Contents

1. Historical survey ........................................... 80
2. Contour dynamics ........................................... 81
3. Contour surgery (overview) .................................. 84
4. Contour surgery (the algorithm) ................................ 87
   4.1. The equations of contour dynamics ......................... 87
   4.2. Contour representation ...................................... 89
   4.3. Interpolation .................................................. 90
   4.4. Node redistribution ........................................... 92
   4.5. Velocity determination ...................................... 92
   4.6. Time integration ............................................. 94
   4.7. Surgery ......................................................... 94
   4.8. Spherical contour dynamics/surgery ........................ 95
   4.9. Cylindrical or singly periodic contour dynamics/surgery 99
   4.10. Three-dimensional quasi-geostrophic contour dynamics/surgery 101
5. Accuracy ...................................................... 103
   5.1. Instantaneous and short-time accuracy ..................... 103
   5.2. Accuracy of extended integrations ......................... 107
   5.3. Comparison with a spectral calculation .................... 112
6. High-resolution calculations using contour surgery ............ 115
7. Discussion and outlook ........................................ 136
Appendix. The global invariants of two-dimensional flows .......... 141
References ....................................................... 144
CONTOUR DYNAMICS AND CONTOUR SURGERY: NUMERICAL ALGORITHMS FOR EXTENDED, HIGH-RESOLUTION MODELLING OF VORTEX DYNAMICS IN TWO-DIMENSIONAL, INVISCID, INCOMPRESSIBLE FLOWS

David G. DRITSCHEL
Department of Applied Mathematics and Theoretical Physics, University of Cambridge, Silver Street, Cambridge CB3 9EW, UK

Received 1 May 1988

The complex flow situations that regularly develop in a two-dimensional vortical flow have traditionally, indeed almost exclusively, been studied using Eulerian numerical methods, particularly spectral methods. These Eulerian methods have done remarkably well at modelling low to moderate Reynolds number flows. However, at the very high Reynolds numbers typical of geophysical flows, Eulerian methods run into difficulties, not the least of which is insufficient spatial resolution. On the other hand, Lagrangian methods, such as point vortex and contour dynamics methods, are inherently inviscid. It would appear, therefore, that Lagrangian methods are ideally suited for the modelling of flows at very high Reynolds numbers. Yet in practice, Lagrangian methods have themselves been limited by the frequent, extraordinary increase in the spatial complexity of inviscid flows. As a consequence, Lagrangian methods have been restricted to relatively simple flows which remain simple.

Recently, an extension of contour dynamics, “contour surgery”, has enabled the modelling of complex inviscid flows in wholly Lagrangian terms. This extension overcomes the buildup of small-scale structure by truncating, in physical space, the modelled range of scales. The result of this truncation, or “surgery”, is to make feasible the computation of flows having a range of scales spanning four to five orders of magnitude, or one to two orders of magnitude greater than ever considered by Eulerian or hybrid Eulerian–Lagrangian methods.

This paper discusses the history of contour dynamics which led to contour surgery, gives details of the contour surgery algorithm for planar, cylindrical, spherical, and quasi-geostrophic flow, presents new results obtained with high-resolution calculations, including the first every comparison between contour surgery and a traditional pseudo-spectral method, and outlines some outstanding problems facing contour dynamics/surgery.
1. Historical survey

The deceptively simple, idealized system of an inviscid, incompressible fluid restricted to a two-dimensional surface has attracted the interest of many scientists, including Euler, Helmholtz, Hill, von Kármán, Kirchhoff, Love, Poincaré, Rankine, Rayleigh, Stokes, J.J. Thomson, and W. Thomson (Kelvin) not to mention the far greater number of scientists from our own century. Its attraction might be thought to be largely a result of its simplicity when compared with three-dimensional flow; however, two-dimensional inviscid flow is neither simple nor unrealistic. Recent calculations [1–13], and others illustrated herein, exhibit flow developments of unexpectedly great complexity, while the strong analogy between two-dimensional inviscid flow and the fluid dynamics of the Earth's stratosphere (e.g., ref. [9] and references therein) has shown that two-dimensional flow is highly relevant to the understanding of many of the fundamental problems in geophysical fluid dynamics.

This paper focuses on one particular Lagrangian computational method, contour dynamics, whose recent extension [3,4] has elevated this method from computing the evolution of a highly restricted class of fluid motions while the fluid motions remain simple, to computing the evolution of fluid motions of unparalleled complexity. This extended form of contour dynamics, "contour surgery" [3], is able not only to compete with more traditional Eulerian methods, such as pseudo-spectral methods [14], but also to follow a wide variety of flow developments with far greater detail.

Lagrangian methods, the most common involving "point" vortices (see, e.g., ref. [15] and references therein), follow the motion of individual fluid elements as they wander through the fluid. The fluid elements individually conserve their vorticity, and the positions of all of the fluid elements at any given time determines the velocity field which moves them about (one can also add to the velocity field any irrotational velocity field plus an arbitrary rigid motion). By contrast, Eulerian methods calculate the changes in various quantities (vorticity, streamfunction) as the fluid sweeps past fixed locations within a fixed and finite domain (in the pseudo-spectral approach, these quantities are represented as a finite series of spectral harmonics, but for reasons of efficiency, these quantities and nonlinear products involving them are regularly transferred to and from a fixed grid).

It can be debated which of the two types of methods is better. In certain cases, one type might be much better suited than the other. Lagrangian methods have the effect of concentrating resolution in the vortex structures, the parts of the fluid which determine the entire flow evolution, whereas Eulerian methods cannot adjust resolution, and, at very high Reynolds numbers, this may lead to the bulk of the resolution being spent on the quasi-passive advection of weak vorticity [6–8]. This difficulty may well be the most troublesome for Eulerian methods, particularly in light of the well-known emergence of coherent structures (vortices) from an initially random vorticity field [14,16,17], the vorticity-gradient intensification occurring to vortices through the weak [7] and strong [11–13] interactions with other vortices, and the recognized quasi-passive role of the strips or filaments of vorticity that are regularly produced in both strong and weak vortex interactions [5–8].

Among Lagrangian methods, point vortex methods were the first to be used to study the dynamical evolution of two-dimensional flows. Modern calculations have employed thousands of vortices to simulate the breakup of a free shear layer into a row of vortices (coherent clouds of
point vortices) and follow the subsequent “pairing” or “merging” of adjacent vortices (see ref. [15] and references therein). Such calculations were pioneered by Roberts and Christiansen [18,19] who developed a hybrid Lagrangian/Eulerian numerical method, called the “cloud-in-cell” method, which has made point vortex calculations significantly more efficient than they had been in the pure Lagrangian framework. However, the approximations employed introduced a significant degree of numerical dispersion. By using a larger and larger number of vortices, this dispersion can be made as small as one wishes, theoretically, and, in the limit of an infinite number of “de-singularized” point vortices, it is possible to recover the exact equations of motion (see, e.g., ref. [20]). Nevertheless, point vortex calculations with a dynamic range nearly as large as that employed by present day spectral or contour dynamics (contour surgery) calculations have yet to be performed.

Point vortex methods probed deep into the unknown waters of vortex dynamics at a time when alternative methods were either not yet discovered or relatively inefficient, and some of the most fundamental nonlinear phenomena in vortex dynamics were seen more clearly than ever before in numerical computations [18,19,21]: vortex formation, merging, and tearing. One of these original studies [21] was contributed in part by Zabusky, who, six years later in 1979 with Hughes and Roberts [22] published the ingenious method of contour dynamics, a method related to the “waterbag” model developed for the Vlaslov equation by Berk and Roberts in 1970 [23]. In this original article on contour dynamics, computations indicated that two identical, initially circular vortices would merge (form one aggregate vortex with encircling filamentary debris) if their initial separation distance was less than about 3.4 vortex radii. A subsequent paper (appearing however a year earlier) by Deem and Zabusky [24] used contour dynamics to observe several additional fundamental phenomena, namely the near-recurrence of certain unsteady vortex motions, and “breaking waves” or “filamentation” [5]. They also greatly extended the class of steadily translating or rotating vortex equilibria, or “V-states”, using what may be termed “contour statics”, beyond the circular and elliptical equilibria discovered analytically in the nineteenth century, and the family of axisymmetric vortex rings, a member of which is Hill’s spherical vortex, computed by Norbury [25] in 1973. For further historical remarks, see the reviews by Aref [15] and Melander, Overman, and Zabusky [26].

2. Contour dynamics

The term “contour dynamics” means the dynamics of boundaries of vorticity discontinuity (contours); implicit is the assumption that the vorticity distribution is piecewise constant. This assumption, while apparently radical, does not in any way violate the equations of motion (the Euler equations), and the result is a closed dynamical system for the evolution of the fluid particles which lie on the boundaries of vorticity discontinuity – the instantaneous positions of the contours alone determine the velocity at any point in the fluid. Contour dynamics is possible because of the conservation of vorticity on fluid particles or elements and because the fluid is incompressible. So, the same set of fluid particles remain on a contour for all time.

This assumption of piecewise-constant vorticity may appear to severely limit the use of contour dynamics because “real” flows often have distributed or nonuniform vorticity. However, in principle, calculations using contour dynamics are no less able to compute the evolution of
flows with distributed vorticity than are calculations using, for example, pseudo-spectral techniques. Contour dynamics simply offers an alternative and complementary approach to the spatial discretization of two-dimensional flows – rather than divide a finite region of space into a two-dimensional array of grid points, contour dynamics divides space into a finite number of dynamic regions and discretizes the contours separating distinct regions of vorticity. In contour dynamics, the discretization of the vorticity proves to be exact, only the discretization along the contours introduces errors.

Contour dynamics has advanced rapidly along several fronts. New vortex equilibria – V-states have been obtained using the static version of contour dynamics. The numerical procedure entails finding the bounding shapes of one or more regions of uniform vorticity and a rotating and/or translating frame of reference in which the bounding shapes remain stationary. In general, the bounding shapes cannot be written down analytically. Early efforts led to the discovery of the families of m-fold symmetric, singly connected vortices (m = 2 corresponding to the Kirchhoff elliptical vortex) [24,27,28], the translating vortex pair [27–29], the pair of corotating vortices [30,27,31], the periodic row of vortices [32,33], finite-amplitude periodic waves [34,35], the asymmetric translating and rotating vortex pair [27], the double row of vortices [36], and corotating vortices up to eight in number [31]. More recently, families of noncircular vortices with multiple levels of vorticity have been discovered, a possibility foreseen by Zabusky [27]. One such family consists of vortices whose contours are described simply and analytically in terms of elliptical coordinates [37]. A wide range of vortices is possible depending, for instance, on the number of levels of vorticity and the value of the vorticity within each level. However, there are no vortices for which the vorticity magnitude decreases from the innermost level to the outermost one. This latter structure appears to be an ubiquitous feature of vorticity profiles in conventional simulations of two-dimensional turbulence [1,2,10,14,16,17], and in section 6 vortex equilibria with this structure are presented for the first time. Not unexpectedly perhaps, it appears that all disjoint, uniform-vorticity equilibria extend straightforwardly to nested, multi-contour equilibria, essentially because one can almost always replace a single contour by any number of contours that are sufficiently tightly nested.

The stability of some of these equilibria has been examined from two perspectives, linear and nonlinear. In the case of linear stability, numerically intensive treatments have been required for all but simply shaped equilibria. The analytical treatment of the stability of the elliptical vortex by Love [38] in the previous century took great advantage of the ability to write down the natural eigenmodes explicitly; the numerical treatments have not had this advantage. Owing to the complexity of the numerical treatments, only three families have been considered in detail: the corotating vortices [31], the double row or von Kármán vortex street [36–40], and finite-amplitude periodic waves [34]. For the corotating vortices (centered, in equilibrium, on the vertices of a regular polygon), six or fewer vortices are stable unless their boundaries are too close to one another. Seven or more vortices are always unstable, except in the case of seven point vortices [41,31] (seven finite-area vortices are unstable). For the von Kármán vortex street, the single stable point vortex configuration [42] was shown to pass continuously into a single stable finite-area configuration, rather than a range of stable configurations or none at all (see also ref. [43]). Finally, for the finite-amplitude periodic waves on the interface between two semi-infinite bodies of uniform-vorticity fluid, the small- to moderate-amplitude waves were found to be stable to subharmonic disturbances [34].
More often, direct nonlinear calculations have been used to assess stability. In this approach, one may follow instabilities through some degree of nonlinear development, at least until the computational cost or the numerical error becomes intolerable. Zabusky and colleagues have subjected ellipses [24,27,44], corotating pairs of vortices [44], and translating pairs [45] to a variety of perturbations, Kozlov and Makarov [46–49] have examined a wide range of equilibria, from ellipses and corotating pairs to 3- and 4-fold symmetric V-states [24,27,28] and unstable, multi-level circular vortices. Further calculations of disturbed ellipses, corotating pairs, and circular vortices are presented in ref. [50], with additional results for three or more corotating vortices [31]. The calculations in ref. [50] suggest that the margins of linear and nonlinear stability coincide for all of these equilibria, yet no general proof of nonlinear stability is known to exist (but see ref. [51]). Also in ref. [50], it is demonstrated that an ellipse of a certain aspect ratio (nearly 6:1) can evolve into a stable corotating pair, with arbitrarily little filamentary debris.

Many additional calculations have explored flow developments starting far from equilibrium. Overman and Zabusky [44,45], for example, have illustrated the interaction between two initially circular vortices of greatly differing sizes and strengths, and the collision between translating pairs of vortices.

Dynamical calculations have also been performed in slightly different geometries. Whereas contour dynamics was originally developed for an infinite domain, for which the Green function appearing in the contour integral formulation of the velocity field is simplest, it can be extended, by the method of images, to simple geometries, e.g., the half-plane [52] and singly periodic geometry [52–59,5,6,8]. Such modifications have enabled computations of the roll-up of a free shear layer [53–59,6,8], with “pairing” being calculated for the first time using nested contours by Jacobs and Pullin [55,60]. Pullin [52] also examined the process of “filamentation” in the context of a wall-bounded, periodic vortex layer (see also the early demonstration of filamentation on a disturbed circular vortex by Deem and Zabusky [24]; for more contemporary views, see ref. [5] and section 6). It is noteworthy that filamentation, the process by which thin filaments of vorticity form on contours, has plagued contour dynamics from the beginning, leading to the premature ending of many calculations. Pulling and Jacobs, in a series of papers, extended contour dynamics to include a uniform three-dimensional strain, in a first attempt to capture the effect of three-dimensional stretching on the evolution of an unstable shear layer [54–56,60]. They also examined vortex merging, or the suppression of it, in a three-dimensional strain field [53,56]. Even axisymmetric flows [61], e.g., vortex rings of uniform azimuthal vorticity, and flows on the surface of a sphere [4] have been studied using variants of contour dynamics. In the spherical case, it is a remarkable fact that the expression for the velocity field is the same as that for infinite planar flow, except that the two planar coordinates are replaced by three, Cartesian, coordinates restricted to the surface of the sphere (see section 4.8 for details).

On another front, refinements were being made to the basic contour dynamics algorithm. Perhaps the central numerical problem of contour dynamics concerns the placement or distribution of nodes, or computational points, along the contours of vorticity discontinuity. Many numerical algorithms connect the nodes with straight line segments [22,24,27,28,44–49,52,62], but, more recently, local and global spline interpolation has been implemented to achieve higher-order accuracy [50,53–58,60,63–65]. Whatever the interpolation, though, it has been recognized that efficient, accurate calculations require some means of adding and deleting nodes.
as a means to represent as many degrees of freedom as possible [44,45,54–58,60,65]. Most schemes for node distribution and redistribution have followed that developed by Overman and Zabusky [44,45]. In their scheme, nodes are placed so that the distance between adjacent nodes approximately scales with the local radius of curvature, with the constraints that the distance between adjacent nodes is neither too small nor too large both relative to the distance between successive node pairs and overall. Other schemes utilize local or global spline interpolation to achieve a smoother distribution of nodes [58,65] (see also sections 3 and 4).

From the positions of the nodes and the shape of the contour between nodes, the velocity field at each node is obtained by integration of a logarithmic kernel around the contours (see section 4). This integration is done differently by almost every researcher. When linear interpolation is used, the integral between each pair of nodes can be evaluated explicitly, but the answer as well as the speed of the algorithm depends on whether or not the kernel or its transformation into a nonsingular kernel via an integration by parts is used [62]. When nonlinear interpolation is used, numerical quadrature is necessary, and again there are different answers depending upon an integration by parts, whether or not nearby contributions are given special treatment, the type of numerical quadrature (e.g., Gaussian), and the form of interpolation (see refs. [50,54,57,58,60,63–65] for all the possibilities). The contour surgery algorithm [3,4] described in sections 3 and 4 does part of the integration explicitly and part by numerical quadrature.

Recently, Buttke [66] has developed a method whereby the velocity field is not found by contour integration. This method is purported to be significantly faster than traditional contour dynamics using linear interpolation (but see the remarks in section 5). In Buttke’s method, the interior of a region of uniform vorticity is divided into squares of varying size with the smallest squares straddling the vortex boundary, itself comprised of nodes connected by straight line segments. The velocity at each node is calculated by summing the contributions from each square, the only error coming from the boxes that straddle the boundary.

