does not appreciably change the error. For $N = M = 64$ the numerator in (38), which is a measure of both the numerical errors in the initial-value code and the error due to approximating the instability with linear theory, is less than 10^{-9} while the denominator which is a measure of the amplitude of the linearly calculated perturbation is approximately 10^{-5}. The nonlinear corrections to linear theory are of order $(e^{\omega t}\delta/h_0^2)^2$ which is 5×10^{-10} at $t = 10$, so a fully resolved numerical calculation cannot produce a numerator less than 5×10^{-10}. Thus $\varepsilon_\xi(N, M)$ at $t = 10$ for a fully resolved numerical calculation is between 10^{-4} and 10^{-5}.

Zou *et al.* examined the linear instability of this same 4 : 1 ellipse by initializing their second-order calculation with both the stable and unstable $m = 3$ eigenmodes. We have reproduced their calculation with our spectral initial-value code and found for $\Delta t = 0.164$ and $A = \pi$ that at $t = 6.545$ ($\frac{1}{8}$ rotation period for a 4 : 1 ellipse) our code with $N = M = 32$ has a value of error ε_ξ that is smaller than theirs when they used 128 boundary points for the same boundary point distribution.

5.5. *Test of an Elongating Kida Ellipse*

By computing the Kida ellipse with $\Delta q = 1$, $\bar{\mathbf{v}} = y\hat{\mathbf{x}}$, with the major axis aligned initially along the y-axis and with the initial aspect ratio of 2 : 1, we have tested our code's ability to compute vortices that are stretched to infinity by $\bar{\mathbf{v}}(y)$ but that do not shed filaments. The ellipse does not oscillate but elongates continuously. The aspect ratio λ increases without bound as time increases as shown in (27). This calculation must break down. It is used to test where the CD method fails as a function of the average spacing between boundary points (or collocation points) with respect to the distance between the two nearly parallel contours that make up the opposite edges of a highly elongated ellipse. Figure 11 shows the evolution of ε_r with $N = M$ up to $t = 4$ when $\lambda = 45.72$. The fractional error in the area is nearly identical to ε_r. We find that ε_r increases beyond 0.01 when λ increases beyond $N/2$. This means that the calculation breaks down when the spacing between boundary or collocation points is a little larger than the width of the elongated vortex. This is not surprising and can be understood by a simple argument. Approximate a highly elongated ellipse of constant q' with a vortex layer where the boundary contours are parallel lines separated by distance d. Let the layer be parallel to the x-axis with the two contours at $y = 0$ and $y = d$. From (21), $\mathbf{v}'$ at the origin (or any other location on the lower boundary) is

$$\mathbf{v}' = \frac{q'\hat{\mathbf{x}}}{2\pi}\int_{-\infty}^{\infty}\frac{d^2}{x'^2 + d^2}\,dx' = \frac{q'd\hat{\mathbf{x}}}{2}. \tag{39}$$

With the collocation points distributed uniformly with a spacing of Δx, the numerical approximation of (39) becomes

$$\mathbf{v}'(x, y = 0, \Delta x, h) = \frac{q'\hat{\mathbf{x}}}{2\pi}\frac{d^2}{\Delta x}\sum_{n=-\infty}^{\infty}\frac{1}{(n + h)^2 + (d/\Delta x)^2}; \quad -\frac{1}{2} < h < \frac{1}{2}, \tag{40}$$

where the collocation point nearest the y-axis is a distance $h\Delta x$ away (hence h is the relative phase of the collocation points).

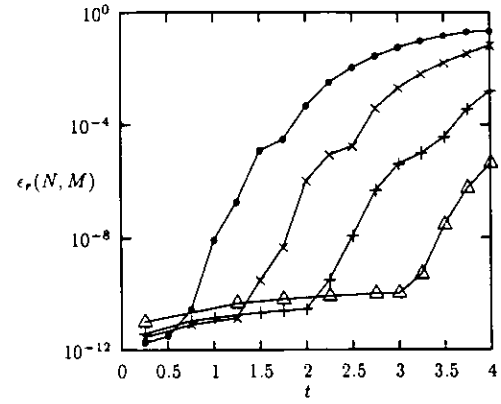

FIG. 11. The error ε_r as a function of time for a continuously elongating Kida ellipse with $\Delta q = 1$ in a $\bar{\mathbf{v}}$ with an adverse shear of -1 for (●) $N = 32$, (×) $N = 64$, (+) $N = 128$, and (Δ) $N = 256$. All calculations have $N = M$.

The sum in (40) can be evaluated in closed form for $h = 0$ and $h = \frac{1}{2}$,

$$\mathbf{v}'(x, y = 0, \Delta x, h = 0) = \frac{q' d\hat{\mathbf{x}}}{2} \coth(\mu) = \frac{q' d\hat{\mathbf{x}}}{2}(1 + 2e^{-2\mu} + \cdots) \tag{41}$$

$$\mathbf{v}'(x, y = 0, \Delta x, h = 0.5) = \frac{q' d\hat{\mathbf{x}}}{2} \tanh(\mu) = \frac{q' d\hat{\mathbf{x}}}{2}(1 - 2e^{-2\mu} + \cdots), \tag{42}$$

where $\mu \equiv \pi d/\Delta x$. Hence, in this simple case we can prove that our approximate quadrature converges exponentially in (Δx^{1}) to the exact velocity. Furthermore, we find that when the collocation point spacing Δx is equal to the distance between contours d, that the relative magnitude of the leading order error term is 0.004, in reasonable agreement with our earlier observation that ε_r is approximately 0.01 in our initial-value method when the collocation point spacing is equal to the width of a highly elongated ellipse.

