JOURNAL OF COMPUTATIONAL PHYSICS 39, 305–328 (1981)

# Simulation of Three-Dimensional Incompressible Flows with a Vortex-in-Cell Method

BENOIT COUËT AND OSCAR BUNEMAN

*Institute for Plasma Research, Stanford University, Stanford, California 94305*

AND

ANTHONY LEONARD

*NASA–Ames Research Center, Moffett Field, California 94035*

Received January 3, 1979; revised January 21, 1980

A new method for the numerical simulation of three-dimensional incompressible flows is described. Our vortex-in-cell (VIC) method traces the motion of the vortex filaments in the velocity field which these filaments create. The velocity field is not calculated directly by the Biot–Savart law of interaction but by creating a mesh-record of the vorticity field, then integrating a Poisson's equation via the fast Fourier transform to generate a mesh-record of the velocity field. The computed scales of motion are assumed to be essentially inviscid. Viscous of subgrid-scale effects are incorporated into a filtering procedure in wave vector space. Results of tracing a periodic array of single vortex rings are compared with a Green's function calculation. The agreement is very good.

## 1. INTRODUCTION

In this paper, we describe a new method for the numerical simulation of three-dimensional incompressible flows. Our approach differs from other numerical fluid simulations in that, rather than solving the Navier–Stokes equations on an Eulerian mesh, we emphasize the vortical part of the flow by solving the vorticity equation, using a hybrid method. Making use of the fact that vortices naturally preserve their identity, we follow vortex filaments in a Lagrangian frame. An Eulerian mesh is used only to compute the velocity field required to move the filaments.

Historically, the first numerical calculation using a two-dimensional discrete vortex element method was made by Rosenhead [1], using a few vortices. Since then, the same method has been applied to various two-dimensional shear flows. This work is summarized in a literature survey by Fink and Soh [2]. Most of these efforts were directed toward understanding the time evolution of finite-area vorticity regions or the initial break-up of the laminar shear layer (usually done for short times or small regions with periodic boundary conditions). Some of these recent computer calculations involved thousands of vortices [3, 4, 5]. Other important flows were also



0021-9991/81/020305-24$02.00/0

considered: two-dimensional turbulence [6–9], separated flows [10], and stratified flows in both homogeneous and porous media [11, 12].

A flow field can be represented to any required precision by a sufficiently large assembly of discrete vortices and the time evolution of the field can be transcribed into a kind of particle mechanics of these vortices. If one integrates analytically, in two or three dimensions, the Poisson's equation relating the velocity field to the vorticity field [see Eq. (2.4)], the result is a direct Biot–Savart integration or summation over all the vortices making the vorticity field [13]. With $N$ vortices, a time step in a direct summation scheme will involve evaluating $N-1$ terms for the velocity which displaces a single vortex, and thus on the order of $N \times N$ terms per time step. For large $N$, such a code will be very time consuming. Thus, the two-dimensional mixing layer calculation of Ashurst [7], where the number of vortices increased during the calculation from 1 to 800, required 250 hours of computing time on a CDC 6600. Understandably, previous direct summation calculations were much more modest. Michalke [14] studied the linear and partly nonlinear instability of 72 vortices arranged on three close parallel lines. Acton [15], still in two dimensions, studied the same problem, but with 96 vortices arranged initially on four close sinusoids, and ran for a longer time.

An alternative method for time-stepping a set of vortices in two dimensions, the cloud-in-cell (CIC) method as it is sometimes called, was described by Christiansen [16, 5, 17]. In this algorithm, the basic variables are still the positions and strengths of the vortices, but now a grid is laid down in the plane perpendicular to the vortex filaments and covering the flow area. At every time step, a grid vorticity is generated by distributing the vorticity from each vortex over the four neighboring grid points suitably weighted. This grid vorticity is then used to generate a grid stream function by solving Poisson's equation. The stream functions is differenced on the grid to produce a velocity field, which is finally interpolated back to the vortex positions. The advantage of this apparently rather elaborate detour is that the Poisson inversion can be accomplished by Fast Fourier transform techniques. If the grid is $M \times M$, this is an order $M^2 \log_2 M$ calculation. In a typical $2D$ calculation, the number of grid squares $M \times M$ is of the same order of the total number of vortices $N$ and thus the FFT requires on the order of $N \log_2 N$ operations per time step. The smoothing and interpolations take of the order $N$ operations. When $N$ is large, $N \log_2 N$ is considerably smaller than $N \times N$.