Contour dynamics is just one of many applications of self-deforming surfaces. Several closely related applications involve, generally, the combined effects of vorticity and velocity jumps, i.e. contours of discontinuous vorticity and velocity. These include Batchelor flows [67,68], surface and internal gravity waves [69–72], Hele–Shaw flows [73 and references therein], and plasma clouds [64]. In applications where small-scale structure contributes little to the global dynamics yet gets in the way of extended calculations, it is possible that an extension along the lines described for contour dynamics in the following sections may lead to a more versatile numerical algorithm.

3. Contour surgery (overview)

Despite its now extensive use, contour dynamics (hereafter CD) has never been able to compute (accurately) beyond the first stage in the nonlinear development of an unsteady flow. For example, in an earlier work [50], significant but principally large-scale deformations of the vortex shapes were computed, such as the beginning stages of the coalescence of three vortices. The calculations could not, however, cope with the rapid development of small-scale features, most notably the formation of regions of extreme and fast-growing curvature. The calculations
further suffered from an unmanageable growth in contour perimeters resulting from the continual deformation of the contours.

"Contour surgery" (hereafter CS) represents an extensive overhaul and refinement of CD that enables one to compute far beyond CD [3]. CS accomplishes this by truncating the dynamic range through the automatic removal of contour features smaller than a prescribed scale, say \( \delta \). For example, CS abandons resolving, with sufficient detail, parts of the contour whose radius of curvature becomes comparable with \( \delta \). CS also topologically reconnects contours or distinct parts of a contour when these contours or distinct parts thereof become closer than \( \delta \). These "surgical operations" greatly control the build-up of resolution associated with both the formation of regions of high curvature and the growth in total contour perimeter.

Surgery does introduce a new error, for contours cannot really break in an inviscid fluid, but the philosophy underpinning CS views surgical errors as simply another approximation of the (contour-dynamical) equations of motion, the other approximations being the use of finite resolution along contours and finite temporal resolution. In practice, calculations using CS make comparable errors in all three approximations. Direct tests of accuracy are given in section 5.

The surgery that gives CS its name is not simply a program module which may be attached to any CD algorithm. Rather, the basic CD algorithm had to be modified extensively to work together with surgery. The basic modifications to CD include:

1. nonlocal node (resolution) adjustment at every time step,
2. the interpolation of the contour between nodes by local cubic polynomials (nonlinear interpolation),
3. the explicit calculation of the velocity field (from contour integrals) to first order in the (small) departure of the curved contour from a straight line segment connecting adjacent nodes, and
4. automatic surgery.

The action of any one of these modifications within the CS algorithm depends on those of the others. The node adjustment scheme, described in more detail below, distributes a variable number of nodes in such a way that the interpolated curve between adjacent nodes remains very nearly straight, and this in turn enables the contour integral formulation of the velocity field to be expanded in a perturbation series in the small departure of the curve from a straight line segment between any pair of adjacent nodes (see fig. 1). The terms in the perturbation series can be evaluated explicitly, without resort to numerical quadrature, and in practice, only terms up to first order in this small departure are necessary to maintain a degree of accuracy consistent with all of the approximations. (The explicit evaluation of the velocity field hinges on the assumption

![Fig. 1. The nonlinear interpolation of a contour between successive nodes. The quantity \( e \) is the straight line distance along the cord connecting nodes \( i \) and \( i+1 \), \( p \) is the fractional distance between the nodes (0 ≤ \( p \) ≤ 1), and \( \eta \) is the distance, relative to \( e \), from the cord to the curved contour at \( x \). The interpolation scheme assumes \( \eta(p) = \alpha_i p + \beta_i p^2 + \gamma_i p^3 \) with the constants \( \alpha_i, \beta_i, \) and \( \gamma_i \) being determined wholly in terms of the locations of nodes \( i-1 \) through \( i+2 \).]
that the Green function is proportional to the logarithm of the distance between two points; in case where the Green function is more complicated (see sections 4.9 and 4.10), it is advantageous to split it into a logarithmic part and a remainder, to evaluate the logarithmic part explicitly, and to use numerical quadrature on the remainder.) The explicit evaluation of the velocity field overcomes the problems associated with numerical quadrature noted in ref. [50].

Nodes are redistributed at each time step in an attempt to represent the most degrees of freedom with the fewest nodes. Experience has shown that the local instantaneous properties of the contour (e.g., the curvature) are not enough to determine the optimal placement of nodes. Schemes designed on this basis blind contours to the approach of small-scale structures which, with perfect resolution, would tend to increase the curvature on the contours (see fig. 2). The result is that contours inevitably cross in violation of the equations of motion. It is necessary, instead, that the nodes be placed where the contour is going to bend; in other words, the node redistribution must be sensitive to the variation of the velocity field along the contour. By much trial and error, a scheme was adopted whereby the local density of nodes (the number of nodes per unit length of contour) is proportional to the two-thirds power of a weighted sum of the curvatures at all nodes with the weights chosen to mimic the variation of the velocity field. The choice two-thirds was found to minimize errors associated with the calculation of the velocity field [3] and is also consistent with the fact that the largest scales determine the greatest proportion of the dynamics.

The namesake of the algorithm came from its ability to perform automatic surgery. One aspect of surgery concerns the formation of corners. The node-adjustment scheme ensures that regions

Fig. 3. Surgery. If the distance between two contours or two distinct parts of a contour becomes less than the cutoff scale $\delta$, the contours or contour parts as the case may be are joined by shifting the node $i$ to the opposite contour where it is labelled $i'$. The new node $i'$ actually consists of two nodes, one belonging to the contour or part of the contour to the right and the other to the one on the left.
of extreme curvature turn into corners by allowing three successive nodes along a contour to bend into an acute angle when the curvature exceeds the inverse of the surgical scale $\delta$. A corner node is a specially labelled node which cannot be moved by the node-adjustment scheme and which is connected to its neighbors by straight line segments rather than cubic polynomials. A corner node may lose its special status if the angle at the corner becomes obtuse again. The second aspect of surgery concerns the operation that takes place when two contours or two distinct parts of the same contour become closer than $\delta$. The contours are automatically reconnected as shown in fig. 3. This aspect of surgery also removes the ends of thin filaments should they become excessively thin (see fig. 4).

4. Contour surgery (the algorithm)

4.1. The equations of contour dynamics

In this section, the specific modifications and extensions of the basic contour dynamics algorithm which give rise to contour surgery are described in detail, and in the following section, the algorithm's accuracy is assessed. These two sections essentially reproduce and combine parts of three earlier papers [3-5], although new results are given concerning accuracy, comparisons with traditional Eulerian methods, and extensions of the algorithm to cylindrical and quasi-geostrophic flows. Those readers who are not immediately concerned with the specific nature of the algorithm may wish to skip these sections and continue with section 6.

The origin of contour dynamics lies in the simple way in which the inversion of Laplace's equation, $A\psi = \omega$, symbolically written as

$$t+b=\?\frac{\partial}{\partial t}G(x, y; x', y')\omega(x', y') \, dx' \, dy',$$

$(G=(2\pi)^{-1} \log |x-x'|)$ simplifies when the vorticity distribution $\omega(x, y)$ is piecewise constant or uniform. (Material conservation of vorticity ensures that $\omega$ will remain piecewise constant and, moreover, that the boundaries or contours separating regions of distinct, uniform vorticity will be comprised of the same fluid particles for all time.) One obtains the particle velocities, $dx/dt = u(x) = (-\partial_y, \partial_x)\psi$, at any point $x$ in the fluid by simply applying the operator $(-\partial_y, \partial_x)$ to $G\omega$, exchanging $(\partial_y, -\partial_x)$ for $(-\partial_y, \partial_x)$ within the integrand, and finally using Green's theorem to convert the area integral over each region of uniform vorticity to line or contour integrals around the boundaries. Since two regions often have a common boundary, only
the jumps in vorticity $\tilde{\omega}_k$ crossing each contour $C_k$ inwards matter. The well-known result is

$$\frac{dx}{dt} = u(x) = -\sum_k \tilde{\omega}_k \int_{C_k} G(x; x_k) \, dx_k,$$

(2)

where $x_k$ is a point on $C_k$, and the the region interior to $C_k$ is to the left of $dx_k$. This equation applies to all points $x$ in the fluid, including those on the contours.

Most researchers have preferred to remove the logarithmic singularity at $x = x_k$ in the integrand of (2) to obtain a more stable numerical algorithm. By an integration by parts, the integrand is transformed into $(x - x_k)(\partial G/\partial x_k \cdot dx_k)$, which is continuous through $x = x_k$ but, in general, has infinite slope at $x = x_k$. For piecewise-linear contours $C_k$, this slope singularity is unimportant (see, e.g., ref. [62]), but it can introduce numerical instabilities when nonlinear interpolation is used and the integration is performed using numerical quadrature (see ref. [60]).

The contour-integral formulation (2) also applies to more general relationships between streamfunction and vorticity. Suppose $\mathcal{L}\psi = \omega$ where $\mathcal{L}$ is some linear operator. If, simply, the Green function $G$ of the inversion (see (1)) has the property $G(x; x') = F(x' - x)$, then (2) applies. For example, for the Helmholtz operator $\mathcal{L} = \Delta - \lambda^2$, $F(\xi) = -(2\pi)^{-1}K_0(\lambda |\xi|)$, where $K_0$ is a modified Bessel function.

In principle, boundaries or spatial periodicity may also be incorporated by using the method of images. Additionally, the two-dimensional surface need not be flat; however, not all curved surfaces have the requisite symmetry for the Green function. Some simple surfaces which do admit a contour-integral formulation like (2) are the cylinder [5] and the sphere [4], and for these surfaces, it turns out to be simpler to use three Cartesian coordinates (the constraint that the fluid particles remain on the surface being automatically satisfied; see sections 4.8 and 4.9).

For general, orientable two-dimensional surfaces which are sufficiently smooth, one can still formulate contour dynamics, only the expression for the velocity field is not as simple as that given by (2). Let $\xi$ and $\eta$ be orthogonal right-handed coordinates on such a surface, and let $g(\xi, \eta) \, d\xi$ and $h(\xi, \eta) \, d\eta$ be the incremental distance elements in both directions. Then an incremental area on this surface is $J \, d\xi \, d\eta$ where $J = gh$, and so the streamfunction is given by

$$\psi = \mathcal{G}\omega = \int \int G(\xi, \eta; \xi', \eta') \omega(\xi', \eta') J(\xi', \eta') \, d\xi' \, d\eta',$$

(3)

where $G$ is the Green function appropriate to the surface (in principle, $G$ exists). The component velocities in the $\xi$ and $\eta$ directions, $g \, d\xi/dt$ and $h \, d\eta/dt$, are given by derivatives of the streamfunction in the usual way,

$$g \frac{d\xi}{dt} = -\frac{\partial \psi}{h \, \partial \eta}, \quad h \frac{d\eta}{dt} = \frac{\partial \psi}{g \, \partial \xi}.$$

(4)

When $\omega(\xi', \eta')$ is piecewise constant, the area integrals implicit on the right-hand side of the above equations can be converted into contour integrals using Green’s theorem for a general (orientable) surface:

$$\int \int J \left( \frac{1}{g} \frac{\partial Q}{\partial \xi} - \frac{1}{h} \frac{\partial P}{\partial \eta} \right) \, d\xi' \, d\eta' = \int_{\partial \mathcal{S}} (P g \, d\xi' + Q h \, d\eta').$$

(5)
Here, \( \mathcal{R} \) denotes the region of integration and \( \mathcal{C} \) its bounding contour(s), and \( P, Q, g, h, \) and \( J \) are functions of \( \xi' \) and \( \eta' \). Identifying

\[
P(\xi, \eta; \xi', \eta') = \frac{1}{h(\xi, \eta)} \int \frac{\partial G}{\partial \eta} (\xi, \eta; \xi', \eta') h(\xi', \eta') \, d\eta',
\]

\[
Q(\xi, \eta; \xi', \eta') = \frac{1}{g(\xi, \eta)} \int \frac{\partial G}{\partial \xi} (\xi, \eta; \xi', \eta') g(\xi', \eta') \, d\xi',
\]

one arrives at the equations for contour dynamics:

\[
g(\xi, \eta) \frac{d\xi}{dt} = \bar{\omega} \int \mathcal{C} P(\xi, \eta; \xi', \eta') g(\xi', \eta') \, d\xi', \quad (7a)
\]

\[
h(\xi, \eta) \frac{d\eta}{dt} = \bar{\omega} \int \mathcal{C} Q(\xi, \eta; \xi', \eta') h(\xi', \eta') \, d\eta'. \quad (7b)
\]

where \( \bar{\omega} \) is the jump in vorticity across the contour \( \mathcal{C} \), and in general the total velocity represents a sum over such contours, as in (2).

It remains an open question whether or not it is possible to formulate contour dynamics on a Möbius strip or other non-orientable surfaces.

4.2. Contour representation

Each contour, in theory an infinite series of points, is approximated by a finite but adjustable number of “nodes” connected together by simple curved contour segments (see fig. 1 and below). The number of nodes may change in time in response to the changing complexity of the contour; where the curvature along the contour is increasing, the local density of nodes (the number of nodes per unit distance along the contour) generally increases, while, similarly, the density of nodes generally decreases when the curvature decreases locally. However, it often happens that a part of a contour with a relatively low curvature encounters a small-scale feature, such as a small vortex or the tip of a small vorticity filament, and if the nodes along the contour are spaced too far apart, the ensuing distortion of the contour by the small-scale feature will be nothing like that of the true, infinitely flexible contour (see fig. 2).

The expression for the local density of nodes \( \rho \) which emerged after many tests of alternative expressions is sensitive to both the local curvature \( \kappa \) and the presence of nearby, small-scale features embodied by the definition of the nonlocal curvature \( \bar{\kappa} \) introduced below. When nonlocal effects are weak, and when the radius of curvature greatly exceeds the surgical scale \( \delta \), \( \rho \) has the approximate form

\[
\rho = (\mu L)^{-1}(\kappa L)^{a}
\]

where \( \mu \) is a small non-dimensional number, \( L \) is a length typical of the large-scale vorticity distribution, \( a \) is a number between 0 and 1 which controls how steeply \( \rho \) rises with curvature \( \kappa \),

\[
(8)
\]
and, at each node \( i \), \( \kappa \) is computed by passing a circular arc through the nodes \( i-1 \), \( i \), and \( i+1 \).

If \( a = 1 \), then \( \rho = \kappa / \mu \), and this implies that the same number of nodes would be used to resolve a given contour shape of any size. Larger vortices, however, have a greater influence on the total velocity field than do smaller vortices, so values of \( a \) less than one are more consistent with the relative importance of different-sized vortices (or features). Indeed, the currently used value \( a = 2/3 \) minimizes certain measures of error for the algorithm (see section 5).

To incorporate nonlocal effects, the local curvature \( \kappa_i \) in (8) is replaced by a weighted sum of the curvature values at all nodes,

\[
\bar{\kappa}_i = \sum_j e_j |\omega_j| \kappa_j \left( \sum_j \frac{e_j |\omega_j|}{d_{ij}^2} \right)^{-1},
\]

where \( e_j = |x_{j+1} - x_j| \) is the distance between nodes \( j \) and \( j+1 \), \( \omega_j \) is the jump in vorticity across the contour segment joining \( j \) and \( j+1 \), \( \bar{\kappa}_j = (\kappa_j + \kappa_{j+1})/2 \), \( d_{ij} = |x_i - \frac{1}{2}(x_j + x_{j+1})| \), and the summation is over all nodes on all contours. Because of the \( d_{ij}^{-2} \) factor, distant contours or parts of a contour contribute little to \( \bar{\kappa} \) unlike these have large jumps in vorticity \( \omega_j \).

Finally, the scales near the surgical scale are deliberately under-resolved to make way for one aspect of surgery. The poor resolution causes nodes in sufficiently high curvature regions to turn through a right angle (two line segments connecting three successive nodes form an acute angle). When this happens, the node at the vertex is labelled a “corner”, and the curvature \( \kappa \) at all three nodes is set to zero to reduce resolution near the corner. With this aspect of surgery taken into account, the local density of nodes, in fact the average density between two nodes, say \( i \) and \( i + 1 \), assumes the following form

\[
\rho_i = \frac{\hat{\kappa}_i}{1 + \delta \hat{\kappa}_i / \sqrt{2}},
\]

(10a)

\[
\hat{\kappa}_i = \frac{1}{3}(\bar{\kappa}_i + \bar{\kappa}_{i+1}),
\]

(10b)

\[
\bar{\kappa}_i = (\mu L)^{-1}(\bar{\kappa}_i L)^a + \sqrt{2} \bar{\kappa}_i,
\]

(10c)

with \( \bar{\kappa}_i \) given in (9). This form limits the minimum distance between two nodes to nearly \( \delta \) and the maximum computed curvature to nearly \( \delta^{-1} \).

The specific way in which nodes are distributed so as to be consistent with the computed density of nodes depends upon the interpolation between nodes, discussed next.