5.6. *Test of a Vortex with Tails at the Stagnations Points*

As an example of a flow that produces tails at the stagnation points but does not filament, we consider the vortex in Fig. 12 which initially is a circular contour with area 30, $\Delta q = 1$, infinite L_r, center at the origin, and $\bar{\mathbf{v}}(y) = (-y + y^3/3)\hat{\mathbf{x}}$. The $\bar{\mathbf{v}}$ advects the top and bottom of the vortex to $\pm\infty$. Every 20th iteration (with $\Delta t = 0.05$) marker points located at $|x| > 9.5$ were removed, and the remaining points were reconnected using the Fourier interpolation of Eqs. (15)–(17) and then rezoned with equal arc-lengths between points. This calculation is representative of the geophysical vortices reported in [18], where the vortices relax to equilibria by the dissipation implicit in the removal of the outlying sections of the vortex tails. The physical motivation for that approximation is discussed in [20]. Here we confine ourselves to showing that the numerical algorithm is accurate when the tails are removed. The flow in Fig. 12 has no analytic solution, but the area and reduced energy are conserved during the periods between the times that sections of tail are removed. The reduced energy is defined [20]

$$E \equiv -\Delta q \iint \bar{\psi}\, dxdy + \frac{(\Delta q)^2}{32\pi} \oint\oint r^2[\ln(r^2) - 1](dxdx' + dydy') + \frac{(\Gamma)^2}{8\pi}\left(\frac{1}{2} + \ln\frac{\Gamma}{\pi}\right), \tag{43}$$

where $r^2 = (x - x')^2 + (y - y')^2$ and Γ is the potential circulation. (The reduced energy is the difference in energy between the actual flow and the flow if all of the potential circulation were in a circular contour.) The fractional errors in area and reduced energy at a late time are given in Table IV. The errors are small and decrease exponentially with N. A more stringent test is to use Eq. (43) to compute ΔE_l, the change in reduced energy at the l^{th} time that a section of tail is removed. After L sections of tail have been removed energy balance requires

$$\left| E(0) - E(t) - \sum_{l=1}^{L} \Delta E_l \right| / E(0) = 0. \tag{44}$$

For the flow in Fig. 12 at $t = 26$ and $L = 24$ the left-hand side of Eq. (44) is 1.13×10^{-2}, 3.68×10^{-3}, and 7.00×10^{-4} for N equal 512, 1024, and 2048 which shows that the error is small and decreases approximately exponentially with N. Note that $t = 26$ is considered a late time because the reduced energy at that time is approximately one half of its initial value.

6. CALCULATION OF STEADY SOLUTIONS

In this section we develop a Newton, pseudo-arclength, continuation method using spectral techniques for computing steady, uniformly translating, and rotating equilibria. The method is particularly useful for computing bifurcation diagrams for stable and unstable equilibria. Other authors have also developed methods to calculate steady CD solutions. Pierrehumbert's relaxation algorithm [24] in particular could be easily modified to be spectral. The computational work per relaxation step for Pierrehumbert's method and Wu *et al.*'s method [13] scale as N^2 but neither method computes the linear stability of the equilibria or their bifurcations as our continuation method does. Moreover, despite the fact that the work of a continuation method scales as N^3, we have found that our continuation method converges quickly (usually less than 10 iterations, compared to about 100 in [13]), and because our method is spectral it requires about four times fewer boundary points than finite-difference methods for the same accuracy.

6.1. *Boundary Advection Condition*

The condition that a vortex be steady in some frame implies that the component of the velocity normal to the contour at the contour is zero. Using (20) this gives the following condition on $x(\tau)$ and $y(\tau)$:

$$\left(\frac{dy}{d\tau}\hat{\mathbf{x}} - \frac{dx}{d\tau}\hat{\mathbf{y}}\right) \cdot [\bar{\mathbf{v}}(\mathbf{x}) + \Delta q \oint [\mathbf{x} - \mathbf{x}'(\tau)]\dot{G}[|\mathbf{x} - \mathbf{x}'(\tau)|] \frac{d[|\mathbf{x} - \mathbf{x}'(\tau)|]}{d\tau} d\tau \equiv 0. \tag{45}$$

We can multiply (45) by any function of τ that does not pass through zero to get an equivalent condition.