A disadvantage of the could-in-cell method (as against the direct interaction method) is having to account for grid effects. To counteract unwanted grid effects, Wang [18] did some extensive two-dimensional simulations in which he improved on the CIC method by first, using cubic splines for interpolation (i.e., referring to the nearest 16 grid points for each vortex node rather than to the nearest 4) and second, applying a Gaussian shape factor or "filter" in wave vector space. The filter provides each vortex with a finite core and distributes vorticity within the core with cylindrical symmetry. The insensitivity of the resulting potential contours or flow lines to the underlying grid is shown in some diagrams presented in [18].

In three dimensions, it is natural to represent the vorticity field as a collection of

vortex filaments, each one following a three-dimensional space curve. The direct summation scheme has been applied to a small number of simple vortex filaments [19–22], but at considerable cost because of the time required to sum all the mutual Biot–Savart interactions between the many elements in all the filaments. In our method, the velocity field is not calculated directly by the Biot–Savart law of interaction but by creating a mesh-record of the vorticity field, then integrating a Poisson's equation to generate a mesh-record of the velocity field. As in two-dimensional problems, the "cell" or "mesh" method speeds up the calculations of the interactions and allows the three-dimensional vortex tracing method to be applied to a space densely filled with vortex filaments, each filament being resolved in fine detail along its length. Filtering in wave vector space is used to provide each vortex filament with a finite core thereby minimizing unwanted grid effects.

An outline of the remaining sections of the paper is as follows: The fundamentals of vorticity dynamics relevant to our technique are discussed in Section 2 and a description of the computational method is given in Section 3. In Section 4, the results of computational experiments involving vortex rings are presented and discussed while conclusions are given in Section 5.

## 2. Basic Principles

We assume unbounded incompressible flow, fully periodic in each of the three dimensions. Starting with the incompressible Navier–Stokes equations where we assume no external forces, the equation of motion of the velocity field $\mathbf{u}$ is given by:

$$\frac{D\mathbf{u}}{Dt} = -\frac{1}{\rho}\nabla p + \nu\nabla^2\mathbf{u}, \tag{2.1}$$

where $p$, $\rho$ and $\nu$ are respectively the pressure, density and viscosity and where the "material acceleration" is defined: $D\mathbf{u}/Dt \equiv (\partial\mathbf{u}/\partial t) + \mathbf{u}\cdot\nabla\mathbf{u}$. Rather than solving (2.1) on an Eulerian mesh, we want to trace the motion of the vortex filaments in the velocity field which these filaments create. The collection of vortex filaments in each periodic box forms the vorticity field, $\boldsymbol{\omega} = \nabla\times\mathbf{u}$. We therefore solve the incompressible vorticity equation, obtained by taking the curl of (2.1):

$$\frac{D\boldsymbol{\omega}}{Dt} = \boldsymbol{\omega}\cdot\nabla\mathbf{u} + \nu\nabla^2\boldsymbol{\omega}. \tag{2.2}$$

By using the equation of continuity,

$$\nabla\cdot\mathbf{u} = 0, \tag{2.3}$$

we find that the velocity field can be determined kinematically from

$$\nabla^2\mathbf{u} = -\nabla\times\boldsymbol{\omega}. \tag{2.4}$$

In the following, we use the large-eddy simulation approach [23] where one calculates directly the large-scale motions and models approximately the effects of the finer subgrid scales. In the present method, the computed scales of motion are assumed to be essentially inviscid. Any actual viscous effects are on a subgrid scale and are incorporated into the filtering procedure described in more detail in the next section.