4.3. Interpolation

Between adjacent nodes \( i \) and \( i + 1 \), the contour takes the form of a cubic spline,

\[
x(p) = x_i + pt_i + \eta(p)n_i,
\]

(11a)

\[
t_i = (a_i, b_i) = x_{i+1} - x_i,
\]

(11b)

\[
n_i = (-b_i, a_i),
\]

(11c)

\[
\eta(p) = \alpha p + \beta p^2 + \gamma_i p^3,
\]

(11d)
for $0 \leq p \leq 1$ (see fig. 1). The cubic interpolation coefficients $\alpha_i$, $\beta_i$, and $\gamma_i$ are determined by the conditions $\eta(1) = 0$, $\kappa(0) = \kappa_i$ and $\kappa(1) = \kappa_{i+1}$, where

$$\kappa_i = \frac{2}{\left| e_{i-1}^2 t_i + e_i^2 t_{i-1} \right|} a_{i-1} b_i - b_{i-1} a_i$$

is the curvature found by passing a circle through the nodes $i - 1$, $i$, and $i + 1$, and

$$\kappa(p) = \frac{d^2 \eta/dp^2}{e_i \left(1 + (d \eta/dp)^2\right)^{3/2}}$$

(subscripts on the functions $\eta(p)$ and $\kappa(p)$ are omitted to simplify notation). The density of nodes is such that the departure of the contour from a straight line segment is always small (except near incipient corners), and this allows one to neglect $(d \eta/dp)^2$ in the expression for $\kappa(p)$ above. Momentarily putting aside questions of error, this approximation leads to simple expressions for the cubic interpolation coefficients:

$$\alpha_i = -\frac{1}{3} e_i \kappa_i - \frac{1}{6} e_i \kappa_{i+1},$$

$$\beta_i = \frac{1}{2} e_i \kappa_i,$$

$$\gamma_i = \frac{1}{6} e_i (\kappa_{i+1} - \kappa_i).$$

Note that because $\kappa_i$ uses information at nodes $i - 1$ through $i + 1$, the above coefficients depend on information at four successive nodes, $i - 1$ through $i + 2$. If the approximations made above are justified, the order of accuracy of the interpolation should be cubic, two greater than linear interpolation (which uses information at only two nodes). The results in section 4.4 prove that this is indeed the case – there, it is shown that the interpolation error for an elliptical contour, the error measured as the positive area difference between the interpolated and exact contour shape, falls off like $n^{-4}$, where $n$ is the total number of nodes used, as opposed to $n^{-2}$ for linear interpolation.

This method of interpolation leaves a slight discontinuity in the tangent slope at each node, but the discontinuity does not appear to produce adverse effects. It is simple enough to develop a method for continuous tangent slopes, but the interpolation errors of such a method (one that forces the contour, at each node $i$, to be tangent to the vector $t_{i-1}/e_{i-1}^2 + t_i/e_i^2$), while of the same order of accuracy, are greater than those of the curvature-continuity method above. Indeed, it is not surprising that one combination of information at four successive nodes leads to greater accuracy than another.

Near corners, the points along a contour where the curvature has exceeded approximately $\delta^{-1}$, the interpolation is poor. It is poor because the smallest scales of the calculation are assumed unimportant – all hope has been given up trying to model the proper dynamics at these scales. Because the curvature at the corner node and its two adjacent neighbors is set to zero, the interpolation along these two segments is simply linear.
4.4. Node redistribution

After moving the nodes on all of the contours through one time step by the method described in the next subsection, one computes the quantity \( \sigma_i = \rho_i e_i \) for each node \( i \). Recall that \( \rho_i \) is the desired average density of nodes between \( i \) and \( i+1 \) while \( e_i \) is the distance between nodes \( i \) and \( i+1 \). Therefore, \( \sigma_i \) can be thought of as the fractional number of nodes to be placed between \( i \) and \( i+1 \). If \( \sigma_i > 1 \), more resolution is demanded, while if \( \sigma_i < 1 \), less is.

The node redistribution process begins by fixing one point, say node 1, along a contour. If the contour has corners, one begins at a corner and distributes nodes without moving the corners. Let \( n \) be the node preceding the next corner or the last node on the contour (the node preceding 1). Compute

\[
q = \sum_{i=1}^{n} \sigma_i
\]

and define \( \tilde{n} = [q] + 2 \), the nearest integer to \( q \) plus two. The \( n-2 \) "old" nodes between the fixed nodes will be replaced by \( \tilde{n} - 1 \) entirely new nodes in such a way that the spacing of new nodes is approximately consistent with the desired average density. In addition, the new nodes will lie along the curved contour segments connecting the old nodes. Let \( \sigma'_i = \sigma_i \tilde{n}/q \) so that \( \sum_{i=1}^{n} \sigma'_i = \tilde{n} \). Then, the positions of the new nodes \( j = 2, \ldots, \tilde{n} \) are found successively by seeking \( i \) and \( p \) such that

\[
\sum_{i=1}^{i-1} \sigma'_i + \sigma'_i p = j - 1
\]

and placing each new node \( j \) between the old nodes \( i \) and \( i+1 \) at the position \( x(p) \) given in (11) (see fig. 5).

With regard to accuracy, the placement of the new nodes along the curved segments between the old nodes means that the order of accuracy of node redistribution is the same as that of interpolation. For example, the error in area of approximating a circle by \( n \) nodes is \( O(n^{-4}) \), and the additional error caused by subtracting one node and redistributing the rest is also \( O(n^{-4}) \), and of comparable magnitude.

4.5. Velocity determination

The weak departure of the interpolated contour from a straight line segment between adjacent nodes suggests progress can be made by expanding the velocity field contour integrals (see (2)) in

![Fig. 5. Node redistribution. The new nodes, marked by "○", are distributed along the curved contour corresponding to the old nodes, marked by "●", between the fixed nodes \( i=1 \) and \( n \). Once the new nodes are placed, the old ones, except for the fixed nodes, are removed.](image-url)
a series in $\eta$ (see (11)). First, write the velocity $u$ at a point $x$ as a sum of contributions from the contour segments adjoining all the adjacent nodes:

$$u = \frac{1}{2\pi} \sum_i \omega_i \Delta u_i.$$  

(16)

Here, $\omega_i$ is the vorticity jump across the contour to which $i$ belongs, as before, and

$$\Delta u_i = -\int_0^1 dp \log |x_i - x + pt_i + \eta n_i| \left(t_i + \frac{d\eta}{dp} n_i\right).$$  

(17)

Expanding $\Delta u_i$ in a series in $\eta$,

$$\Delta u_i = \Delta u_{i0} + \Delta u_{i1} + \ldots,$$

the first couple of terms are

$$\Delta u_{i0} = -t_i \int_0^1 dp \log |x_i - x + pt_i|,$$

(18a)

$$\Delta u_{i1} = c_i t_i \int_0^1 \frac{\eta dp}{|x_i - x + pt_i|^2} - n_i \int_0^1 dp \frac{d\eta}{dp} \log |x_i - x + pt_i|,$$

(18b)

where $c_i = n_i \cdot (x - x_i)/e_i^2$. The first term in this series is just the contribution from integrating along straight line segments between nodes, and this can be evaluated analytically when the Green function is logarithmic, the case considered here (eq. (18a) was used in the first contour dynamics calculations [22]). All of the remaining terms of the series can also be evaluated analytically, in principle, but it is sufficient to evaluate $\Delta u_{i1}$, because higher-order terms contribute errors of similar magnitude to those arising from the interpolation and node redistribution. The explicit formula for $\Delta u_i$ to first order in $\eta$ and without essential subscripts is

$$\Delta u = T t + N n,$$

$$T = 1 - dr - (1 - d)r' - c^2 h + c\left[\beta + \left(\frac{1}{2} + 2d\right)\gamma + f(s - r) + gh\right],$$

$$N = a + \left(\frac{1}{2} + d\right)\beta + \left(\frac{1}{2} + \frac{1}{2}d + d^2 - c^2\right)\gamma + g(s - r) - c^2 hf,$$

$$c = n \cdot (x - x_i)/e^2, \quad d = t \cdot (x - x_i)/e^2, \quad e = |t|,$$

$$f = \frac{d\eta}{dp} (d) - c^2 \gamma, \quad g = \eta (d) - c^2 (\beta + 3d\gamma), \quad h = \frac{1}{c} \left[\tan^{-1}\left(\frac{1 - d}{c}\right) + \tan^{-1}\left(\frac{d}{c}\right)\right],$$

$$r = \log |x - x_i|, \quad r' = \log |x - x_{i+1}|.$$  

(19)

Minor changes to these formulae must be made when $x$ is close to a contour segment connecting, say, $i$ and $i + 1$. In this case, the term $\eta n_i$ in the logarithm in (17) cannot be
considered small compared with $x_i - x + pt_i$. Since $\eta$ is small, it is sufficient to make a correction only if $x$ lies within the circle whose diameter connects the points $x_i$ and $x_{i+1}$, and this condition is met if $d > c^2 + d^2$. Supposing then that this condition is met and noting that $x = x_i + dt_i + cn_i$, $0 < d < 1$, consider splitting $x_i - x + pt_i + \eta(p)n_i$ into the parts $x_i - x + pt_i + \eta(d)n_i$ and $[\eta(p) - \eta(d)]n_i$. Now, the second part is always much smaller than the first (especially since $\eta$ is small) no matter how close $x$ is to the contour between nodes $i$ and $i + 1$. With this substitution in (18), the integrations can again be done explicitly, being scarcely different than before. In (19), $c$ is decremented by $\eta(d)$ with the consequent implicit changes to $f$, $g$, $h$, $T$, and $N$, and $g$ is decremented by $\eta(d)$ but only where it appears in $T$ (the $g$ appearing in $N$ is unchanged).

4.6. Time integration

The time integration is performed using the fourth-order Runge-Kutta scheme [74, p. 896, eq. (25.5.10)] in order to achieve high-order temporal accuracy. It is necessary to use a self-starting scheme because all of the nodes are redistributed before each time step. The choice of the time step, $\Delta t$, depends crucially on the maximum possible velocity derivatives at any node. Basically, if the velocity derivatives multiplied by the time step are small, then one does achieve the high-order accuracy for which the scheme is designed. The actual velocities do not matter because the scheme can integrate uniform velocities perfectly. A fair estimate for the maximum possible velocity derivatives is the magnitude of the peak vorticity in the fluid, $\omega_m$, because there is no faster time scale in the fluid. Also, since vorticity is conserved, it is sufficient to use a fixed time step inversely proportional to $\omega_m$. In practice,

$$\Delta t \leq 0.1\pi/\omega_m$$

is chosen, although larger values of $\Delta t$ are sometimes used in the later stages of particularly complex calculations (see section 6). By experience, the above choice for $\Delta t$ has been found to virtually eliminate the possibility of two contours crossing one another; larger values greatly increase the probability of this error. This error is not fatal, it only detracts from the credibility of the calculation.

4.7. Surgery

Surgery basically consists of two operations, the generation and removal of corners and the reconnection of close contours or parts of a contour. As described above, corners are produced when the curvature exceeds approximately $\delta^{-1}$ as a consequence of deliberately poor resolution. Corners may also be removed. If the angle made by the line segments adjacent to the corner becomes obtuse, the corner is returned to the status of a simple node.

The second surgical operation is illustrated in fig. 3. This operation is permitted only between contours enclosing the same value of vorticity. In this way, a single contour may break into two pieces or two contours may merge. In addition, this operation removes the ends of excessively thin filaments (see fig. 4). Surgery is not done between contours that enclose different values of
vorticity, however useful this might be, simply because the logical manipulations that would be necessary appear very, very complicated.

Note that the decision to perform surgery rests on the distance between a node and the straight line segment connecting a pair of adjacent nodes; for reasons of simplicity, the curvature of the contour between node pairs is neglected. The use of straight line segments, however, limits the minimum size of $\delta$ essentially because $\delta$ should be at least as great as the normal departure of the contour $e\eta$ between two nodes. To obtain an estimate for the minimum size of $\delta$, consider an example in which $e\eta$ is simple to calculate, a circular contour of radius $R$. Assuming, a priori that $e\eta$ is largest when $R \gg \delta$, one obtains from (8)

$$e = \rho^{-1} = \mu L(R/L)^a.$$ 

Further, the maximum of $\eta$ (over $p$) for a circular arc connecting two nodes is approximately $\frac{1}{8}e/R$, so that the maximum of $e\eta$ is

$$\frac{1}{8R} \mu^2 L^2 (R/L)^{2a}.$$ 

This expression is largest when the circle has a radius comparable to the prescribed large-scale length $L$ (for $a > \frac{1}{2}$), so the requirement that the surgical scale be larger than $e\eta$ is satisfied when

$$\delta/L > \frac{1}{8} \mu^2.$$ 

(20)

In practice, (20) appears to be an overestimate, but most calculations are performed with $\delta \approx \frac{1}{8} \mu^2 L$ because smaller $\delta$ only apparently increases the resolution of the calculation. The resolution at scales of $O(\frac{1}{8} \mu^2 L)$ is already poor. For example, the approximate number of nodes which would be used to resolve a circular contour of radius $L$ is $2\pi/\mu$, while the corresponding number for a circular contour of radius $\frac{1}{8} \mu^2 L$ is only $\pi/\mu^{1/3}$ (here $a = 2/3$, and the simple expression (8) for the node density has been used). For a typical parameter value $\mu = 0.03$, 209 nodes would be used to resolve the $R = L$ contour while only 10 would be used to resolve the $R = \frac{1}{8} \mu^2 L$ contour.

Table 1 summarizes the contour surgery algorithm.

4.8. Spherical contour dynamics/surgery

Remarkably simple equations govern the motion of an inviscid, incompressible fluid on the surface of a sphere; in fact, the contour dynamical equations on a sphere and on an infinite plane are identical, except that the position vectors in the spherical case are three-dimensional ones stemming from the center of the sphere [4]. This subsection includes a derivation of this simple result and a summary of the corresponding contour surgery algorithm.

One starts by finding the Green function $G$ of the inversion of Laplace's equation $\Delta \psi = \omega$ on a unit sphere. With $\theta$ defined as co-latitude and $\phi$ as longitude, it is a simple matter to show

$$G(\theta, \phi; \theta', \phi') = \frac{1}{4\pi} \log(1 - \cos \Theta),$$

(21)
Table 1
Flow chart of the contour surgery algorithm

1. Initialization
   a. Read in data specifying the initial conditions and algorithm parameters.
   b. Calculate the cubic interpolation coefficients (eqs. (6)–(9)).
   c. Redistribute nodes (eq. (10)).
   d. Save data for post-processing.
2. Advection
   a. Re-calculate the cubic interpolation coefficients.
   b. Calculate the velocity field (eq. (14)).
   c. Repeat steps (a) and (b) three more times to complete the Runge–Kutta integration.
3. Surgery
   a. Search contours for new corners.
   b. Search for contour merger situations (fig. 3). If a single contour satisfies the contour merger condition, break it into two contours, and introduce two new corners, one for each new contour, at the break (at node \(i'\) in fig. 3).
      If two contours enclosing the same interior vorticity satisfy the contour merger condition, join them together, and introduce two new corners (at the double node \(i'\) in fig. 3). And, if the surgery is done near the end of a filament (see fig. 4), remove the end of the filament if it consists of 4 or fewer nodes after surgery.
   c. Repeat step (b) until all merger possibilities have been exhausted.
4. Post-surgery
   a. Re-calculate the cubic interpolation coefficients.
   b. Redistribute nodes.
   c. Periodically save data for post-processing.
   d. Return to item (2) unless the desired number of time steps have been taken.

where \(\cos \Theta = \cos \theta \cos \theta' + \sin \theta \sin \theta' \cos (\phi - \phi')\) is the inner product of the two position vectors \(\mathbf{x} = (\sin \theta \cos \phi, \sin \theta \sin \phi, \cos \theta)\) and \(\mathbf{x}' = (\sin \theta' \cos \phi', \sin \theta' \sin \phi', \cos \theta')\). The expression for \(G\) builds in the requirement that the surface integral of the Laplacian, or the vorticity, on a closed surface is precisely zero. The streamfunction is therefore

\[
\psi(\theta, \phi) = \frac{1}{4\pi} \int \int \omega(\theta', \phi') \log(1 - \cos \Theta) \, d\Omega',
\]

where \(d\Omega' = \sin \theta' \, d\theta' \, d\phi'\) is the incremental solid angle or area.