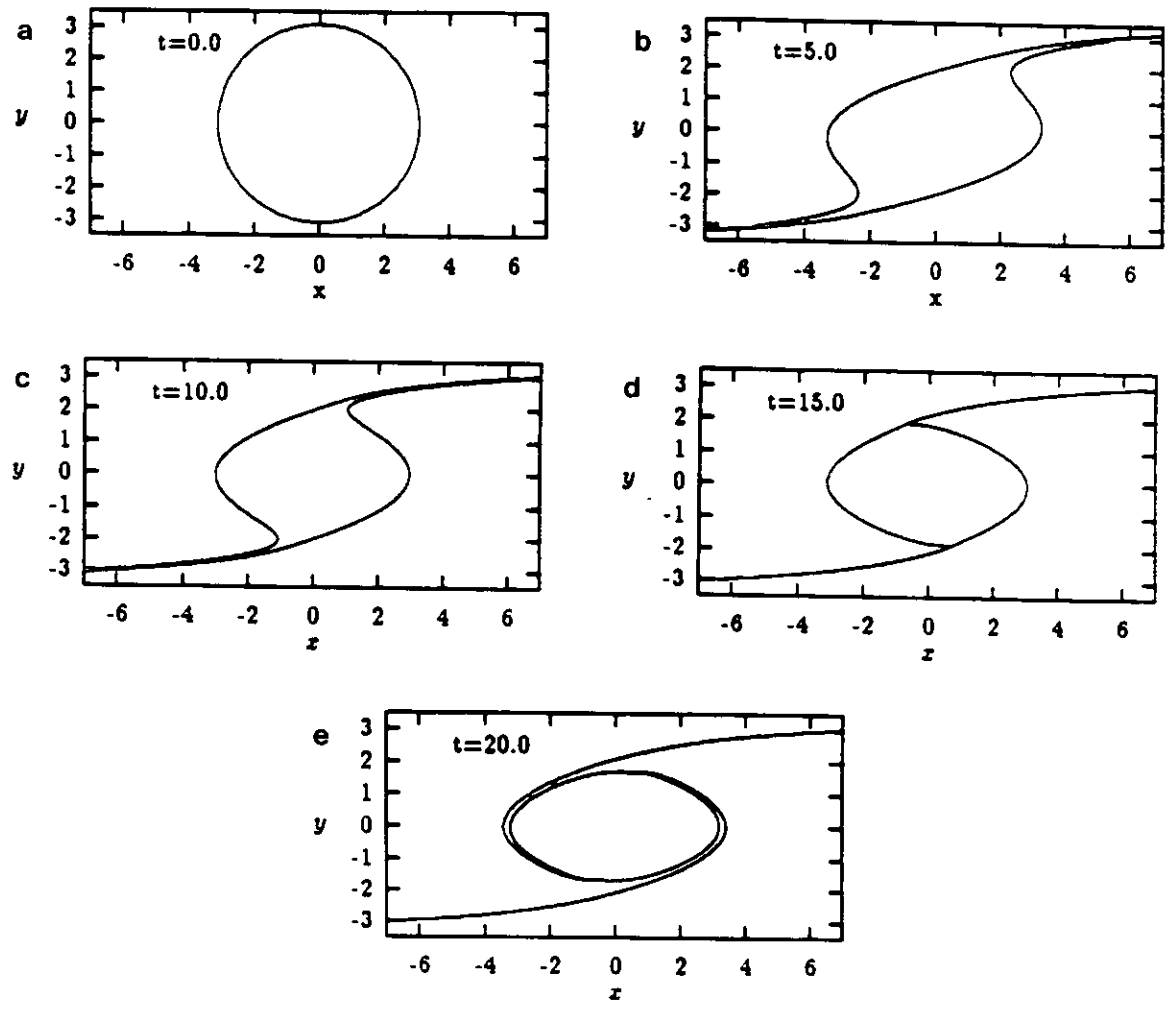

FIG. 12. The evolution at five different times of the boundary contour of an initially circular vortex with $A = 30$, $\bar{\mathbf{v}} = (-y + y^3)\hat{\mathbf{x}}$, and $\Delta t = 0.5$. Every 20 time-steps potential vorticity in the two vortex tails at $|x| > 9.5$ is removed.

The initial-value method in Section 5 advects the N boundary points as passive tracers. Clearly, the advection of a contour does not depend on the tangential component of the velocity at the contour. Hence, even though our boundary points have both tangential and normal degrees of freedom, we are interested in only changes in the latter. Therefore in our steady-state solver, we constrain the boundary points to move along a set of fixed lines that we shall call transversals. The direction of these transversal lines are chosen arbitrarily, fixed for all iterations, and defined below. Define the $2N$ real unknowns that specify the j^{th} iterate of the i^{th} boundary point$(x(\tau_i, j), y(\tau_i, j)) \equiv \mathbf{x}_i^j$, where τ_i is defined as in (17) and where $\{\mathbf{x}_i^0\}_{i=1}^N$ is the initial location of the boundary points. Label the N unit-length transversal vectors $\hat{\mathbf{t}}_i$, where $\hat{\mathbf{t}}_i \cdot \hat{\mathbf{n}}_i \neq 0$ and $\hat{\mathbf{n}}_i$ is the local unit normal to the boundary. The N conditions that constrain the points to move along transversal lines that intersect the initial values $\mathbf{x}_i^0$ are

$$\hat{\mathbf{t}}_i \times \Delta\mathbf{r}_i^j \equiv 0, \tag{46}$$

where $\Delta\mathbf{r}_i^j \equiv \mathbf{x}_i^j - \mathbf{x}_i^0$. The condition that the boundary is steady becomes

$$F(\tau_i) \equiv F_i \equiv \left.\frac{d\mathbf{x}_i}{dt}\right|_{\hat{\mathbf{t}}_i \times \Delta\mathbf{r}_i = 0} = \frac{\mathbf{v}(x(\tau_i), y(\tau_i)) \cdot \hat{\mathbf{n}}_i}{\hat{\mathbf{t}}_i \cdot \hat{\mathbf{n}}_i} = 0. \tag{47}$$

Equation (47) shows that the definition of $\hat{\mathbf{t}}_i$, as long as $\hat{\mathbf{t}}_i \cdot \hat{\mathbf{n}}_i \neq 0$, does not effect equilibria, but its choice will determine the rate and radius of convergence of our method. In practice some care must be taken in choosing the $\hat{\mathbf{t}}_i$, because some choices imply boundary constraints that cannot be satisfied by the steady solutions. An example of this is setting $\hat{\mathbf{t}}_i$ to a set of unit radial vectors. With this choice, if we begin with a boundary that encircles the origin and is single-valued in θ,

TABLE IV

Fractional Change in Area and Reduced Energy at Late Times in the Vortex Evolution Shown in Fig. 12

N	$\left\lvert\frac{A(t=16)-A(t=15)}{A(t=15)}\right\rvert$	$\left\lvert\frac{E(t=16)-E(t=15)}{E(t=15)}\right\rvert$
256	1.2×10^{-4}	3.2×10^{-4}
512	9.2×10^{-5}	2.7×10^{-4}
1024	2.6×10^{-5}	8.7×10^{-5}
2048	2.1×10^{-6}	1.7×10^{-5}

Note. There is no removal of vortex tail and no rezoning between $t = 15$ and $t = 16$.

then the boundary is constrained to remain single-valued. If we attempt to calculate a solution that is not single-valued in θ, the denominator in (47) goes to zero. When that happens, we find that either the Newton method does not converge, or it converges to a non-physical solution that has a sharp jump in the radius as a function of θ.

A second problem due to a bad choice of $\hat{\mathbf{t}}_i$ arises if the transversals cross each other near the boundary. When transversals cross on the boundary, the boundary develops a figure-8 pattern (which is unphysical) or is double-valued in our parameterization. In either case, the denominator in (47) approaches zero and the method either fails to converge or converges to an unphysical solution.