From Helmholtz's theorem, the motion is purely kinematic and the vortex filaments follow material lines. Also, by definition of $\omega$, $\nabla \cdot \omega = 0$, and Kelvin's theorem states that, in an ideal fluid, the velocity circulation $\Gamma$ around a closed "fluid" contour is constant in time, that is $D\Gamma/Dt = 0$ where

$$\Gamma = \oint \mathbf{u} \cdot d\mathbf{l} = \int_A \int \omega \cdot d\mathbf{A}. \tag{2.5}$$

Here $A$ is the cross-section area of the filament. In particular, the effective vorticity field $\omega$ in each periodic box representing the large scale motions is taken to be:

$$\omega(\mathbf{r}, t) = \iiint G(\mathbf{r} - \mathbf{r}')\tilde{\omega}(\mathbf{r}', t)\, d\mathbf{r}', \tag{2.6}$$

where $G$ is a filter function and the unfiltered vorticity $\tilde{\omega}$ is generated by the space curve $\mathbf{r}_i(\xi, t)$ as follows,

$$\tilde{\omega}(\mathbf{r}, t) = \sum_i \Gamma_i \oint \delta(\mathbf{r} - \mathbf{r}_i(\xi, t)) \frac{\partial \mathbf{r}_i}{\partial \xi} d\xi \tag{2.7}$$

and is highly singular. Here $\xi$ is a parameter which traces each filament along its length at any instant in time. The summation is over individual vortex filaments. The representation of the vorticity by (2.6) and (2.7) is a natural generalization of representations used in two-dimensional vortex methods. Using (2.7) above is the analog of the point vortex representation; combined with (2.6), we have the analog of the representation by vortices with finite cores [31]. The evolution of each space curve is determined from the continuous velocity field by

$$\frac{\partial \mathbf{r}_i(\xi, t)}{\partial t} = \iiint G(\mathbf{r}_i - \mathbf{r}')\, \mathbf{u}(r', t)\, d\mathbf{r}' \tag{2.8}$$

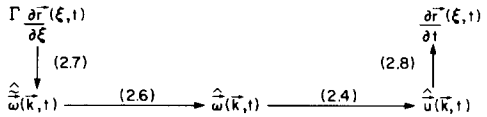
with $\mathbf{u}$ determined from (2.4).

In summary, equations (2.6), (2.7), (2.8), and (2.4) describe the physics of vortical flow which must be discretized for solution on the computer.

## 3. Description of Computational Method

For a variety of reasons, the vorticity field and other fields are conveniently expressed in Fourier space, just as in the more successful numerical attacks on the turbulence problem by solution of the Navier–Stokes equation [24–27]. The main reason field components are recorded in spectral form is that calculus (differentiation and integration) translates to algebra (multiplication and division) in the spectral domain. Of course, this implies periodicity in all three dimensions [38].

Working in Fourier space also provides additional benefit from the control one obtains over the filtering operation: The convolution integral in (2.6) becomes a product in Fourier space. Translating in Fourier space the basic principles given in the previous section, we can draw a computational chain of operations. Starting with a given set of vortex lines with their circulation $\Gamma$, we want to find their displacement:




The numbers refer to the equations of the previous section or their equivalent translated to Fourier space representation. The different parts of the scheme will now be explained, namely the numerical modeling of the vortex filaments, the interpolations that are required in using the fast Fourier transform, the shaping of the vortices and the time-stepping procedure.