For the contour dynamical equations, consider, without loss of generality, a single contour \(C\) across which the vorticity jumps by \(\tilde{\omega}\) (the uniform, constant vorticity to the immediate left exceeds that to the immediate right by \(\tilde{\omega}\) when the contour is traversed in a right-handed sense; otherwise, exchange “left” and “right”). First, note that the velocity components in spherical coordinates satisfy

\[
\frac{d\theta}{dt} = u_\theta = -\frac{1}{\sin \theta} \frac{\partial \psi}{\partial \phi}, \tag{22a}
\]

\[
\sin \theta \frac{d\phi}{dt} = u_\phi = \frac{\partial \psi}{\partial \theta}. \tag{22b}
\]
From (22a) and (21),
\[
\sin \theta \frac{d\theta}{dt} = -\frac{\tilde{\omega}}{4\pi} \int \int \frac{d}{d\phi} \log(1 - \cos \Theta) \sin \theta' \, d\theta' \, d\phi'
\]
\[= +\frac{\tilde{\omega}}{4\pi} \int \int \frac{d}{d\phi'} \log(1 - \cos \Theta) \sin \theta' \, d\theta' \, d\phi'
\]
\[= -\frac{\tilde{\omega}}{4\pi} \int \log(1 - \cos \Theta) \sin \theta \, d\theta'. \tag{23}\]

The second equality follows because the Green function depends on \(\phi\) and \(\phi'\) only in the combination \(\phi - \phi'\), and the third equality follows from Green's theorem. In the final expression, \(\theta'\) and \(\phi'\) are to be regarded as tracing out the contour \(\mathcal{C}\).

One can also obtain a similar equation for \(\sin \theta \frac{d\phi}{dt}\), but this turns out to be unnecessary. Instead, consider expressing (23) in three Cartesian coordinates; use \(x = \sin \theta \cos \phi\), \(y = \sin \theta \sin \phi\), and \(z = \cos \theta\). First note that, since \(\cos \Theta = x \cdot x'\) and \(|x| = |x'| = 1\), we have
\[1 - \cos \Theta = \frac{1}{2} |x - x'|^2.\]
Next, identifying \(-\sin \theta \, d\phi\) with \(dz\), observe that (23) may be written
\[
\frac{dz}{dt} = -\frac{\tilde{\omega}}{2\pi} \int \log |x - x'| \, dz'. \tag{24}\]

Now, by the symmetry of the sphere, (24) must be equally true for the other two coordinates \(x\) and \(y\), so that
\[
\frac{dx}{dt} = -\frac{\tilde{\omega}}{2\pi} \int \log |x - x'| \, dx', \tag{25}\]

which, apart from the dimensionality of \(x\), are precisely the same equations as those governing flow on an infinite plane. Of course, the three components of (25) are not independent, since \(x \cdot x = 1\) implies \(x \cdot dx/dt = 0\).

The contour surgery algorithm for the spherical case differs little from that for the infinite planar case; for the most part, simply three-dimensional vectors are substituted for two-dimensional ones. The significantly new aspects of the algorithm are described below.

The interpolation between two nodes on a contour, say \(i\) and \(i + 1\), has a new component reflecting the basic curvature of the spherical surface:

\[x_i(p) = x_i + pt_i + \eta(p) n_i + \xi(p) s_i, \tag{26a}\]
\[t_i = x_{i+1} - x_i, \tag{26b}\]
\[n_i = x_i \times x_{i+1}, \tag{26c}\]
\[s_i = (x_i + x_{i+1})/2, \tag{26d}\]
\[\eta(p) = \alpha_i p + \beta_i p^2 + \gamma_i p^3, \tag{26e}\]
\[\xi(p) = \frac{1}{3} e_i^2 p (1 - p). \tag{26f}\]
where $0 < p < 1$ (cf. (11)). As before, $e_i = |t_i| = |n_i|$. The new term involving $\xi(p)$ gives the variation of the contour, in the direction of $s_i$, with errors of $O(e_i^4)$ neglected. $\eta$ gives the variation of the contour in the plane perpendicular to $s_i$. The coefficients $\alpha_i$, $\beta_i$, and $\gamma_i$ are calculated in terms of the curvature $\kappa_i$ in the plane perpendicular to $x_i$ (and the curvature $\kappa_{i+1}$ in the plane perpendicular to $x_{i+1}$) using (14). Explicitly,

$$
\kappa_i = \frac{2x_i \cdot (t_i \times t_{i-1})}{|t_i e_{i-1}^2 + t_{i-1}e_i^2|}
$$

(cf. (12)). Strictly, $x(p)$ lies on the spherical surface only at the endpoints $p = 0$ and $1$; however, the discrepancy between $x(p)$ and the actual surface for $0 < p < 1$ is so small that the cubic accuracy of the interpolation, characteristic of the planar algorithm, is preserved in the spherical one.

The local density of nodes has the same expression as in the planar case except $\kappa_i$ is replaced by the total curvature (including that of the spherical surface), $\sqrt{1 + \kappa_i^2}$. Also, the large scale length $L$ appearing in (10) is set to unity, the radius of the sphere.

The velocity field can again be evaluated explicitly to first order in $\eta$ as well as $\xi$. The formulae corresponding to (19), again without essential subscripts, are

$$
\Delta u = Tt + Nn + Ss,
$$

$$
T = 1 - dr - (1 - d) r' - e^2 h_0 + c(\alpha h_1 + \beta h_2 + \gamma h_3) + b(h_1 - h_2),
$$

$$
N = \alpha q_1 + \beta q_2 + \gamma q_3, \quad S = \frac{1}{2}e^2(q_1 \quad q_2), \quad a = |x - x_i|^2/e^2,
$$

$$
b = \frac{1}{2}s \cdot (x - x_i), \quad c = \frac{n \cdot (x - x_i)}{e^2}, \quad d = \frac{t \cdot (x - x_i)}{e^2},
$$

$$
\hat{c} = \frac{|t \times (x - x_i)|}{e^2}, \quad e = |t|, \quad h_0 = \frac{1}{\hat{c}} \left[ \tan^{-1}\left(\frac{1 - d}{\hat{c}}\right) + \tan^{-1}\left(\frac{d}{\hat{c}}\right) \right],
$$

$$
h_1 - r' - r + dh_0, \quad h_n = \frac{1}{n - 1} 2dh_{n-1} - ah_{n-2} \quad (n > 1),
$$

$$
q_n = h_{n+1} - dh_n, \quad r = \log |x - x_i|, \quad r' = \log |x - x_{i+1}|.
$$

As in the planar case, these formulae must be modified slightly when the evaluation point, projected onto the plane perpendicular to $s_i$, lies within the circle in the plane whose diameter connects $x_i$ with $x_{i+1}$. This condition is met when $d > a$, and one can still use (28) if $a$ is decremented by $\eta(d)[2c - \eta(d)]$, then $c$ is decremented by $\eta(d)$ (with the consequent implicit changes to all other quantities involving $a$ and $c$), and finally $T$ is decremented by $ch_0\eta(d)$.

It is not a mere coincidence that (19) and (28) look similar. Eq. (28) reduces to (19) simply by dropping terms in (28) which multiply $b$ and $s$, replacing $\hat{c}$ by $c$, and thinking of all vectors as being two-dimensional.
The surgical operations for the spherical case are precisely analogous to those in the planar case because, at the scale of surgery, the sphere may be assumed locally planar.

Finally, numerical errors associated with time-stepping, spatial discretization, velocity evaluation, and surgery cause points to displace slightly off the spherical surface. Whenever this occurs, points are adjusted radially such that $|x_i| = 1$ for each node $i$.

A calculation is presented in section 6.

4.9. Cylindrical or singly periodic contour dynamics/surgery

Contour dynamics may also be formulated for flow on the surface of an infinitely long cylinder. Let $z$ be the coordinate along the axis of the cylinder and $\phi$ be the azimuthal angle (the other two Cartesian coordinates are $x = \cos \phi$ and $y = \sin \phi$). For a cylinder of unit radius, Laplace’s equation takes the simple form

$$\Delta \psi = \frac{\partial^2 \psi}{\partial \phi^2} + \frac{\partial^2 \psi}{\partial z^2} = \omega(\phi, z), \tag{29}$$

a form which is not different from that for flow on a planar surface. The new feature, however, is the periodicity of one of the coordinates, $\phi$. The periodicity leads to a more complicated Green function $G$ (see ref. [42]) which, however, has the requisite property for contour dynamics,

$$G(\phi, z; \phi', z') = \frac{1}{4\pi} \log \left\{ 4 \sin^2 \left[ (\phi - \phi')/2 \right] + 4 \sinh^2 \left[ (z - z')/2 \right] \right\}, \tag{30}$$

so that, by (2),

$$\frac{d\phi}{dt} = -\sum_k \tilde{\omega}_k \int_{\mathcal{C}_k} G(\phi, z; \phi_k, z_k) \, d\phi_k \tag{31a}$$

$$\frac{dz}{dt} = -\sum_k \tilde{\omega}_k \int_{\mathcal{C}_k} G(\phi, z; \phi_k, z_k) \, dz_k, \tag{31b}$$

where $\phi_k$ is the azimuthal angle of a point on the $k$th contour $\mathcal{C}_k$, and $z_k$ is the corresponding axial coordinate. In the following, the notation $\phi'$ and $z'$ is used to denote a point on any of the contours $\mathcal{C}_k$.

One may devise numerical algorithms for (31) (see, e.g., refs. [52–59]), but it is actually simpler to work in three Cartesian coordinates $(x, y, z)$ constrained to the surface of the cylinder $(x = \cos \phi, y = \sin \phi)$. One then avoids the problem of $\phi$ jumping discontinuously; the Cartesian coordinates along any contour on a cylindrical surface vary continuously. Furthermore, the Green function $G$ simplifies, and one may use a slightly modified version of the spherical contour surgery algorithm discussed above.

In Cartesian coordinates, the $x$- and $y$-velocity components are related to $\Omega = d\phi/dt$ by

$$\frac{dx}{dt} = \frac{d\cos \phi}{dt} = -\sin \phi \frac{d\phi}{dt} = -y \Omega, \tag{32a}$$

$$\frac{dy}{dt} = x \Omega, \tag{32b}$$
the differential $d\phi$ needed to evaluate $\Omega$ is simply $x' \, dy' - y' \, dx'$, and the Green function becomes

$$G(x, x') = -\frac{1}{4\pi} \log \left| x - x' \right|^2 + H(z - z'),$$

(33)

where $H(\xi) = e^\xi + e^{-\xi} - 2 - \xi^2$. Note that, when $\left| x - x' \right|^2$ is small, $H(z - z')$ is much smaller because $H(\xi) = \frac{1}{12} \xi^4 + O(\xi^6)$ for small $\xi$. One may therefore split $G$ into the parts $G_a = -\left(1/4\pi\right) \log \left| x - x' \right|^2$ and $G_b = -\left(1/4\pi\right) \log \left[1 + H(z - z')/\left| x - x' \right|^2\right]$, the latter being well-behaved for $\left| x - x' \right|^2$ small. In the numerical algorithm discussed next, integrals involving $G_a$ are calculated explicitly as in the spherical case while those involving $G_b$ are calculated by numerical quadrature.

The contour surgery algorithm closely parallels the spherical one above. For example, the contour shape $x'(p)$ between adjacent nodes has the same form as that given in (26) except $s_i = (\frac{1}{2}(x_i + x_{i+1}), \frac{1}{2}(y_i + y_{i+1}), 0)$ (the $z$-component is the only one that differs), $n_i = s_i \times t_i$, and $\hat{s}_i^2 = t_{xi}^2 + t_{yi}^2$ is used in place of $e_i^2$ in (26f) ($t_i = (t_{xi}, t_{yi}, t_{zi})$). These changes occur because the surface does not bend in the $z$-direction. Similarly, the expression for the curvature $\kappa_i$, the curvature in the plane tangent to the cylinder at $x_i$, is given by the spherical expression (27) except $x_i$ in (27) is replaced by $(x_i, y_i, 0)$.

The velocity $u$ of a point $x$ on the cylindrical surface is divided into two parts, the first $u_a$ being calculated in terms of the logarithmic part of Green's function $G_a$ and the second $u_b$ in terms of the nonsingular part $G_b$. Consider $u_a$ first. As in (16)–(18), $u_a$ is expanded in a perturbation series in the small departures, $\eta$ and $\xi$, of the contour between adjacent nodes, say $i$ and $i + 1$. Noting that $dz'/dp = t_{zi} + (d\eta/dp)n_{zi}$ and $d\phi'/dp = n_{zi} - (d\eta/dp)t_{zi} + [\xi - (p - \frac{1}{2}) d\xi/dp]n_{zi}$ (to first order in $\eta$ and $\xi$), one obtains formulae analogous to the spherical ones (28) for $\Omega_a$ and $w_a$:

$$\Omega_a = \frac{1}{2\pi} \sum_i \omega_i \left[ (T_i + S_i) n_{zi} - N_i t_{zi} \right],$$

(34a)

$$w_a = \frac{1}{2\pi} \sum_i \omega_i (T_i t_{zi} + N_i n_{zi})$$

(34b)

(recall $u = -y\Omega$ and $v = x\Omega$). Indeed, the quantities $T$ and $N$ (suppressing subscripts) have the definitions given in (28), while $S$ is only slightly changed to

$$S = \frac{1}{2} \hat{s}_i^2 (q_1 - q_2 + \frac{3}{2}q_3).$$

Next, consider $u_b$. Since $G_b$ is a smoothly varying function, $\Omega_b$ and $w_b$ can be evaluated safely by three-point Gaussian quadrature,

$$\Omega_b = \frac{1}{2\pi} \sum_i \omega_i \sum_{l=1}^{3} \sigma_l G_b(x; x'(p_l)) \frac{d\phi'}{dp}(p_l),$$

(35a)

$$w_b = \frac{1}{2\pi} \sum_i \omega_i \sum_{l=1}^{3} \sigma_l G_b(x; x'(p_l)) \frac{dz'}{dp}(p_l),$$

(35b)
where $p_1 = 0.112701665379\ldots$, $p_2 = 0.5$, $p_3 = 1 - p_1$, $\sigma_1 = 5/18$, $\sigma_2 = 8/18$, and $\sigma_3 = 5/18$ [74]. The complete velocity field results from using (34) and (35).

Surgery is performed just in the spherical case, which, it will be recalled, differs from the planar case only in the use of three-dimensional vectors. Finally, nodes tend to displace slightly from the cylindrical surface as a consequence of the numerical approximations, so they are adjusted by scaling the $x$ and $y$ coordinates so that $x_i^2 + y_i^2 = 1$ for all nodes $i$.

A couple of calculations are given in section 6.

4.10. Three-dimensional quasi-geostrophic contour dynamics / surgery

Rapidly rotating, strongly stratified fluids, such as (parts of) the atmosphere and the oceans (at times), may be approximately described by the three-dimensional, quasi-geostrophic equations (see ref. [75]). These equations have the property that the potential vorticity (or rather the quasi-geostrophic approximation thereof) is materially conserved on each isentropic (constant entropy or potential temperature) surface when the fluid is inviscid. Assuming that the fluid consists of a finite number of stacked, uniform-density layers, the conservation of potential vorticity applies, in a vertically integrated sense, to each layer as a whole (see ref. [75], p. 386) — that is, two-dimensional conservative dynamics applies to each of a finite number of layers.

Consider $N$ layers of fluid bounded below and above by rigid, frictionless surfaces. Let $\omega_n$ denote the potential vorticity (hereafter, read “potential vorticity” for “vorticity”) of the $n$th layer and $\psi_n$ the corresponding streamfunction. The velocity components are related to the streamfunction in the usual way, $u_n = -\partial \psi_n / \partial y$, $v_n = \partial \psi_n / \partial x$, but the vorticity, while materially conserved ($\partial \omega_n / \partial t + u_n \partial \omega_n / \partial x + v_n \partial \omega_n / \partial y = 0$), depends on the streamfunction in several layers:

\[
\begin{align*}
\omega_1 &= \Delta \psi_1 - \lambda_1^2 \psi_1 + \lambda_2^2 \psi_2, \\
\omega_n &= \Delta \psi_n + \lambda_{n-1}^2 \psi_{n-1} - 2 \lambda_n^2 \psi_n + \lambda_{n+1}^2 \psi_{n+1}, \quad n = 2, \ldots, N-1, \\
\omega_N &= \Delta \psi_N + \lambda_{N-1}^2 \psi_{N-1} - \lambda_N^2 \psi_N. 
\end{align*}
\]

Here,

\[
\lambda_n^2 = f_0^2 \rho_n / gD_n \Delta \rho,
\]

where $f_0$ is the Coriolis parameter, $\rho_n$ is the density of the $n$th layer, $g$ is the acceleration due to gravity, $D_n$ is the thickness of the $n$th layer, and $\Delta \rho$ is the density difference between adjacent layers (assumed equal for all layers). For a single layer, (36) reduces to $\omega = \Delta \psi - \lambda^2 \psi$, and $\lambda^{-1}$ is often referred to as the Rossby radius of deformation.

The structure of (36) makes the inversion to find the $\psi_n$ in terms of the $\omega_n$ more challenging than before, and we need this inversion to find out if contour dynamics is possible. Let $\psi = (\psi_1, \psi_2, \ldots, \psi_N)$, etc., and rewrite (36) as

\[
\Delta \psi - T \psi = \omega,
\]
where $\mathbf{T}$ is a tridiagonal matrix comprised of the constants $\lambda_n^2$. Next, find all of the eigenvalues $\alpha_n^2$ and associated unit eigenvectors $\mathbf{X}_n$ of the matrix $\mathbf{T}$ ($\mathbf{T}\mathbf{X}_n = \alpha_n^2\mathbf{X}_n$, $n = 1, 2, \ldots, N$). The most general solution to (37) therefore has the form

$$\psi = \sum_{n=1}^{N} \varphi_n(x, y, t) \mathbf{X}_n,$$

(38)

and substituting this expression into (37) yields

$$\sum_{n=1}^{N} [\Delta \varphi_n - \alpha_n^2 \varphi_n] \mathbf{X}_n = \mathbf{\omega},$$

and finally, therefore,

$$\Delta \varphi_n - \alpha_n^2 \varphi_n = \mathbf{X}_n \cdot \mathbf{\omega}$$

(39)

after multiplication by $\mathbf{X}_n$ ($\mathbf{X}_n \cdot \mathbf{X}_m = \delta_{mn}$).