To solve (47) with a Newton method, we would first compute the Jacobian,

$$(DF)_{ik} \equiv \left(\frac{\partial F_i}{\partial x_k}\hat{\mathbf{x}} + \frac{\partial F_i}{\partial y_k}\hat{\mathbf{y}}\right) \cdot \hat{\mathbf{t}}_i, \tag{48}$$

where the $\partial F_i/\partial x_k$ and $\partial F_i/\partial y_k$ are calculated explicitly from the numerical formula for F_i in (47) and where (48) is not summed over the index i. Thus (DF) is an $N \times N$ matrix. Although it appears that there are $2N$ unknowns, x_i^j and y_i^j at each iteration j, there are in fact only N scalar unknowns: $\Delta r_i^j \equiv |\Delta \mathbf{r}_i^j| = |\mathbf{x}_i^j - \mathbf{x}_0^j|$. Or equivalently, the $2N$ scalars in $\mathbf{x}_i^j$ can be recovered from the N scalars Δr_i^j (and their previous iterate) from

$$\mathbf{x}_i^j = (\Delta r_i^j - \Delta r_i^{j-1})\hat{\mathbf{t}}_i + \mathbf{x}_i^{j-1}. \tag{49}$$

The N scalars Δr_i^j are determined with the usual Newton method

$$\Delta r_i^j = \Delta r_i^{j-1} - \sum_{k=1}^{N} (DF)_{ik}^{-1} F_i, \tag{50}$$

where F_k is determined by (47). In practice, the Jacobian given by (48) is singular, so it cannot be used to compute numerical solutions. It must be modified as we now show.

6.2. *Iteration with Constraints*

In general the condition (47) is not sufficient for determining a unique solution because it does not conserve the area, momentum, or energy. For the Newton equation to be well posed, new constraints that conserve area and one or more of the momenta must be added. When the conditions in (47) are augmented with these constraints, the steady, uniformly translating, or rotating solutions no longer form a continuum but are distinct from each other. Without these additional constraints the Jacobian (DF) in (48) is singular. For example, when $\bar{\mathbf{v}} = 0$, the solution can be translated by an infinitesimal amount in any direction and still remain a solution. The invariant translations correspond to eigenvectors of (DF) with zero eigenvalue, hence (DF) is singular.

For a zonal $\bar{\mathbf{v}}$, i.e., $\bar{v}_y = 0$ and $\partial \bar{v}_x/\partial x = 0$ (or $\bar{v}_x = 0$ and $\partial \bar{v}_y/\partial y = 0$), constraining the center of vorticity and the area interior to the contour is sufficient to make the Newton equations well-posed. The constraints to be added to (47) are

$$F_{N+1} \equiv \frac{1}{A}\iint y\,dxdy - \bar{y}_0 = \frac{1}{A}\oint xy\frac{dy}{d\tau}d\tau - \bar{y}_0 = 0 \tag{51}$$

$$F_{N+2} \equiv \frac{1}{A}\iint x\,dxdy - \bar{x}_0 = \frac{1}{A}\oint xy\frac{dx}{d\tau}d\tau - \bar{x}_0 = 0 \tag{52}$$

$$F_{N+3} \equiv \iint dxdy - A = \oint x\frac{dy}{d\tau}d\tau - A = 0, \tag{53}$$

where the contour integrals are evaluated spectrally using M collocation points, where the center of potential vorticity is constrained to be $(\bar{x}_0, \bar{y}_0)$, and where the area is constrained to be A. Note that for a zonal $\bar{\mathbf{v}}$, (1) conserves the momentum in the x (or y) direction which is proportional, up to a constant, to F_{N+1} (or F_{N+2}).

For the Newton method, the Jacobian (DF) must be maintained as a square (and invertible) matrix. Thus we must add a new variable to the velocity for every new constraint that we enforce. Calling these new variables V_{N+1}, V_{N+2}, and V_{N+3}, the total velocity $\mathbf{v}(\mathbf{x})$ becomes

$$\mathbf{v}(\mathbf{x}) = (V_{N+1} + V_{N+3}x)\hat{\mathbf{x}} + (V_{N+2} + V_{N+3}y)\hat{\mathbf{y}} + \bar{\mathbf{v}}(\mathbf{x}) + \mathbf{v}'(\mathbf{x}), \tag{54}$$

where $\bar{\mathbf{v}}(\mathbf{x})$ and $\mathbf{v}'(\mathbf{x})$ are as defined in Section 2. The V_{N+1}, V_{N+2}, and V_{N+3} add a translation in x, in y, and a divergence to the velocity. We choose these as supplementary unknowns because these three additional velocity fields are complementary to the constrained quantities. The V_{N+1} changes $\bar{x}_0$, V_{N+2} changes $\bar{y}_0$, and V_{N+3} changes the area. It is easily shown that if $\nabla \cdot \bar{\mathbf{v}} = 0$ and if the Newton method converges to a fixed point, then V_{N+3} must converge to zero, so $\nabla \cdot \mathbf{v}' = 0$. Moreover, if $\bar{v}_y = 0$ and $\partial \bar{v}_x/\partial x = 0$ (or $\bar{v}_x = 0$ and $\partial \bar{v}_y/\partial y = 0$) then V_{N+2} (V_{N+1}) converges to zero. A finite value of V_{N+1} (V_{N+2}) corre-

sponds to a uniformly translating vortex in the x (y) direction. (For an azimuthal $\bar{\mathbf{v}}$, where $\bar{v}_r = 0$ and $\partial \bar{v}_\theta / \partial \theta = 0$, the initial-value equations conserve the angular momentum. The constraints needed to augment(47) and the new variables are more complicated and are discussed in the Appendix). We have found that the Newton method with the $(N + 3) \times (N + 3)$ Jacobian corresponding to (47) augmented with (51)–(53) is robust and converges quadratically in less than 10 iterations.

6.3. *Refinements of the Newton Method*

Instead of solving Eqs. (47) and (51)–(53) directly, we solve their Fourier transform. The robustness of the method can be significantly enhanced by modifying the equations so that the equation for the highest Fourier mode of F_i is replaced with the constraint that the highest Fourier mode of the boundary shape $\mathbf{x}(\tau)$ remain zero for all iterations, or equivalently,

$$\sum_{i=1}^{N} (-1)^n \Delta r_i \equiv 0. \tag{55}$$

These modified equations damp the high wave number oscillations of the boundary without noticeable loss of accuracy and give the method a much larger radius of convergence.

A second change that we have made to our Newton algorithm is that we have made it into a pseudo-arc-length continuation method so that we can follow bifurcation curves around saddle nodes [25]. We take unequal step sizes in the bifurcation parameter, where its $(l + 1)^{\text{th}}$ iteration α^{l+1} is determined from its previous value α^l by

$$\left[B \sum_{i=1}^{N} |\mathbf{x}_i(\alpha^{l+1}) - \mathbf{x}_i(\alpha^l)|^2 + (\alpha^{l+1} - \alpha^l)^2 \right]^{1/2} - \Delta S = 0, \tag{56}$$

where B and ΔS are parameters that we choose for numerical convenience. By taking constant ΔS steps rather than constant $\Delta\alpha$ steps, we can follow families of solutions around turning points.