Eq. (39) is precisely what is needed to establish the possibility of contour dynamics. The left-hand side is a simple Helmholtz operator acting on $\varphi_n$ while the right-hand side is a weighted sum of the vorticities in the $N$ layers. Green’s function $G$ corresponding to this operator involves the modified Bessel function $K_0$,

$$G(x, x') = F_m(x - x') = -\frac{1}{2\pi} K_0(\alpha_m |x - x'|),$$

(40)

except for the single zero eigenvalue, say $\alpha_1 = 0$, for which $F_1 = (2\pi)^{-1} \log |x - x'|$. Green’s function therefore has the required form for contour dynamics, so that when the $\omega_n$ are piecewise constant,

$$\mathbf{v}_m(x, t) = \left(-\frac{\partial \varphi_m}{\partial y}, \frac{\partial \varphi_m}{\partial x}\right) = -\sum_{n=1}^{N} \mathbf{X}_{nm} \sum_{k=1}^{N_n} \mathbf{\omega}_{kn} \int_{\xi_{kn}} F_m(x - x_{kn}) \, dx_{kn},$$

(41)

where $\mathbf{X}_{nm}$ is the $n$th component of $\mathbf{X}_m$, $N_n$ is the number of contours in the $n$th layer, $\mathbf{\omega}_{kn}$ is the vorticity jump across the $k$th contour $\xi_{kn}$ in the $n$th layer, and $x_{kn}$ is a point on the $k$ th contour in the $n$th layer. Finally, one obtains the velocity of a point $x$ in the $l$th layer by a linear combination of the $\mathbf{v}_m$:

$$\mathbf{u}_l = \sum_{m=1}^{N} \mathbf{X}_{lm} \mathbf{v}_m.$$  

(42)

Numerical methods for one- and two-layer quasi-geostrophic contour dynamics have been developed by Stern [76], Stern and Pratt [77], Polvani [78], Polvani, Zabusky and Flierl [79], and Helfrich and Send [80] (see also the earlier work of Kozlov [81] in which a three-layer calculation is presented). Here, the extensions of the contour surgery algorithm to the general quasi-geostrophic case are noted.
Apart from some simple matrix algebra, the contour dynamics equations (40)-(42) differ little from those governing planar, two-dimensional flow. Consequently, the contour surgery algorithm for planar, two-dimensional flow discussed above may be extended to three-dimensional, quasi-geostrophic flow with minor effort. Indeed, the only truly novel feature stems from the evaluation of contour integrals involving the modified Bessel function \( K_0 \) (see (40)). These cannot be evaluated explicitly, in contrast to integrals involving \( \log |x - x'| \) (see (17)-(19)). However, \( K_0(\alpha_m, \xi) \) can be split into a logarithmic part \(-\log(\xi)\) and a nonsingular part, denoted \( R_n(\xi) \). Integrals over the logarithmic part are evaluated explicitly as before, while those over the nonsingular part are evaluated using three-point Gaussian quadrature (cf. (35)). Series for \( R_n(\xi) \) suited for efficient numerical calculation may be found in ref. [74] (p. 379).

A one-layer calculation is presented in section 6.

5. Accuracy

Tests are presented in this section which quantify the accuracy of the individual components of the planar contour surgery algorithm as well as the full algorithm. The first tests examine just how accurately the algorithm can interpolate and calculate the velocity for an elliptical contour, a contour shape for which the boundary and the velocity field is known analytically [42]. A further, more stringent test examines the error in the positions of the nodes after one time step and node redistribution, for both an elliptical contour and an unsteady, distorted elliptical contour.

The second series of tests considers the evolution, over many time steps, of both forced, periodic flows and unforced flows which develop small-scale structure. The periodic flows are strained elliptical vortices [82] which remain elliptical, in theory, and which have a definite period of evolution that can be tested. The unforced flows are linearly unstable, disturbed ellipses, and for these, visual comparisons are presented between a calculation done at very high resolution (small \( \Delta t, \mu, \) and \( \delta \)) and calculations done at coarser resolution. In addition, the accumulation of phase error (defined below) is calculated for each of the calculations.

Finally, a comparison is presented between contour surgery and a standard pseudo-spectral method.

5.1. Instantaneous and short-time accuracy

An elliptical contour may be described parametrically by \( x = \cos \theta, y = \lambda^{-1} \sin \theta, 0 \leq \theta \leq 2\pi \), for any aspect ratio \( \lambda \). The undisturbed contour rotates uniformly at the rate \( \Omega = \omega \lambda/(\lambda + 1)^2 \) where \( \omega \) is the vorticity within the contour. (In all of the examples presented below, \( \omega = 2\pi \).)

The first series of tests assumes \( \delta = 0, L = 1 \), but \( a = 1/3, 2/3, \) or \( 1 \) (cf. (8) and (10)). \( n \) nodes \( (n = 256 \) or 512\) are first placed at equal intervals in \( \theta \) and then eqs. (10)-(15) are applied five times to achieve the desired node density distribution (depending on \( a \)), each time dividing \( x_i \) and \( \lambda y_i \) for each node \( i \) by the factor \( \sqrt{x_i^2 + \lambda^2 y_i^2} \) to keep the nodes on the elliptical contour.

Two measures of error are considered. The first \( \epsilon_\Lambda \) measures interpolation errors,

\[
\epsilon_\Lambda = \frac{\lambda}{\pi} \sum_{i=1}^{n} e_i^2 \int_0^1 |\eta - \eta_e| \, dp,
\]

(43a)
Table 2
Velocity and area errors for three distributions of nodes on an elliptical contour \(^a\)

<table>
<thead>
<tr>
<th>(\lambda)</th>
<th>(a = 1/3)</th>
<th>(a = 2/3)</th>
<th>(a = 1)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(C_u) (C_A)</td>
<td>(C_u) (C_A)</td>
<td>(C_u) (C_A)</td>
</tr>
<tr>
<td>1</td>
<td>0.0290</td>
<td>0.0125</td>
<td>0.0290</td>
</tr>
<tr>
<td>2</td>
<td>0.0423</td>
<td>0.116</td>
<td>0.0288</td>
</tr>
<tr>
<td>3</td>
<td>0.0658</td>
<td>0.247</td>
<td>0.0287</td>
</tr>
<tr>
<td>4</td>
<td>0.0944</td>
<td>0.385</td>
<td>0.0288</td>
</tr>
<tr>
<td>5</td>
<td>0.127</td>
<td>0.527</td>
<td>0.0291</td>
</tr>
<tr>
<td>6</td>
<td>0.164</td>
<td>0.673</td>
<td>0.0295</td>
</tr>
<tr>
<td>7</td>
<td>0.205</td>
<td>0.821</td>
<td>0.0299</td>
</tr>
<tr>
<td>8</td>
<td>0.250</td>
<td>0.971</td>
<td>0.0305</td>
</tr>
<tr>
<td>9</td>
<td>0.299</td>
<td>1.12</td>
<td>0.0311</td>
</tr>
<tr>
<td>10</td>
<td>0.352</td>
<td>1.28</td>
<td>0.0317</td>
</tr>
</tbody>
</table>

\(^a\) \(a\) is the power to which the curvature is raised in the expression for the local density of nodes \(\rho\), \(C_u\) and \(C_A\) are the coefficients of the velocity and area errors \(\epsilon_u\) and \(\epsilon_A\) defined in the text, that is \(\epsilon_u = C_u\hat{\mu}^3\) and \(\epsilon_A = C_A\hat{\mu}^4\), \(\hat{\mu} = 2\pi/n\), and \(n\) is the number of nodes put on the contour. Here, \(n = 512\) nodes are used. The comparison is shown as a function of the aspect ratio \(\lambda\) of the ellipse.

and the second \(\epsilon_u\) measures velocity errors,

\[
\epsilon_u^2 = \sum_{i=1}^{n} |u_i - u_{ei}|^2 / \sum_{i=1}^{n} |u_{ei}|^2.
\] (43b)

Here, \(\eta(p)\) is the known normal variation of the ellipse from a straight line segment connecting two adjacent nodes, \(\eta(p)\) is the approximate variation found from (11), (12), and (14), \(u_{ei}\) is the known velocity at the point \(x_i\), and \(u_i\) is the velocity calculated by (19).

The errors found for \(n = 256\) and 512 nodes confirm the asymptotic dependences,

\[
\epsilon_A = C_A(\lambda, a)\hat{\mu}^4,
\] (44a)

\[
\epsilon_u = C_u(\lambda, a)\hat{\mu}^3,
\] (44b)

where \(\hat{\mu} = 2\pi/n\), and table 2 lists the coefficients \(C_A\) and \(C_u\) for ten different values of the aspect ratio and three different values of \(a\). Overall, the choice \(a = 2/3\) leads to the smallest errors, particularly with regard to the computed velocity field, whose error shows little sensitivity to the aspect ratio of the ellipse. Indeed, results for intermediate values of \(a\) in a confirm that \(a = 2/3\) minimizes \(C_u\) for the values of \(\lambda\) tested. It was on this basis that \(a = 2/3\) was chosen to be the operational value.

It is interesting to compare these errors with those obtained with simple linear interpolation between nodes. It may be shown analytically that the minimum interpolation error occurs when \(a = 1/3\) or when the node density is proportional to the one-third power of curvature [62], and in this case \(\epsilon_A = \hat{\mu}^2/6\) (asymptotically), independent of \(\lambda\). The corresponding velocity error, computed using (43b) and (19) when \(\alpha = \beta = \gamma = 0\) for all nodes, has the asymptotic dependence \(\epsilon_u = C\hat{\mu}^2\) where \(C = 0.084, 0.089, 0.105,\) and 0.136 for \(\lambda = 1, 2, 4,\) and 8 respectively. Note that
the order of accuracy of the velocity error is the same as that of the interpolation error. This is not true of the cubic-interpolation scheme above; while the interpolation error is two orders more accurate than that for linear interpolation, the velocity error is only one order more accurate. An extra order of accuracy could be gained by computing the velocity field to quadratic order in the departure of the contour from a straight line segment between nodes (i.e. including $\Delta u_{ij}$ in (19)); however, the great number of additional terms this would introduce would appear to reduce the overall computational efficiency to an extent which outweighs the gain in order of accuracy.

Even though the velocity error of cubic interpolation is only one order more accurate than that of linear interpolation, the extra effort needed to calculate the velocity field using cubic interpolation is worthwhile. This follows because more nodes are required in linear interpolation than in cubic interpolation for the same velocity accuracy $\epsilon_v$. Consider, for example, the above results for the elliptical vortex when $\lambda = 4$. The number of nodes $n_r$ in linear interpolation necessary to make $\epsilon_v$ the same as in cubic interpolation (using $n_c$ nodes) is $n_r = 0.762n_c^{3/2}$. But the cost, the seconds of CPU time needed per time step, rises with the square of the number of nodes in each case. In particular, using a two-pipe, 64-bit, vector-processing CYBER 205 supercomputer, the cost of cubic interpolation is $T_c = 1.03 \times 10^{-5}n_c^2 + 1.36 \times 10^{-3}n_c$, whereas the corresponding cost of linear interpolation is $T_l = 7.81 \times 10^{-6}n_c^2 + 1.00 \times 10^{-3}n_c$; in effect, $T_c$ rises with the cube of the number of nodes used in cubic interpolation. (Similar, although about 30–40% smaller results have been obtained using one processor of a CRAY X-MP supercomputer.) For $n_c = 100$, $T_c = 0.239$ s and $T_l = 8.81$ s, so that the calculation with cubic interpolation is far more efficient. Calculations presented below use hundreds to thousands of nodes, so there is no question that cubic interpolation is worthwhile.

In Buttke's [66] hybrid vortex element–contour dynamics method (hereafter BHM), the computational cost is simply linear in the number of elements, and so one might expect his method to be far more efficient than contour dynamics with linear interpolation (hereafter CDL). Briefly, an element in BHM is a square region of variable size contained within a region of uniform vorticity. The smallest elements, of size $\Delta \xi$ by $\Delta \xi$, straddle the vortex boundary, itself represented by a polygon, as in CDL. By judicious choice of the element sizes, the total number of elements $n_e$ needed to fill a vortex is proportional to $P/\Delta \xi$ where $P$ is the perimeter of the vortex. To compare BHM with CDL, however, it is necessary to estimate to cost of BHM in terms of the total number of boundary nodes, $n_r$. The connection between $n_e$ and $n_r$ in BHM follows from the accuracy results reported in ref. [66], where it is verified that the root-mean-square area and velocity errors are proportional to $P/\Delta \xi$ and the inverse square of the number of boundary nodes, $1/n_r^2$, the latter error being the same as in CDL. In order that BHM and CDL have comparable accuracy, it follows that $P/\Delta \xi$ must be proportional to $1/n_r^2$. That is to say, the number of elements, proportional to $P/\Delta \xi$, scales as $P^2n_r^2$. Therefore, the computational cost, while linear in the number of elements, is actually quadratic in the number of nodes, just as in CDL. Hence, BHM is not likely to be significantly faster than CDL, particularly when it is taken into account that the total number of computer operations (e.g., additions) per boundary node in CDL is only a few tens. Furthermore, since the cost of BHM also rises with the square of the vortex perimeter, and since BHM at present lacks some form of surgery, it would appear that BHM is even more severely restricted than CDL from performing long time, accurate calculations of complex flows.
Table 3
The rms normal node displacement error after one time step and node redistribution for an elliptical contour a)

<table>
<thead>
<tr>
<th>λ</th>
<th>a = 1/3</th>
<th>a = 2/3</th>
<th>a = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.8440</td>
<td>0.8440</td>
<td>0.8440</td>
</tr>
<tr>
<td>2</td>
<td>2.735</td>
<td>2.814</td>
<td>3.007</td>
</tr>
<tr>
<td>3</td>
<td>5.302</td>
<td>5.349</td>
<td>4.980</td>
</tr>
<tr>
<td>4</td>
<td>6.892</td>
<td>6.800</td>
<td>6.676</td>
</tr>
<tr>
<td>5</td>
<td>7.777</td>
<td>7.697</td>
<td>6.451</td>
</tr>
<tr>
<td>6</td>
<td>8.211</td>
<td>8.103</td>
<td>17.68</td>
</tr>
<tr>
<td>7</td>
<td>8.533</td>
<td>7.831</td>
<td>19.60</td>
</tr>
<tr>
<td>8</td>
<td>8.712</td>
<td>7.722</td>
<td>12.64</td>
</tr>
<tr>
<td>9</td>
<td>9.042</td>
<td>7.725</td>
<td>32.04</td>
</tr>
<tr>
<td>10</td>
<td>9.629</td>
<td>7.782</td>
<td>56.48</td>
</tr>
</tbody>
</table>

a) Shown is $10^7$ times the error $\epsilon_t$ (defined in the text). $a$ is the power to which the curvature is raised in the expression for the local density of nodes $\rho$, and $n = 256$ nodes are used at a time step $\Delta t = 0.05$. $\lambda$ is the aspect ratio of the ellipse.

Consider next the error accumulated over one time step resulting from the time integration scheme, the velocity computation, the interpolation, and the node redistribution. $n = 256$ nodes are distributed along an elliptical contour as described above. Then, the system of nodes is advanced one time step using a typical time step of $\Delta t = 0.05$, and redistributed using (10)–(15) (recall $\delta = 0$ and $L = 1$). Next, the normal distance $d_i$ of each node from the known position of the ellipse is computed, and from this the following error is defined:

$$
\epsilon_t^2 = \frac{\lambda}{n} \sum_{i=1}^{n} d_i^2
$$

(45)

(the factor $\lambda$ is included because $\lambda^{-1/2}$ is a representative length scale for the ellipse). Table 3 lists $10^7 \epsilon_t(\lambda, a)$ for ten values of $\lambda$ and three values of $a$. The choice $a = 2/3$ continues to give the best overall results while $a = 1$ is seen to behave poorly.