6.4. *Convergence and Errors*

To illustrate the nature of our spectral continuation method we have used it to calculate known analytic solutions and their properties. Here we compare our numerical solutions to the Moore–Saffman elliptical steady vortices for the case where the shear of $\bar{\mathbf{v}}$ is opposite (adverse) to Δq [26].

Let M be the number of collocation points $\hat{\tau}_m$ in computing the contour integrals in (22) and let N be the number of boundary points $\mathbf{x}_i$. Then the number of operations required for evaluating the elements of the Jacobian is proportional to N^2M and for inverting the Jacobian proportional to N^3. We often find that the errors in our calculation are dominated by the quadrature errors. Because the quadrature error decreases exponentially with increasing M, we choose to set $M = 2N$ rather than $M = N$. This results in a two to three orders of magnitude decrease in error for an increase in a computation cost that is less than a factor of 2. We begin our test by computing an equilibrium vortex with $\Delta q = 10$ with $\bar{\mathbf{v}} = y\hat{\mathbf{x}}$ (so $\bar{\mathbf{v}}$ has a shear adverse to Δq). We march along the bifurcation curve with Δq initially decreasing. The method has no difficulty going through the saddle-node bifurcation at $\Delta q = 1/(3 - 2\sqrt{2})$. In these calculations we choose τ such that $\theta = \tau + \sin(2\tau)/2$ so that points are concentrated near the y-axis (i.e., where the ellipses have their largest curvature), and we choose our transversals $\hat{\mathbf{t}}_n$ to be radial vectors. This forces the boundary points to lie on radial rays equally spaced in τ. We choose our constraints such that $A \equiv 1$, and $(\bar{x}_0, \bar{y}_0) = (0, 0)$. We find that the auxiliary variables V_{N+1}, V_{N+2}, and V_{N+3} all converge to zero to within 10^{-12}. In all cases the algorithm converged in fewer than 10 iterations. Figure 13 shows ε_r defined by (29) as a function of the aspect ratio λ of the steady equilibrium ellipses on the unstable branch of solutions for various N. In all cases $M = 2N$. The figure shows that 64 boundary points are often sufficient for ellipses with $\lambda < 10$. For a highly elongated ellipse with $\lambda = 20$, ε_r is less than 10^{-3} for $N = 128$. The complicated structure of ε_r as a function of λ is a reflection of the complicated bifurcation behavior of these unstable Moore–Saffman ellipses. This branch of solutions has a sequence of linear instabilities as a function of λ, and the local extrema in Fig. 13 are near these bifurcation points because at each of these bifurcations, the Jacobian (DF) is singular and non-invertible [17, 18].

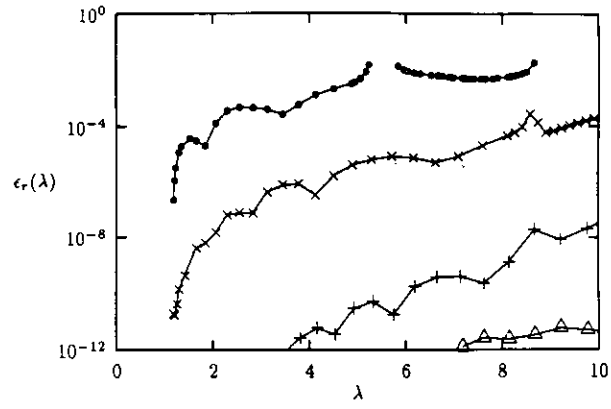

FIG. 13. The error $\varepsilon_r(N, M = 2N)$ of our continuation method for the branch of unstable Moore–Saffman ellipses with $\bar{\mathbf{v}} = y\hat{\mathbf{x}}$ (adverse shear). The errors are shown as a function of aspect ratio λ for (●) $N = 32$, (×) 64, (+) 128, and (Δ) 256. The complicated structures of the curves and the large (off-scale) value of the $N = 32$ curve near $\lambda = 5.5$ are because this branch of solutions has a sequence of bifurcations as a function of λ. The errors are large near each of the bifurcation points.

We have also tested our ability to calculate the linear stability of steady solutions. As described in Section 4, a Kirchhoff ellipse of aspect ratio λ is a steady solution with $\bar{\mathbf{v}} = -\Omega(-y\hat{\mathbf{x}} + x\hat{\mathbf{y}})$, where $\Omega = \Delta q\lambda/(\lambda + 1)^2$. In this test we examine the linear stability of a 4 : 1, $\Delta q = 1$, $A = 1$, Kirchhoff ellipse. We know analytically from Section 4 that the ellipse

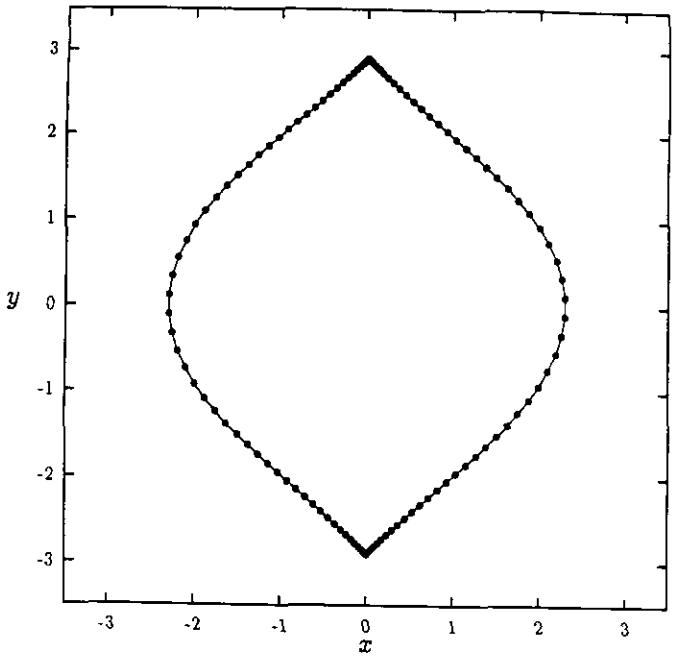

FIG. 14. The boundary shape used as the input into the inverse-solver. This boundary shape is the equilibrium output of our Newton code with $\bar{\mathbf{v}}(y) = (-y + y^3/3)\hat{\mathbf{x}}$, for $\Delta q = 15$, $A = 16.48087$, and $L_R = 1$. It has two corners at $x = 0$. The collocation points are plotted as circles.

has one stable and one unstable eigenmode with m = 3 and that all of the others are neutral. In our calculations we define τ such that $\theta = \tau + \sin(2\tau)/2$ and such that the major axis of the ellipse lies along the y-axis. We use N boundary points $\mathbf{x}_i$ of the ellipse in our numerical calculation of the elements of the *unaugmented* $N \times N$ Jacobian in (48). We find the fractional error of the numerically computed stable and unstable eigenvalues of (48) with respect to the analytically known eigenvalues are less than 10^{-1}, 10^{-5}, 10^{-10}, and 10^{-10} for $N = 32$, 64, 128, and 256, respectively (where $M = 2N$). The unstable eigenmode is known analytically from (34), and the error ε_ξ (defined in (38)) of our numerically computed eigenmode is 0.2, 1.4×10^{-3}, 7×10^{-7}, and 8×10^{-10} for $N = 32$, 64, 128, and 256, respectively (where $M = 2N$).

7. THE INVERSE PROBLEM

In the previous section, we outlined how to calculate the contour shape of a steady vortex, given $\bar{\mathbf{v}}$. In this section we examine the inverse problem: given the boundary and Δq of a potential vortex, find a $\bar{\mathbf{v}}$ (and a function β in (7)) that makes that vortex a steady equilibrium. This inverse problem is useful for several reasons. First, as we shall show, the solution $\bar{\mathbf{v}}$ can be solved for directly *without iteration*. Comparison between the solution found with the inverse method and the continuation method is yet another test of our code. A second interesting use of the inverse method is that we can input contours with singularities (e.g., boundaries with corners as in Fig. 14) into the code and find the $\bar{\mathbf{v}}$ that make them equilibria. We can then examine their stability by evolving them with our initial-value code. Finally, a third use of the inverse problem is in our studies of long-lived planetary vortices in which the bounding contour of a vortex is observed. Assuming the vortex has a constant potential vorticity, we can infer the $\bar{\mathbf{v}}$ of the surrounding flow [19].

To solve the inverse problem, we note that the condition (45) for a vortex to be steady can be rewritten as

$$\hat{\mathbf{n}} \cdot \bar{\mathbf{v}}(\mathbf{x}) = -\hat{\mathbf{n}} \cdot \mathbf{v}'(\mathbf{x}), \tag{57}$$

where $\hat{\mathbf{n}}$ is the unit normal vector of the boundary. Because we are given the location of the boundary, we can calculate both $\hat{\mathbf{n}}$ and $\mathbf{v}'$ (using (22)). Thus the right-hand side of (57) is known, and we can solve for the normal component of $\bar{\mathbf{v}}$ at every point along the boundary. When the functional form of $\bar{\mathbf{v}}$ is sufficiently constrained, we can uniquely calculate $\bar{\mathbf{v}}$ from its normal component. For example, we are often interested in zonal flows of the form $\bar{\mathbf{v}} \equiv \bar{v}(y)\hat{\mathbf{x}}$ and azimuthal flows $\bar{\mathbf{v}} \equiv \bar{v}(r)\hat{\theta}$. In these cases

$$\bar{v}(y) = -(\hat{\mathbf{n}} \cdot \mathbf{v}')/(\hat{\mathbf{n}} \cdot \hat{\mathbf{x}}) \tag{58}$$

and

$$\bar{v}(r) = -(\hat{\mathbf{n}} \cdot \mathbf{v}')/(\hat{\mathbf{n}} \cdot \hat{\theta}). \tag{59}$$

The inverse problem has a unique solution, except where $(\hat{\mathbf{n}} \cdot \hat{\mathbf{x}}) = 0$ or $(\hat{\mathbf{n}} \cdot \hat{\theta}) = 0$. There, the solution for $\bar{\mathbf{v}}$ is indeterminate. For these forms of $\bar{\mathbf{v}}$ certain solvability criteria must be met. For example, for $\bar{\mathbf{v}} = \bar{v}(y)\hat{\mathbf{x}}$ the solvability conditions are sufficiently satisfied when the boundary does not cross itself and is reflection symmetric about the y-axis. Furthermore, if the boundary has a discontinuous tangent vector (i.e., a corner), the discontinuity must lie on the y-axis (as in Fig. 14).

We have used the inverse problem to test the consistency of our continuation method by taking a steady-state solution calculated with the continuation method with an arbitrary zonal velocity $\bar{v}(y)\hat{\mathbf{x}}$ and then used that boundary as an input to the inverse problem and calculated the zonal velocity $\bar{v}_{\text{inverse}}(y)\hat{\mathbf{x}}$ that satisfies (54). The difference between the $\bar{v}(y)\hat{\mathbf{x}}$ that was input to the continuation method and the $\bar{v}_{\text{inverse}}(y)\hat{\mathbf{x}}$ output from the inverse problem is a measure of the consistency of our methods.