Similar accuracy results apply also to unsteady contours. The procedure to obtain $\epsilon_t$ differs somewhat from that for the ellipse above because the exact time evolution of the unsteady contour is unknown. Instead, the unsteady evolution is compared to a calculation of the same flow done at twice the spatial and temporal resolution, and without node redistribution. Consider a contour shape initially defined by $x = \cos \theta$, $y = (1 - x/x_0) \lambda^{-1} \sin \theta$, $0 < \theta < 2\pi$. Here, $x_0$ is a free parameter and the limit $x_0 \to \infty$ gives back an elliptical contour. The coarse-resolution calculation has the same parameter values as above, $n = 256$ nodes and $\Delta t = 0.05$ (so the fine-resolution calculation has $n = 512$ and $\Delta t = 0.025$). As a test of the method, the result for $x_0 \to \infty$ and $\lambda = 1$ is $\epsilon_t = 1.007 \times 10^{-7}$ compared to $0.844 \times 10^{-7}$ in table 3. In a representative unsteady contour calculation, such as for $\lambda = 5$ and $x_0 = 2$, $\epsilon_t = 4.921 \times 10^{-7}$ for $a = 1/3$, $3.695 \times 10^{-7}$ for $a = 2/3$, and $8.622 \times 10^{-7}$ for $a = 1$. Again, the value $a = 2/3$ performs best.
5.2. Accuracy of extended integrations

Two series of calculations are discussed, each of which compares calculations of the same flow done at differing spatial and temporal resolution. The first series tests the algorithm's ability to reproduce the known, periodic evolution of an elliptical vortex in a uniform strain field [82,83] (both the aspect ratio and orientation of the vortex vary periodically with time). The second series of calculations examines the reproducibility of the unforced evolution of a disturbed, linearly unstable elliptical vortex. This evolution generates much fine-scale structure and so directly tests surgery.

Kida [82] has shown that a uniform elliptical vortex subjected to the strain forcing,

$$u_s = \gamma x - \Omega y, \quad v_s = -\gamma y + \Omega x,$$

remains elliptical for all time. Here, $\gamma$ is the strain rate and $\Omega$ is (minus) the rotation rate of the principal axes of strain. If the strain rate is sufficiently small compared to the vorticity $\omega$ within the vortex, then the aspect ratio and orientation of the ellipse vary periodically [84].

To what extent can the contour surgery algorithm reproduce this known periodic behavior? Suppose the vortex is initially circular. Subjected to pure strain ($\Omega = 0$) of magnitude $\gamma = 0.1\omega$, the vortex reaches a maximum aspect ratio of $\lambda = 2.626269818714\ldots$ before returning to a circular shape at $t = T = 17.04512066595\ldots/\omega$ (see ref. [84]). Consider then a series of numerical calculations beginning with a circular vortex, with $\omega - 2\pi$, and with $T$ being an integer multiple of time steps $\Delta t$. Table 4 lists $\epsilon_i$ (45), the root-mean-square displacement of nodes from a circular boundary at $t = T$, for five calculations differing in spatial and temporal resolution. In general, higher spatial or temporal resolution results in smaller errors, but not always. For example, when $\Delta t = T/28$, a refinement of spatial resolution from $\mu = 0.04$ to $\mu = 0.02$ does not reduce $\epsilon_i$; however, when the time step is halved, $\epsilon_i$ drops significantly. (Note: smaller time steps are typically required in such strained calculations because the maximum velocity gradients can be greater.)

Next, consider a series of calculations for an unforced but unstable vortex. The initial state for each calculation consists of an ellipse of uniform vorticity $\omega = 2\pi$ and aspect ratio $\lambda = 4$

Table 4
The rms normal node displacement error after one period of evolution of a strained elliptical contour a)

<table>
<thead>
<tr>
<th>Case</th>
<th>$\mu$</th>
<th>$\Delta t$</th>
<th>$n_{\text{min}}$</th>
<th>$n_{\text{max}}$</th>
<th>$\epsilon_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.02</td>
<td>$T/56$</td>
<td>310</td>
<td>324</td>
<td>$5.180 \times 10^{-6}$</td>
</tr>
<tr>
<td>2</td>
<td>0.04</td>
<td>$T/56$</td>
<td>162</td>
<td>167</td>
<td>$5.000 \times 10^{-6}$</td>
</tr>
<tr>
<td>3</td>
<td>0.06</td>
<td>$T/56$</td>
<td>112</td>
<td>115</td>
<td>$2.817 \times 10^{-5}$</td>
</tr>
<tr>
<td>4</td>
<td>0.02</td>
<td>$T/112$</td>
<td>310</td>
<td>324</td>
<td>$8.928 \times 10^{-7}$</td>
</tr>
<tr>
<td>5</td>
<td>0.02</td>
<td>$T/28$</td>
<td>310</td>
<td>324</td>
<td>$1.083 \times 10^{-4}$</td>
</tr>
</tbody>
</table>

a) $\mu$ is a dimensionless parameter proportional to the average spacing between nodes along the contour, the cutoff scale $\delta = \frac{1}{2} \mu^2$ in all cases, $\Delta t$ is the time step, $T$ is the period of evolution of the strained ellipse, and $n_{\text{min}}$ and $n_{\text{max}}$ are the minimum and maximum number of nodes used during the calculations.
Fig. 6. (a) The evolution of an unstable ellipse. Time proceeds to the right and down the page. $\mu = 0.02$, $\delta = 5 \times 10^{-5}$ and $\Delta t = 0.05$. $\epsilon_c = 0.067^\circ$ by $t = 20.35$. In this and all subsequent calculations, the peak or only value of vorticity is $\omega = 2\pi$, unless otherwise stated. (b) Shaded image of the vortex at $t = 20$. 
disturbed with a superposition of the sole linearly unstable eigenmode and its stable conjugate (see ref. [50]); the initial contour shape is

\[ x(\theta, 0) = (\cos \theta, \lambda^{-1} \sin \theta) + \frac{a \lambda^{-1} \cos m \theta}{\sin^2 \theta + \lambda^{-2} \cos^2 \theta} (\lambda^{-1} \cos \theta, -\sin \theta), \]  

(47)

with \( \lambda = 4, \ a = 0.005, \) and \( m = 3. \) The period of rotation of the undisturbed vortex is 6.25. One of the three calculations, the one performed at the highest spatial and temporal resolution (\( \mu = 0.02, \ \Delta t = 0.05 \)), took six hours of single-processor CPU time on a CRAY X-MP to reach \( t = 20.35 \) (see fig. 6) during which the total number of nodes climbed from 239 to 8357. The other two calculations used either twice \( \mu \) (coarser spatial resolution) or twice \( \Delta t \) (coarser temporal resolution) and were carried to \( t = 20 \) (see table 5). Fig. 7 compares the coarser resolution calculations (dotted lines) with the fine resolution calculation (solid lines) at \( t = 20 \) and at different magnifications. Importantly, all three calculations differ insignificantly on the inter-

<table>
<thead>
<tr>
<th>Case</th>
<th>( \mu )</th>
<th>( \Delta t )</th>
<th>( n_{\min} )</th>
<th>( n_{\max} )</th>
<th>( \epsilon_C )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.02</td>
<td>0.05</td>
<td>239</td>
<td>7542</td>
<td>0.0642</td>
</tr>
<tr>
<td>2</td>
<td>0.02</td>
<td>0.10</td>
<td>239</td>
<td>8034</td>
<td>0.475</td>
</tr>
<tr>
<td>3</td>
<td>0.04</td>
<td>0.05</td>
<td>128</td>
<td>2359</td>
<td>2.28</td>
</tr>
</tbody>
</table>

\( a) \) \( \mu \) is a dimensionless parameter proportional to the average spacing between nodes along the contour, the cutoff scale \( \delta = \frac{1}{2} \mu^2 \) in all cases, \( \Delta t \) is the time step, \( n_{\min} \) and \( n_{\max} \) are the minimum and maximum number of nodes used during the calculations, and \( \epsilon_C \) is the phase error accumulated by \( t = 20. \)
mediate to large scales, despite the complex history of the vortex's evolution. And even on the smaller scales, the coarser resolution calculations capture much of the detail, if not the precise location of this detail. The precise dynamics of scales of motion orders of magnitude smaller
than the largest scale do not appear to significantly affect the dynamics at the intermediate to large scales, at least for moderate integration times.

This error in the positions of structures is called "phase error". It can be quantified, at least in a global sense, by monitoring the degree to which various global invariants depart from their exact values. To be precise, the phase error $\epsilon_c$ accumulated by time $t = t_m$ is defined by

$$\epsilon_c = \Delta t \sum_{t=1}^{m} |C(t_i) - C(0)|,$$

(48)

with $t_i = l \Delta t$ and $C = \Gamma^2/4\pi J$, the ratio of the circulation squared to the angular impulse. Here, $\Gamma = \iint \omega(x, y) \, dx \, dy$ and $J = \iint \omega(x, y)(x^2 + y^2) \, dx \, dy$ (these and other global invariants have contour integral expressions when $\omega$ is piecewise constant, see the appendix). Table 5 lists $\epsilon_c$ at $t = 20$ for the three calculations. Referring to fig. 7, the greatest perceptible phase error occurs for the calculation done at the coarsest spatial resolution, an observation that is consistent with the higher value of $\epsilon_c$ in this case.

Finally, it is also possible to monitor the nonconservation of (excess) energy (see refs. [31,50] and the appendix) but it was found in ref. [3] that errors in energy, angular impulse, and circulation often closely parallel one another, and that $\epsilon_c$ is therefore a typical measure of errors associated with loss of conservation of global invariants. Of course, tests of reproducibility like those in fig. 7 are the true measures of error, but, in light of the qualitative correspondence between the perceptible errors in fig. 7 and those turned up by the quantitative measure $\epsilon_c$, it does not appear necessary to perform such tests for every calculation. In other words, in most cases it should be sufficient to keep $\epsilon_c$ small. Doing so appears to guarantee reproducibility except at the smallest scales.
5.3. Comparison with a spectral calculation

First results are emerging from a comparative study of CS and traditional pseudo-spectral methods [7]. One of the most important issues is the degree to which a piecewise-constant distribution can model the dynamics of a continuous distribution. That is, how many contours are needed in CS to closely match the evolution observed using a traditional method?

A single comparison is reported. The initial condition consists of a compact circular distribution of vorticity, \( \omega(r, 0) = \pi(1 + \cos \pi r) \) (for \( r < 1 \)) in a doubly periodic box of size \( 2\pi \) by \( 2\pi \). The vortex is embedded in a steady, uniform shear flow, \( u_z = 0.11(2\pi)^y \), causing the vortex to rapidly depart from its initially circular shape (see below). The pseudo-spectral calculation was performed with a very high resolution of 1024 wavenumbers in each direction (with \( \nu \nabla^6 \))

![Fig. 8. The evolution of a sheared vortex in a doubly periodic box using CS.](image-url)
hyper-diffusion on vorticity). The CS calculation was performed using a modified version of the singly periodic or cylindrical CS algorithm (the algorithm parameters are $\mu = 0.04$, $\Delta t = 0.05$, and $\delta = 0.0002$). Eight contours having equal jumps in vorticity ($\bar{\omega} = 2\pi/8$) were used to approximate the continuous vorticity distribution (see ref. [7] for full details).

Fig. 8 shows the evolution computed with CS up to $t = 13.5$. The external shear very effectively removes the vorticity at the periphery of the vortex, leaving an exceedingly sharp "edge" after only a short period of time. The periodic boundary conditions later enable the

---

Fig. 9. (a) A comparison, at $t = 10$, between CS (left) and PS (right). (b) As in (a), except at $t = 13.5$. 
filaments to re-encounter the vortex, and the vortex responds by distorting from its nearly elliptical shape and developing breaking waves on its edge.

But what does the pseudo-spectral calculation do? Perhaps surprisingly, the evolution seen in fig. 8 closely matches the pseudo-spectral (PS) evolution – see fig. 9 which compares the CS and PS calculations at \( t = 10 \) and 13.5. Give the extent to which the vortex is stripped of its low-lying vorticity and the complex interaction between the filaments and the vortex core later in the evolution, the calculations agree remarkably well. And yet, only eight contours were used in the CS calculation. One does see differences, some of which are effects of piecewise-constant vorticity (e.g., the kink on the filaments above and below the vortex core at \( t = 10 \) and the gradual loss of phase coherence of the interior contours by \( t = 13.5 \)), but others are limitations of the PS method (e.g., note the much steeper edge gradients in the CS calculation).

---

Fig. 10. (a) The evolution of a small, initially shallow disturbance to a uniform circular vortex. The initial shape of the vortex boundary, in polar coordinates, is taken to be \( r(\theta, 0) = 1 + a \exp(-\frac{1}{2}(\theta/\theta_0)^2) \) where \( a = 0.05 \) and \( \theta_0 = \pi/20 \). Plotted is \( r(\theta, t) \) versus \( -\theta \) at equally spaced intervals of time \( t \) going down the page, in the left-hand column, and continuing in the right. In linear theory, the initial disturbance repeats itself every two units of time (see ref. [5]). \( \mu = 0.01, \delta = 1.25 \times 10^{-5} \) and \( \Delta t = 0.05 \). \( \epsilon_c = 0.0034^\circ \). (b) An enlarged view of (a) from \( t = 35.75 \) to \( t = 40.5 \).
6. High-resolution calculations using contour surgery

CS has enabled the examination of fluid dynamical phenomena covering four to five orders of magnitude of spatial scales [3–8], that is, ten to a hundred times the maximum resolution ever used in a pseudo-spectral calculation [7,10]. Perhaps the most immediately striking phenomenon to have been revealed is that of “repeated filamentation” [5]. It develops from the undular
motion of small and initially shallow, in fact *arbitrarily* shallow, disturbances to contours of discontinuous vorticity. This eventually gives way to a nearly periodic and apparently endless expulsion of thin filaments (see, for example, fig. 10). The one-sided nature of filamentation seen in fig. 10 is a consequence of the asymmetry of the velocity field on either side of the mean contour position due to the undisturbed vortex. On the inside of the contour, the undisturbed flow is in rigid rotation, and so pairs of fluid particles have less tendency to separate in time (there is no strain in the undisturbed flow). On the outside, however, fluid particles *do* separate, mainly because of the mean shear. This shearing motion, evident in fig. 10 above the mean

![Diagram](image)

**Fig. 11.** Late stages in the filamentation of a contour straddling the equator on a sphere. The initial shape of the contour is $z(\phi, 0) = a \exp\left[-\frac{1}{2}(\phi/\phi_0)^2\right]$ where $a = 0.05$ and $\phi_0 = \pi/20$. ($z$ is the height of the contour above the equatorial plane.) Plotted is $z(\phi, t)$ versus $\phi$ at equally spaced intervals of time $t$. $\mu = 0.01$, $\delta = 1.25 \times 10^{-5}$, $\Delta t = 0.05$, and the vorticity jump across the contour $\bar{\omega} = 2\pi$. $\epsilon_C = 0.00085^\circ$. 
position of the contour, gives rise to the qualitatively different behavior above and below the contour. By contrast, consider the filamentation of a contour straddling the equator on a sphere, fig. 11. In the case, filaments are ejected to both sides of the mean contour position because the mean velocity field has shear on both sides – on the sphere, the total circulation must vanish and therefore the vorticity on either side of an equatorial contour must be of equal magnitude but opposite sign.

Filamentation appears to be a ubiquitous feature of unsteady, aperiodic contour evolution, rather than an accident occurring only for particular disturbances to simple contour shapes. Consider, for example, one of the broadest, simplest, non-trivial disturbances to a circular vortex (on the plane) involving two linear eigenmodes,

\[ r(\theta, 0) = 1 + 0.05 \cos 2\theta + 0.05 \cos 3\theta. \]  

(49)

Fig. 12 shows that the validity of linear theory does not last long; rapidly, the nonlinear interaction between waves of different symmetry lead to steepening and eventually filamentation. A disturbance half the amplitude as that in (49) also leads to filamentation (see fig. 13), only the time-scale of the steepening is stretched four-fold, consistent with the predictions of the weakly nonlinear theory developed in ref. [5]. Analogous results have been obtained for initially gentle disturbances to a periodic interface (cylindrical flow), see, for example, fig. 14.

One might consider filamentation a warning that discontinuous vorticity is a poor assumption in fluid flow, but, in fact, filamentation is not restricted to discontinuous vorticity. Consider replacing the single disturbed circular contour in fig. 10 by four contours, so that the vorticity diminishes to zero in four equal steps and across a zone much thinner than the amplitude or extent of the initial disturbance. While the vorticity is still discontinuous, the four contours capture the single new ingredient associated with a finite gradient of vorticity, namely frequency dispersion [5]. Without dispersion, in linear theory a disturbance simply undulates, returning to the same shape periodically. Dispersion increase the frequency of shorter waves and thereby creates a competition between the nonlinear steepening of a disturbance and the separation of the small-wavelength components. This effect is demonstrated by two calculations, in figs. 15 and 16, differing only in the respect that the latter begins with an interface thickness four times as wide as the former (note: filamentation for an infinitely thin interface commences at \( t = 13.2 \) [5]). While filamentation does occur in fig. 16, it does so by just overcoming dispersion, seen in the varicose shape of the interface. Basically, filamentation is likely to occur if the rate of nonlinear steepening of the interface, proportional to the square of the interface slope, greatly exceeds the rate of dispersion, proportional to the squared ratio of the interface thickness to disturbance extent (see ref. [5], section 6). This condition is satisfied as long as the amplitude of a disturbance is large compared to the interface width – filamentation does not require infinitely steep vorticity gradients.