Figures 14 and 15 show the results of one such consistency check for a steady potential vortex with a $\bar{v}(y)\hat{\mathbf{x}}$ that has a non-uniform shear. Figure 14 shows the shape of the steady potential vortex found with the continuation method for $L_R = 1$, $A = 16.48$, $\Delta q = 15$, and $\bar{\mathbf{v}}(y) = -(y - y^3/3)\hat{\mathbf{x}}$. This is a severe test of both the continuation and inverse codes due to the two corners along the y-axis. Here $M = 2N = 256$, the parameterization τ is defined by $\theta = \tau + \sin(2\tau)/2$ and the $\hat{\mathbf{t}}_i$ are radial vectors. Figure 15 shows the difference between the original

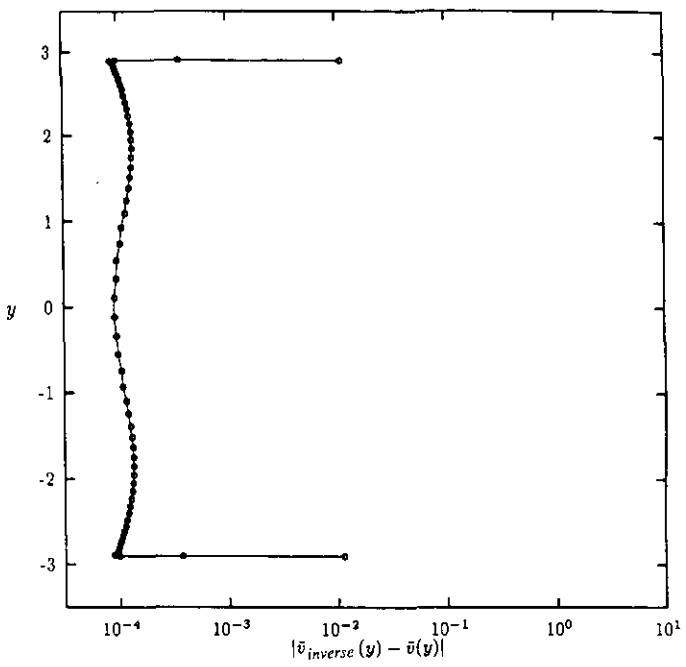

FIG. 15. The absolute value of the difference between the $\bar{v}(y)$ input to the Newton code and the $\bar{v}_{inverse}(y)$ output from the inverse-solver. The greatest error occurs at values of y near $(\hat{\mathbf{n}} \cdot \hat{\mathbf{x}}) = 0$, where the solution of the inverse-solver is indeterminate.

zonal profile and that calculated from (58) at each boundary point as a function of y. The difference is near 10^{-4} over much of the boundary except at the corners, where $\hat{\mathbf{n}} \cdot \hat{\mathbf{x}}$ approaches zero and the difference increases by a factor of 100. This result is consistent with our results from Section 2. Figure 4 shows that the relative error ε_b in representing the boundary of a "cat's-eye" vortex for this parameterization with $N = 128$ is 10^{-7}. If this boundary error is due to aliasing errors in the representation, then the relative error in the derivatives of the boundary should be approximately N times this or 10^{-5}. The normal vector in (58) is calculated by taking a derivative of the boundary, and hence we would expect the relative error in the right-hand side of (58) (from which we calculate $\bar{\mathbf{v}}(y)$) to be of order 10^{-5}. Because the zonal velocity is order 10, the fact that its absolute error is 10^{-4} is consistent with our argument that its relative error should be 10^{-5}.

8. CONCLUSION

The study of the detailed dynamics of inviscid vortices is a fundamentally difficult problem that no one computational method or theoretical technique can resolve. In this paper we have developed basic numerical techniques for studying the dynamics of piece-wise constant distributions of potential vorticity in two dimensions. For a variety of analytic test cases, we have demonstrated the superior performance of spectral methods over finite-difference methods. The improvement is due to the exponential convergence of spectral methods in both the representation of the boundary and the quadrature of the contour integral. We have applied these methods to a Newton continuation method that calculates stable and unstable, steady, uniformly translating, and rotating solutions as well as their linear eigenmodes. We use this method to find bifurcation branches and their limiting solutions [19, 20].

Our initial-value methods often work very well for studying the dynamics of potential vortices of strength Δq embedded in zonal flows $\bar{\mathbf{v}}$ that have the same sign shear as Δq and for flows with finite Rossby deformation radius L_r. (Our motivation for developing the methods in this paper was to study planetary vortices embedded in zonal flows [18].) The presence of $\bar{\mathbf{v}}$ or finite L_r often inhibits the formation of filaments. Typically in these flows the CD methods do not break down and spectral methods require an order of magnitude fewer points than finite-difference methods to obtain the same accuracy. Spectral methods also work very well for flows that form one or two large-scale tails that are advected by $\mathbf{v}$ to infinity but do not produce small-scale filaments. The idea behind the use of a spectral method for these vortices is that with the exception of the tail tips the contours are regular and therefore a high-order (spectral) method should be used to compute the flow except at the tips. At the tips far from the main bodies of the vortices, discontinuities (removals of $\tilde{q}$) are introduced in a physically motivated way to mimic the dissipation left out of the quasi-geostrophic equation (1), but the spectral method is never used to try to capture the discontinuity. (This is analogous to the computation of shocks in Eulers equation: A high-order method can be used with shock fitting, but shock capturing generally works best with a low-order method.)

For potential vortices that form small-scale filaments or large-scale tails that are not advected to infinity, accuracy is lost when the thickness of the filaments or tails is order $1/N$. Because the filaments thin out exponentially fast, a spectral method's factor of 10 in increased accuracy is not that large an advantage for these flows, and all CD methods fail within a few e-folding times after the filaments form. Attempts to save CD algorithms by modeling physical dissipation with contour surgery [9] or smoothing are controversial, especially when the amount of potential vorticity in the excised filaments becomes large. It is not obvious that contour surgery of filaments or tails close to the main bodies of the vortices mimics viscosity, nor has it been established how sensitive the late-time behavior of the vortices is to this mock dissipation; i.e., it has not been proved that the late-time solutions computed with dissipative CD algorithms converge to those of the Navier–Stokes equation in the inviscid limit. For example, two-dimensional turbulence and molecular viscosity diffuse potential vorticity and momenta while conserving their integrals, whereas the removal of filaments changes the values of the integrals. Thus solutions of this class of flows have not been discussed here. However, we note that there are many initial-value problems of astrophysical and geophysical interest that do not have these problems and

that can be computed with CD methods without controversy [18]. For these flows spectral methods appear to be more efficient than finite differences. Moreover, for steady-state, uniformly translating, or rotating solutions found with the Newton method, filamentation does not occur, and spectral CD methods are very practical.

Finally we note that our spectral method (although more accurate) requires no more storage or computational effort than finite-difference methods with the same number of points N. Therefore the same techniques that have been used to accelerate finite-difference CD methods from N^2 to $N \ln (N)$ operation per time step (e.g., the use of grids, multipole approximations, etc.) can also be used with these spectral methods.