Calculations as complicated as those illustrated above put great demands on the computational algorithm. First, filamentation does not occur readily for shallow disturbances, so one must compute, accurately, the undulation and gradual steepening of a disturbance over say ten to twenty revolutions of the vortex just to see the onset of filamentation. This requires a great degree of spatial resolution as well as an algorithm which is sufficiently sensitive to the initially slow growth in the maximum curvature along a contour. For example, in fig. 10, more than 700
Fig. 12. (a) The evolution of a gently disturbed circular vortex. The boundary is shown at multiples of the linear disturbance period, $T_d = 2$ (see ref. [5] for details). Slow nonlinear steepening eventually gives way to filamentation. $\mu = 0.01$, $\delta = 1.25 \times 10^{-5}$ and $\Delta t = 0.05$, $\epsilon_C = 0.0039^\circ$. (b) The continuation of (a) later in the evolution and at shorter intervals of time. Now, the large-scale oscillation of the boundary is evident. The boundary almost periodically ejects new filaments with no sign of abatement.
Fig. 12 (continued).
Fig. 13. Same as in fig. 12 except for a disturbance of half the amplitude. $\mu = 0.0075$, $\delta = 7.03 \times 10^{-6}$ and $\Delta t = 0.05$, $\epsilon_c = 0.019^\circ$. 
Fig. 14. The commencement of filamentation of a periodic vorticity interface, initially of the form $z(\phi, 0) = 0.05 \cos \phi + 0.05 \cos 2\phi$ (the two broadest linear eigenmodes). The vorticity above the interface equals $-2\pi$, while the fluid below is irrotational. A disturbance of half this amplitude also leads to filamentation, but after a time four times as long (not shown). $\mu = 0.01$, $\delta = 1.25 \times 10^{-5}$ and $\Delta t = 0.05$. $\epsilon_c = 0.00032^\circ$ (see appendix for definition).
nodes were used during this initial steepening stage. Once filamentation commences, the curvature at the tip of the first filament soon exceeds the maximum curvature allowed (about 80,000 times the curvature of the undisturbed vortex in this example), well before the formation of the second filament. Some sort of surgery must be used for the calculation to continue. The demand on the computational resources then rises sharply with the emergence of subsequent filaments, not so much as a result of the greater number of filaments, but more because the filaments interact in such a complicated manner with the other filaments and the interface below. Swirling motions ensue (e.g., fig. 10), and the distortion of the interface correspondingly accelerates.

Filamentation therefore qualifies as a stringent benchmark for the testing of advanced algorithms. Fig. 17a shows the contour shape at the onset of filamentation computed at four different resolutions. All four curves compare favorably on the large scales, yet differ visibly on the small scales. The two finest resolution calculations manage to capture two filaments while the coarsest calculations only capture one filament or none at all. Slight differences can be observed between the two finest-resolution calculations just under the two filaments near the point where they are emerging from the interface. These differences augment at later times (see figs. 17b and 17c); however, the large-scale structure of the interface remains very similar. The surgical scale $\delta$ therefore acts as a "regularization" parameter, in sense that larger $\delta$ inhibits the formation of fine-scale structure without altering, at least not immediately, the large-scale structure. Over a sufficiently long time, however, the effect of this regularization accumulates and does eventually alter the large-scale structure. For example, in the elliptical vortex calculation shown in fig. 6, if surgery at a much larger scale had been performed, so as to remove the filament seen at $t = 11$, the large-scale vortex evolution would rapidly depart from that seen in fig. 6 after $t = 13$.

Fig. 15. (a) The evolution of a slightly disturbed circular vortex whose edge consists of four equal jumps in vorticity. The initial disturbance is similar to that used in the calculation shown in fig. 10, except the amplitude $a$ is slightly larger ($a = \sqrt{2}/20$) and the interface is spread out over a thickness $\Delta = a/32$. $\mu = 0.015$, $\delta = 2.81 \times 10^{-5}$ and $\Delta t = 0.05$. $\epsilon_c = 0.013^\circ$. (b) An enlarged view of (a) from $t = 21$ to $t = 24.5$. 

Fig. 15 (continued).
If the fluid evolution is really as unpredictable as it appears to be, how does one measure accuracy? A comparison with the known evolution of steady flows [3, 62], which do not develop small scales, only gives information about an algorithm's ability to follow solely large-scale motions. Steady flows are exceptional. Indeed, it is conjectured that nearly all unsteady flows eventually develop small scales [5]. Long-time calculations must face the problem of incorporating the mean effects of phenomena like filamentation on the large-scale dynamics, or else they cannot be trusted (see section 7 for further comments).

Long-time calculations of unsteady flows may, however, be possible in the case of a quasi-geostrophic fluid. In a single layer of quasi-geostrophic fluid (see section 4.10), there exists a length scale, the Rossby radius of deformation $R_d$, the presence of which greatly weakens the long range interaction between two points in a classical ($R_d \rightarrow \infty$) fluid. (For two points separated by a distance $r$, Green's function is $-(2\pi)^{-1}K_0(r/R_d)$ – see section 4.1.) Polvani [78] and Polvani, Zabusky and Flierl [79] have shown that the coalescence of two identical uniform vortices is strongly inhibited when $R_d$ is small compared to the vortex radius. While the vortices will merge if the distance between their boundaries is sufficiently small, the evolution of the compound vortex is very different from that in a classical fluid. The compound vortex appears to support predominantly large-scale, large-amplitude, non-breaking waves for long periods of time.

The results of a series of high-resolution calculations to investigate the effect of finite $R_d$ on filamentation are discussed next. The initial condition for each calculation is a unit circular vortex disturbed by a small, localized asymmetric bump (in polar coordinates, $r(\theta, 0) = 1 - a(\theta/\theta_0) \exp[-(\theta/\theta_0)^2], -\pi < \theta < \pi; a = 1/40, \theta_0 = \pi/40$). When $R_d = \infty$ (fig. 2 of ref. [5]),

![Diagram](image)

Fig. 16. (a) Same as in fig. 15 except the initial interface is four times as thick, $\Delta = a/8$. Dispersive effects arising from a finite interface spread the disturbance while non-dispersive steepening effects counter and succeed at producing filamentation. An interface twice again as wide greatly retards if not inhibits filamentation [5]. $\epsilon_c = 0.018^\circ$. (b) An enlarged view of (a) from $t = 26.5$ to $t = 29.75$. 
filamentation commences at $t = t_f = 12.6$ or just after six periods of undular motion. As $R_d$ is brought down to finite and successively smaller values, at first filamentation is delayed ($t_f = 12.9$ at $R_d = 2$; $t_f = 15.1$ at $R_d = 1$; $t_f = 23.3$ at $R_d = 1/2$) and then, apparently, altogether prevented for $R_d \lesssim 1/3$ - see fig. 18 for $R_d = 1/3$. As with the multicontour calculations discussed above,
linear frequency dispersion is again responsible for the suppression of filamentation; however, the nature of the dispersion is very different. In the multi-contour case, long waves (long compared to the interface width) all have approximately the same frequency (in a frame rotating with the angular velocity of the undisturbed vortex) while short waves are dispersive; in the quasi-geostrophic case, long waves (long compared to the Rossby radius $R_d$) are dispersive while short waves are nondispersive. In the quasi-geostrophic case, this means that a sufficiently small disturbance of a fixed shape will evolve oblivious to $R_d$ and so evolve into filaments (presumably; the weight of evidence [5] however strongly supports this statement). Furthermore, the lack of dispersion at small scales means that the observed suppression of filamentation in fig. 18 may only be true for a finite span of time; since there is no mechanism to dissipate the waves on the vortex boundary, it is possible that smaller and smaller scales will develop as a result of episodic constructive interference between the waves, and once sufficiently small scales develop, filamentation appears inevitable. It may be true, however, that the intensity of filamentation is weakened considerably by finite $R_d$.

In a classical fluid, thin filaments of vorticity result from a variety of vortex interactions, not just filamentation, and numerical experiments show flow fields strewn with filaments [1,2,9,10]. The filaments appear almost completely dominated by the concentrated centers of vorticity (vortices) seldomly “rolling-up” into strings of miniature vortices as the classic linear stability
Fig. 18. The evolution of a small, shallow disturbance to a circular vortex in a single-layer, quasi-geostrophic setting. Shown is $r(\theta, t)$ versus $\theta$ from $\theta = 0$ (left-hand edge) to $\theta = -\pi/4$ (right-hand edge) in a reference frame rotating with the angular velocity of the undisturbed vortex, $\Omega = \omega I_1(R^{-1})K_1(R^{-1})$; here $\omega = 2\pi$ is the vorticity within the vortex, and $R = 1/3$. $\mu = 0.01$, $\delta = 1.25 \times 10^{-5}$ and $\Delta t = 0.05$. $\epsilon_c = 0.00092^\ominus$. (The calculation continues until $t = 121.5$ without any sign of filamentation.)

analysis of Rayleigh [85] might lead one to believe. The fact is that the vortices exert a great influence over the surrounding sea of filaments because the vortices are almost always straining or shearing the filaments, and the filaments rarely satisfy the conditions of Rayleigh’s analysis.
Fig. 19. A calculation of a linearly unstable strip of vorticity in adverse shear using periodic or cylindrical CS. The initial disturbance is the one which is most unstable in linear theory [6] and has an amplitude 5% of the thickness of the strip. The adverse shear is represented by the addition of the external flow $u_e = \Lambda \omega y$ where $\Lambda = 0.65$, $\omega = 2\pi$ is the vorticity anomaly of the strip, $y$ is the cross-strip coordinate, and the velocity $u_e$ runs perpendicular to $y$. $\mu = 0.03$, $\delta = 1.125 \times 10^{-4}$ and $\Delta t = 0.05$. $\epsilon_C = 0.020^\circ$. 
For a filament caught in a pure strain field (one which simply extends the filament exponentially), it takes a strain rate of only seven percent of the filament's peak vorticity to limit the wave-steepness growth of an arbitrary superposition of linear waves to $e = 2.71828 \ldots$ [8]. Extensional strain typically overwhelms a filament's inherent instability. But filaments do not remain in a pure strain field for long. Vortices sweeping through the sea of filaments capture some of the filaments, and the filaments then begin encircling the vortices. A filament captured in this way becomes progressively more aligned with the circumferential velocity field about the vortex (barring the encroachment of other vortices), and its rate of extension diminishes. Having aligned itself sufficiently with the vortex's velocity field, the filament feels almost no extensional strain, and so might be expected to go unstable. However, the strain field of the vortex is still present, although its principal axis is no longer aligned with the filament locally, and exerts its influence in a different form. A filament whose vorticity has the same sign as that of the vortex feels "adverse shear" (equal to twice the local strain rate) and is stable to linear disturbances if the adverse shear exceeds the peak vorticity of the filament (a result also due to Rayleigh [85]). Nonlinear calculations [6] show that an adverse shear two-thirds as large will do (see fig. 19). On the other hand, a filament whose vorticity is of the opposite sign will tend to destabilize even more rapidly than a filament in isolation. The phenomenon discussed next, however, implies that filaments do not remain around a single vortex for long.

The strain field due to vortices in a fluid not only greatly controls the dynamics of thin filaments, but it also exerts a significant influence on the structure of the vortices themselves. Vortices do not remain dominantly circular entities in the presence of other vortices, rather, vortices react to strain, sensitively under the right conditions, and find themselves often in a highly but manageably distorted state. Nor do vortices simply deform; the strain forcing strips away vorticity from the edge of vortices until the vorticity gradients are sufficiently great to withstand further "stripping" [7], see figs. 20 and 21. In these examples, the distance between the outer contours diminishes considerably over a short interval of time; in fig. 21, for instance, the eight outermost contours collapse to within a distance 6000 times smaller than their initial separation by $t = 10$, and then the remaining vortex pulsates in a nearly periodic fashion without sign of further significant stripping. Filaments of vorticity caught in the vicinity of the stripped vortex do not remain for long as they too are swept in and away by the combined strain field of the vortex and the imposed forcing.

An important finding in this research is that vortices capable of large departures from circularity must have correspondingly steep edges. The same finding also applies to unforced vortices, although such a situation is unlikely to occur or last long in a fluid with many vortices. Nevertheless, recent calculations [86] show that a wide range of disturbed, two-fold symmetric vortex equilibria appear to retain their general form indefinitely (see figs. 22 and 23). For the stable equilibria, the aspect ratio of the vortices correlates directly with the steepness of the vorticity profile; however, beyond a certain critical aspect ratio of about 3 to 1, all these vortex equilibria both look like and destabilize like an elliptical vortex of uniform vorticity (as in fig. 6).

The existence of robust, nonuniform and non-circular vortices is perhaps unexpected in light of the claim that all such vortices will ultimately reduce to circular symmetry (in a coarse-grained sense), a process called "axisymmetrization" [11]. According to Melander et al [11], an initially nonuniform and non-circular vortex spins off filaments of vorticity which thereafter "reconnect" repeatedly, each time dislodging more vorticity from the core and, on average, reducing the
Fig. 20. (a) The interaction between two nonuniform vortices. Each contour represents a jump in vorticity of the amount $\pi/2$ when crossed in the direction leading to the center of each vortex. The radii of the contours within each vortex are $r_j = (j/N)^{1/p}$, $j=1,...,N$, with $N$ being the number of contours (3 and 4 respectively) and $p = 3$. The distance between vortex centers is $3$ initially. $\mu = 0.04$, $\delta = 2 \times 10^{-4}$ and $\Delta t = 0.05$. $c_C = 0.012^\circ$. (b) An enlarged view of the "stripped" smaller vortex in (a) at $t = 10$. 
eccentricity of the core. Their argument, only briefly described here, is wholly inviscid, and applies well to their dissipative, numerical calculations for early times. At later times, dissipation may be hastening the approach to axisymmetry [86]. By contrast, numerous calculations using contour surgery with 8 and 16 contours show no sign of axisymmetrization for long times (up to fifty particle revolution periods $T_p = 4\pi / |\omega|_{\text{max}}$), for initial conditions with more tightly spaced contours than considered by Melander et al. Either axisymmetrization takes an extremely long time, a discrete vorticity distribution is artificial, or dissipation ultimately fuels the process. The first possibility is impossible to verify by numerical means, and there are no analytical proofs of nonlinear stability for noncircular vortex equilibria. The second possibility can, in fact, be true of any discrete numerical method—without an exact equilibrium solution for a smooth profile of vorticity, one cannot be certain that one's numerical solutions are solutions of the Euler equations. In ref. [86], the discretization of the smooth profile used by Melander et al. into 1024 contours suggests however that noncircular, smooth-profiled equilibria do exist, only they have eccentricities much smaller than considered by Melander et al. What appears certain is that equilibrium vortices must have all their vorticity within a finite area (this does not then permit
Fig. 21. The evolution of an initially circular distribution of vorticity in a uniform external shear, $u_s = \gamma(x + y)$, $v_s = -\gamma(x + y)$, with $\gamma = 0.06\omega_{\text{max}}$ and $\omega_{\text{max}} = 2\pi$, as usual. The vorticity jumps by an equal amount across each of the 16 contours, and the contour radii are chosen to mimic the continuous profile $\omega(r) = \pi(1 + \cos \pi r)$ — see ref. [7] for details. $\mu = 0.05$, $\delta = 3.125 \times 10^{-4}$ and $\Delta t = 0.05$. $\epsilon_C = 3.6^\circ$ (see appendix for definition).
Fig. 22. Various near-limiting forms of steadily rotating elliptical-like distributions of vorticity consisting of 16 contours (vorticity jumps). These equilibria, corresponding to the vorticity profile used in ref. [11], are distinguished by the steepness of their gradients or, in these examples, the areas between contours. The aspect ratio of the outer contour is given in the upper left corner of each frame. (See ref. [86] for further details.)

$C^\infty$ vorticity distributions). Of course, one still cannot be sure of nonlinear stability. The third possibility is dissipation. Dissipation is a subtle process in conventional, pseudo-spectral calculations, and its effects depend on both its magnitude and form. Further discussion may be found in refs. [7,86].

Further contour surgery calculations exploring axisymmetrization question whether or not the process goes all the way. Do vortices ultimately become circular in an inviscid fluid? Again, this is a difficult question to answer because of the finite computer resources available, but several lengthy calculations suggest a possible equilibration of the flow at a reduced, but nonzero, eccentricity [86]. An example is given in figs. 24 and 25.

Another line of research concerns the general “survivability” of vortices in the presence of other vortices, approximated by a locally uniform strain field. Strain need not simply strip vortices and scramble filaments, and although sufficiently great strain can turn a vortex into a

Fig. 23. (a) A randomly disturbed equilibrium vortex belonging to the family illustrated in fig. 22 (aspect ratio 4/3). (b) The state of the vortex at $t = 94.5$, or more than 47 particle revolution periods (see text). The initial condition was generated by displacing the x- and y-coordinates of all of the nodes on the equilibrium vortex by a uniformly distributed random number lying between $-10^{-3}$ and $10^{-3}$. $\mu = 0.04$, $\delta = 2 \times 10^{-4}$ and $\Delta t = 0.05$. $\epsilon_C = 0.10^\circ$. 
Fig. 24. (a) The evolution of an initially elliptical distribution of vorticity. The eight contours all have the same aspect ratio \( \lambda = 2 \) and cross the x-axis at the locations \( r_j = (j/8)^{1/p}, \ j = 1, \ldots, 8 \ (p = 4) \). \( \mu = 0.04, \ \delta = 2 \times 10^{-4} \) and \( \Delta t = 0.05, \ \epsilon_c = 2.4 \). (b) The state of the flow in (a) at \( t = 21.5 \). Here, 16,128 nodes are being used to resolve this complex flow, and even this enlarged plot cannot capture all of the detail. Note the persistence of the general two-fold shape of the vortex core (see also (c) and (d)). (c) A shaded image of the flow at \( t = 21.5 \) corresponding to (b). The darkness of the shade is directly proportional to the magnitude of the vorticity. Note how this representation of the flow de-emphasizes the dynamically weak filaments. (d) An enlargement of a part of the flow shown in (c).
thin filament, weaker strain can instead shred vortices into a mass of fine filaments. A recent linear stability analysis backed by nonlinear calculations of Kida's [82] periodic elliptical solutions has uncovered unexpectedly rich instability behavior [84]. Kida's solutions are elliptical vortices of uniform vorticity which remain elliptical, if undisturbed, in the presence of the external strain field (46). In general, the aspect ratio and orientation of the ellipse varies (periodically) with time. If the strain rate $\gamma$ is excessive, however, the ellipse extends indefinitely, even in the absence of disturbances [82–84]. For weaker strain, though, small disturbances to the elliptical shape of the vortex can lead to vigorous instabilities [84]; see, for example, fig. 26. Over time, the incessant strain forcing causes the vortex to give up continually more of its circulation to the surrounding filaments.

Weak strain can also induce vortex merging or coalescence. While simple criteria exist for the merging of two equal vortices, otherwise isolated [12], this simplicity is lost when one takes into account the strain forcing of distant vortices. Essentially, if two vortices are appropriately
positioned, an external strain field like (50) can push the vortices close enough together for coalescence (see fig. 27).

7. Discussion and outlook

Contour surgery offers a viable and in many cases advantageous alternative to traditional Eulerian algorithms for investigating two-dimensional vortex dynamics at very high Reynolds numbers. At present, the dynamic range of typical calculations using CS exceeds the maximum attainable dynamic range for Eulerian methods by one to two orders of magnitude, this increased dynamic range having uncovered the ubiquity of filamentation [5] and the extent to which vortex interactions lead to vorticity-gradient intensification [3,6,7,84,86] in an inviscid fluid.

It appears that the introduction of “surgery” into contour dynamics creates a new form of dissipation. This form of dissipation, however, is quite unlike ordinary dissipation. Surgery does
not affect steep gradients of vorticity whereas ordinary dissipation is most active where the gradients are steepest. Instead, surgery selectively removes thin filaments of fluid, without regard to preserving the global invariance of circulation, for example. Since thin filaments of fluid often behave quasi-passively [6,8], surgery may give a better view of inviscid fluid evolution than one at first sight might believe.

Several outstanding problems remain however. The computational efficiency of CD/CS algorithms hinges on the way in which resolution is distributed along contours. The method devised for the CS algorithm is purely empirical. One would prefer a method which represents, most accurately, the greatest number of degrees of freedom for any given number of nodes. To devise this optimal method, it is first necessary to define what is meant by “accuracy”. Indeed, there may be several possible definitions and different methods associated with each one. Upon arriving at a satisfactory definition, it is still not certain how one would distribute the nodes optimally. It may also be necessary to develop a “regularization” scheme (for instance, ref. [63])
Fig. 25. The effective aspect ratio $\lambda_{\text{eff}}$ versus time for the vortex evolution depicted in fig. 24 (dashed line until $t = 21.5$) and the same computed at coarser resolution (solid line). Here, $\lambda_{\text{eff}}^2 = (J + R)/(J - R)$ with $R^2 = D^2 + S^2$, $D = J_{10} - J_{02}$, $S = 2J_{11}$, $J = J_{10} + J_{02}$ and $J_{n,m} = \int \omega(x, y)x^ny^m \, dx \, dy$ (see appendix and ref. [86]). For a vortex with nested elliptical contours all of the same aspect ratio, $\lambda_{\text{eff}}$ exactly equals this aspect ratio (e.g., $\lambda_{\text{eff}} = 2$ at $t = 0$). The coarser resolution calculation uses $\Delta t = 1/14$, $\mu = 0.05$ and $\delta = 2.81 \times 10^{-4}$ up to $t = 21.5$ and $\Delta t = 1/10$, $\mu = 0.06$ and $\delta = 4.5 \times 10^{-4}$ thenceforth. Even though the coarser resolution calculation is far less accurate than the fine one, $\lambda_{\text{eff}}$ for both calculations are almost indistinguishable over their common time interval (for example, the two curves at $t = 21.5$ agree to within 0.0178%). The close agreement between the two curves gives confidence to the variation of $\lambda_{\text{eff}}$ seen in this plot after $t = 21.5$. $\lambda_{\text{eff}}$ suffers its greatest loss at the onset of axisymmetrization, but it then recovers and proceeds to oscillate, with diminishing amplitude, about a value significantly different from unity.

For further remarks, see ref. [86].

which takes account of the possibly significant, accumulative influence of scales of motion not resolved on those that are. Careful comparisons should be made between fine resolution calculations averaged in some appropriate way and coarse, regularized calculations.

Recent detailed observations and high-resolution numerical experiments of the Earth's stratosphere (see, e.g. ref. [9] and references therein) link many of the fundamental aspects of stratospheric dynamics to those characteristic of an inviscid, one-layer, two-dimensional and incompressible fluid. While the stratosphere is certainly more complicated, there are dynamical sequences shared by both the real and idealized system which merit attention [87]. A central question concerns the structure of the “polar-night vortex”, an intense feature centered close to wintertime pole, and its response to external forcing (e.g., vertically propagating planetary-scale waves generated by the Earth’s topography or by inhomogeneities in solar heating). Calculations with CS have linked the observed sharp vorticity gradients at the edge of the vortex with the effect of weak, large-scale strain forcing (see, for example, fig. 21). There are many other questions about the nearly inviscid, layerwise-two-dimensional, and strongly inhomogeneous nature of the atmosphere that merit a new approach, questions that bear not only upon
Fig. 26. The evolution of a strained, initially randomly disturbed circular vortex (see the caption of fig. 23 for the disturbance shape; here, however, the initial disturbance amplitude is ten times smaller). $\gamma = 0.05\omega$, $\Omega = -0.3\omega$, and $\omega = 2\pi$. $\mu = 0.02$, $\delta = 5 \times 10^{-5}$ and $\Delta t = 0.05$. $\epsilon_c = 0.047^\circ$. Note the big gap of time between the first two frames.
Fig. 27. The evolution of two identical, initially circular vortices in a pure strain field, \(u_x = \gamma x, \quad u_y = -\gamma y, \quad \gamma = 0.04\omega\) and \(\omega = 2\pi. \quad \mu = 0.04, \quad \delta = 2 \times 10^{-4}\) and \(\Delta t = 0.05. \quad \epsilon_C = 0.25^\circ\). The symbol \(\triangle\) marks the positions of the vortices' centroids while \(\triangledown\) marks the positions of point vortices of the same circulation and starting from the same centroid positions as the finite-area vortices. The point-vortex evolution is a useful guide to predicting merger: if the distance between the point vortices diminishes to within 3.4 (finite-area) vortex radii, the finite circular vortices tend to coalesce.
environmental problems such as the ozone hole [88], but also upon the fundamental nature of turbulence. CS and its future extensions (e.g., to slightly compressible flows) may offer such an approach.

Acknowledgement

I wish to thank G.R. Baker, D.D. Holm, M.E. McIntyre and N.J. Zabusky for their invaluable support and stimulating discussions of vortex dynamics and contour dynamics. Particular thanks are due to N.J. Zabusky for his exceedingly helpful comments on this manuscript. The computations were performed on the CYBER 205 at the Manchester Computer Centre and on the CRAY X-MP at Rutherford–Appleton Laboratory, with the support of the UK Science and Engineering Research Council, the US Office of Naval Research, and the Natural Environmental Research Council’s UK Universities’ Global Atmospheric Modelling Project.

Appendix. The global invariants of two-dimensional flows

One measure of error in CS calculations makes use of quantities that are conserved in true inviscid fluid evolution but are not in the algorithm. This appendix lists these quantities and their associated contour-integral expressions for the various flow geometries consider in section 5. In addition, the “phase error”, a recommended measure of the accumulative error, is defined.

A.1. Flow on an infinite planar surface

The independent global invariants for a piecewise-constant vorticity distribution \( \omega(x, y) \) include the area enclosed by each contour \( A_k \), the first moments of the vorticity distribution \( J_{10} \) and \( J_{01} \), the angular impulse \( J = J_{20} + J_{02} \), and the “excess energy” \( E \) [31,50]. Here,

\[
A_k = \int \int_{\varphi_k} dx \, dy
\]

\[
= \frac{1}{2} \int_{\varphi_k} (x_k \, dy_k - y_k \, dx_k), \quad (A.1a)
\]

\[
J_{mn} = \int \int \omega(x, y) x^m y^n \, dx \, dy
\]

\[
= \frac{1}{m+n+2} \sum_k \tilde{\omega}_k \int_{\varphi_k} x_k^m y_k^n (x_k \, dy_k - y_k \, dx_k). \quad (A.2a)
\]
and

\[
E = \lim_{L \to \infty} \left[ \frac{1}{2} \int \int_{|x| < L} \left( u^2 + v^2 \right) \, dx \, dy - \frac{r^2}{4\pi} \log\left( \frac{L}{l} \right) \right]
\]

\[
= \frac{r^2}{4\pi} + \frac{1}{16\pi} \sum \sum_{j} \sum_{k} \omega_j \omega_k \int_{\varphi_j} \int_{\varphi_k} |x'_j - x_k|^2 \log\left( \frac{|x'_j - x_k|}{l} \right) \, dx'_j \, dx_k,
\]

(A.3a)

where \( \omega_k \) is the jump in vorticity crossing the contour \( \varphi_k \) inwards, \( x_k \) is a point on \( \varphi_k \), and, in the expression for \( E \), \( \Gamma = J_{00} = \sum_k \omega_k A_k \) is the total circulation, and \( l \) is a fixed length scale characteristic of the vorticity distribution. The choice of \( l \) is a partly a matter of taste; the recommended choice in ref. [31] is \( l = \sqrt{2J/J} \), but a perhaps better choice, one that is well defined for vanishing circulation, is

\[
\pi l^2 = \frac{1}{\omega_{\text{max}}} \sum \omega_k A_k.
\]

(A.4)

The derivation of the contour-integral form of \( E \) is described in the appendix of ref. [50].

In numerical calculations, one particular combination of global invariants, \( C = \Gamma^2/4\pi J \), a frequency-like quantity, is used to monitor the accumulative growth in error during a calculation, or the “phase error”. This phase error is defined as

\[
\epsilon_C = \Delta t \sum_{t=1}^{m} |C(t) - C(0)|,
\]

(A.5)

where \( t = t_m = m \Delta t \) is the duration of the calculation (\( \Delta t \) is the time step). For some flows, \( J \) could vanish, in which case \( C = \Gamma^2/4\pi J \) would not be an appropriate choice for \( \epsilon_C \). Under these circumstances, a more suitable choice is \( C = \Gamma/\pi l^2 \), with \( l^2 \) defined in (A.4).

In the presence of an external straining flow (see (50)), the number of invariants is reduced, and, in particular, the second moment \( J \) is no longer conserved. The only remaining invariants are the areas enclosed by each contour, \( A_k \), and a modified version of the excess energy, \( E' = E + \gamma J_{II} - \frac{1}{2} \Omega J \) [84]. Both invariants are computed using the anomalous vorticity distribution \( \omega_a = \omega - 2\Omega \) in place of \( \omega \). In calculations, error is monitored by (A.5) using \( C = \Gamma/\pi l^2 \).

A.2. Flow on the surface of a unit sphere

On the sphere, the global invariants include the area \( A_k \) enclosed by each contour \( \varphi_k \), the first moments \( J_{100}, J_{010}, \) and \( J_{001} \), and the energy \( E \). With \( d\Omega \) representing an incremental area element,

\[
A_k = \int \int_{x \in \varphi_k} d\Omega
\]

\[
= \int_{\varphi_k} \frac{x_k \, dy_k - y_k \, dx_k}{1 + z_k},
\]

(A.6a)

(A.6b)
\( J_{001} = \int \int \omega(x) z \, d\Omega \)  

(A.7a)

\[ = \frac{1}{2} \sum_k \tilde{\omega}_k \int_{\phi_k} (x_k \, d\phi_k - y_k \, dx_k), \]  

(A.7b)

\[ E = \frac{1}{2} \int \int (u^2 + v^2 + w^2) \, d\Omega, \]  

(A.8)

where \( J_{010} \) and \( J_{100} \) follow from cyclic permutation of \( x, y, \) and \( z \) in \( J_{001} \). Note that the total circulation, \( J_{000} \), must vanish as it is simply the area integral of Laplace’s equation on a closed surface. The energy cannot be expressed in a simple contour integral formula like (A.3b); here, one cannot simply substitute three-dimensional vectors for two-dimensional ones.

In numerical calculations, the phase error (see (A.5)) is computed with \( C = (2\pi)^{-1} \Sigma_k |\tilde{\omega}_k A_k|, \) a frequency-like quantity by virtue of the fact that the sphere’s radius is unity.

### A.3. Flow on the surface of a cylinder or periodic flow

Periodicity destroys the invariance of some of the global invariants for infinite planar flow, and those that remain are the area \( A_k \) of each contour, the linear impulse \( J_{01} \), and the “excess energy” \( E \) [29]. Let \( z \) be the coordinate along the axis of the cylinder and \( \phi \) that around it. For “irreducible” contours, i.e. those for which \( \int_{\phi_k} d\phi_k \neq 0 \), the area is measured with respect to the plane \( z = 0 \) and, even for reducible contours, the area may be defined

\[ A_k = -\int_{\phi_k} z_k \, d\phi_k. \]  

(A.9)

Similar considerations apply to the definition of the linear impulse

\[ J_{01} = -\frac{1}{2} \sum_k \tilde{\omega}_k \int_{\phi_k} z_k^2 \, d\phi_k. \]  

(A.10)

The remaining conserved quantity, the “excess energy” satisfies

\[ E = -\frac{1}{2} \int \int \omega(\phi, z) \psi(\phi, z) \, d\phi \, dz, \]  

(A.11)

and does not appear to admit a simple contour integral expression like (A.3b) (see ref. [29] for further remarks).

In numerical calculations, the phase error is defined just as in the spherical case above, i.e. (A.5) with \( C = (2\pi)^{-1} \Sigma_k |\tilde{\omega}_k A_k| \).

### A.4. Quasi-geostrophic flow

For flow in \( N \) layers, the area of each contour \( A_{kn} \) in each layer \( n \) is separately conserved. Other global invariants include the density weighted moments \( J_{10}, J_{01}, J = J_{20} + J_{02}, \) and energy
The moments have essentially the same definitions as before, the only new feature being the sum over the $N$ layers; for example,

$$J_{10} = \sum_{n=1}^{N} \rho_n D_n \int \omega_n(x, y) x \, dx \, dy,$$

(A.12)

where $\rho_n$ is the uniform density of the $n$th layer, $D_n$ is the vertical thickness of the layer, and $\omega_n$ is the vorticity of a point $(x, y)$ in the layer.

The energy consists of both a kinetic part $T$ and a potential part $V$. The kinetic part is a simple extension of the single infinite planar expression, namely

$$T = \frac{1}{2} \sum_{n=1}^{N} \rho_n D_n \int \left( u_n^2 + v_n^2 \right) \, dx \, dy,$$

(A.13)

where the limit in (A.3a) is implicit in (A.13). The potential part arises from the departures of the interfaces between layers from flat surfaces. Denoting $\eta_n$ as the departure of the interface between the $n$th and $n+1$st layers, quasi-geostrophic theory implies

$$\eta_n = \frac{f_0}{g \Delta \rho} \left( \rho_n \psi_n - \rho_{n-1} \psi_{n-1} \right),$$

(A.14)

where, as in section 4.10, $f_0$ is the Coriolis parameter, $g$ is the acceleration due to gravity, $\Delta \rho$ is the density difference between layers (equal for all layers), and $\psi_n$ is the streamfunction in the $n$th layer. It is to be understood that the term $\rho_{n-1} \psi_{n-1}$ is absent for $n = 1$. Also, the upper and lower interfaces of the entire fluid are flat and rigid (implying $\eta_N = 0$). One is at liberty to choose $\int \eta_n \, dx \, dy = 0$ for each interface, conservation of mass ensuring that this remains true for all time. In this case, the potential energy, omitting the constant, unavailable part due to the undisturbed density distribution, is simply

$$V = \frac{1}{2} g \Delta \rho \sum_{n=1}^{N-1} \int \int \eta_n^2 \, dx \, dy.$$

(A.15)

The phase error has the same definition as in the single infinite planar case, namely (A.5) with $C = \Gamma^2/4 \pi J$, $\Gamma = J_{00}$, and $J = J_{20} + J_{02}$.

References