This work was supported in part by DARPA/NSF Grant DMS-8919074 and NSF Grant CTS-89-06343. Calculations were done at the San Diego Supercomputer Center. We thank Tom Humphreys for help in the computations.

APPENDIX A: CONSTRAINTS FOR STEADILY ROTATING VORTICES

In this appendix we outline the constraints needed to make our Newton method well posed for computing the boundary contours of steadily rotating potential vortices when $\bar{\mathbf{v}}$ is of the form $\bar{\mathbf{v}} = \bar{v}(r)\hat{\theta}$. We remind the reader that for zonal flows of the form $\bar{\mathbf{v}} = \bar{v}(y)\hat{\mathbf{x}}$, three constraints are needed to specify uniquely each discrete branch of solutions. The three constraints correspond to the conservation of circulation or area and the initial choice of the center of vorticity (x_0, y_0). For $\bar{\mathbf{v}} = \bar{v}(r)\hat{\theta}$ there are five constraints, and they correspond to the conservation of circulation or area, the requirement that $(x_0, y_0) \equiv (0, 0)$, the conservation of angular momentum or $\int r^2 q' dA$, and the choice of the initial angular phase of the non-axisymmetric rotating potential vortex. These constraints can be written as (47)–(49) and (for an m-fold symmetric vortex with $m \neq 0$)

$$F_{N+4} = \oint [R^2(\tau)] \cos[m\theta(\tau)] ds - D = 0 \tag{60}$$

and

$$F_{N+5} = \oint [R^m(\tau)] \sin[m\theta(\tau)] d\tau \equiv 0, \tag{61}$$

where $R^2(\tau) = x^2(\tau) + y^2(\tau)$ and $ds = [(dx/d\tau)^2 + (dy/d\tau)^2]^{1/2} d\tau$. Equations (60) and (61) are not the most transparent form of expressing the two new constraints but are computationally the easiest to work with. The angular phase is specified by the choice of D.

The velocity must now have five unknown parameters:

$$\begin{aligned} \mathbf{v}(\mathbf{x}) = \bar{\mathbf{v}}(\mathbf{x}) + \mathbf{v}'(\mathbf{x}) + [V_{N+1} + V_{N+3}x - V_{N+4}y \\ + V_{N+5}R^{m-1}\cos(m\theta)]\hat{\mathbf{x}} + [V_{N+2} + V_{N+3}y \\ + V_{N+4}x - V_{N+5}R^{m-1}\sin(m\theta)]\hat{\mathbf{y}}, \end{aligned} \tag{62}$$

where V_{N+1}, V_{N+2}, and V_{N+3} have the same effect here as they do in Eq. (50), where V_{N+4} is a uniform rotation that changes the angular phase of the vortex and where V_{N+5} is an m-fold symmetric strain that changes its angular momentum. The Newton-solver finds the values of V_{N+4} and V_{N+5} required to make the vortex steady. (For V_{N+5}, this value is generally zero.)

REFERENCES

1. G. S. Deem and N. J. Zabusky, *Phys. Rev. Lett.* **40,** 859 (1978).
2. D. G. Dritschel, *J. Fluid Mech.* **157,** 95 (1985).
3. L. M. Polvani, N. J. Zabusky, and G. R. Flierl, *Fluid Dyn. Res.* **3,** 422 (1988).
4. L. J. Pratt and M. E. Stern, *J. Phys. Oceanogr.* **16,** 1101 (1986).
5. D. I. Pullin and P. A. Jacobs, *Stud. Appl. Math.* **75,** 77 (1986).
6. D. I. Pullin, *Annu. Rev. Fluid Dyn.* **24,** 89 (1992).
7. N. J. Zabusky, M. H. Hughes, and K. V. Roberts, *J. Comput. Phys.* **30,** 96 (1979).
8. Q. Zou, E. A. Overman, H. M. Wu, and N. J. Zabusky, *J. Comput. Phys.* **78,** 350 (1988).
9. D. G. Dritschel, *J. Comput. Phys.* **77,** 240 (1988).
10. G. R. Baker and M. J. Shelley, *J. Comput. Phys.* **64,** 112 (1986).
11. G. R. Baker and M. J. Shelley, *J. Fluid Mech.* **215,** 161 (1990).
12. L. M. Polvani, Ph.D. thesis, Massachusetts Institute of Technology, 1988 (unpublished).
13. H. M. Wu, E. A. Overman, and N. J. Zabusky, *J. Comput. Phys.* **53,** 42 (1984).
14. D. I. Pullin, P. A. Jacobs, R. H. J. Grimshaw, and P. G. Saffman, *J. Fluid Mech.* **209,** 359 (1989).
15. J. Pedlosky, *Geophysical Fluid Dynamics* (Springer-Verlag, New York, 1979).
16. H. J. Stuart, *Quart. Appl. Math.* **1,** 262 (1943).
17. P. S. Marcus, in *Numerical Analysis*, edited by D. F. Griffiths and G. A. Watson (Longman Sci. Tech., Essex, UK, 1986), p. 125.
18. P. S. Marcus, *Annu. Rev. Astron. Astrophys.* **431,** 523 (1993).
19. R. D. Van Buskirk, Ph.D. thesis, Harvard University, 1991 (unpublished).
20. R. D. Van Buskirk and P. S. Marcus, *J. Fluid Mech.*, submitted.
21. G. R. Baker, D. I. Meiron, and S. A. Orszag, *J. Fluid Mech.* **123,** 477 (1982).
22. S. Kida, *J. Phys. Soc. Japan* **50,** 3517 (1981).
23. A. E. H. Love, *Proc. London Math. Soc.* **25,** 18 (1893).
24. R. T. Pierrehumbert, *J. Fluid Mech.* **99,** 129 (1980).
25. H. B. Keller, in *Applications of Bifurcation Theory*, edited by P. Rabinowitz (Academic Press, New York, 1977), p. 359.
26. D. W. Moore and P. G. Saffman, in *Aircraft Wake Turbulence and Its Detection* (Plenum, New York, 1971), p. 339.