1. *Filament Modeling*

In our model, one describes each vortex filament by a succession of closely spaced markers. Considering a single vortex in (2.7), we have

$$\tilde{\omega}(\mathbf{r}) = \Gamma \oint \delta(\mathbf{r} - \mathbf{r}(\xi)) \frac{\partial \mathbf{r}}{\partial \xi} d\xi$$

at an instant $t$, where $\Gamma$ is given by (2.5). Taking the Fourier transform, we obtain

$$\hat{\tilde{\omega}}(\mathbf{k}) = \iiint e^{i\mathbf{k}\cdot\mathbf{r}} \Gamma \oint \delta(\mathbf{r} - \mathbf{r}(\xi)) \frac{\partial \mathbf{r}}{\partial \xi} d\xi \, d\mathbf{r}.$$

If we now discretize $\mathbf{r}$ into piece-wise linear sections,

$$\mathbf{r}(\xi)_{j,j-1} = \xi \mathbf{r}_j + (1 - \xi)\mathbf{r}_{j-1}, \qquad 0 \leqslant \xi \leqslant 1,$$

where the error is $O(|\mathbf{r}_j - \mathbf{r}_{j-1}|^2)$, then

$$\hat{\tilde{\omega}}(\mathbf{k}) = \sum_{j=1}^{m} \Gamma \int_0^1 (\mathbf{r}_j - \mathbf{r}_{j-1}) e^{-i\mathbf{k}\cdot\mathbf{r}(\xi)} j, j-1 \, d\xi, \tag{3.1}$$

where $m$ is the total number of markers describing the filament and $r_m = r_0$. Integrating (3.1) and letting $\mathbf{k} \cdot [(\mathbf{r}_j - \mathbf{r}_{j-1})/2] = \varepsilon_j$, we obtain:

$$\hat{\boldsymbol{\omega}}(\mathbf{k}) = \Gamma \sum_{j=1}^{m} (\mathbf{r}_j - \mathbf{r}_{j-1}) \exp(-i\mathbf{k} \cdot \tfrac{1}{2}(\mathbf{r}_j + \mathbf{r}_{j-1})) \frac{\sin \varepsilon_j}{\varepsilon_j} \tag{3.2}$$

The labor of evaluating the trigonometric functions for all sections of the filament in (3.2) to obtain one Fourier component $\hat{\boldsymbol{\omega}}(\mathbf{k})$ would be prohibitive if each vortex required at each time step the evaluation of all the Fourier harmonics $\mathbf{k}$. It is more efficient to first distribute vorticity onto a grid according to an interpolation process and then perform an FFT. Considering Eq. (3.2), we see that this procedure requires two steps. First, we need to express $\sin \varepsilon_j/\varepsilon_j$ as a linear combination of $e^{-i\mathbf{k}\cdot\mathbf{f}}$ where $\mathbf{f}$ is a function of $\mathbf{r}_j$ and $\mathbf{r}_{j-1}$. Secondly, the resulting trigonometric exponential functions should be approximated onto the mesh in Fourier space so that the FFT can be performed. To obtain a suitable approximation to $\sin \varepsilon_j/\varepsilon_j$, we simply evaluate the integral in Eq. (3.1) by Gaussian quadrature [28] to obtain,

$$\hat{\boldsymbol{\omega}}(\mathbf{k}) = \Gamma \sum_{j=1}^{m} \frac{(\mathbf{r}_j - \mathbf{r}_{j-1})}{2} \times \{\exp(-i\mathbf{k} \cdot \tfrac{1}{2}[(1 + 3^{-1/2})\,\mathbf{r}_j + (1 - 3^{-1/2})\,\mathbf{r}_{j-1}]) + \exp(-i\mathbf{k} \cdot \tfrac{1}{2}[(1 - 3^{-1/2})\,\mathbf{r}_j + (1 + 3^{-1/2})\,\mathbf{r}_{j-1}])\}.$$

Indeed this is equivalent to the approximation,

$$\frac{\sin \varepsilon_j}{\varepsilon_j} \simeq \frac{e^{-i\epsilon_j 3^{-1/2}} + e^{i\epsilon_j 3^{-1/2}}}{2} = \cos\left(\frac{\varepsilon_j}{3^{1/2}}\right)$$

(See the Appendix for derivation.) We now need to replace the pure harmonic $e^{-i\mathbf{k}\cdot\mathbf{r}}$ for arbitrary $\mathbf{r}$ by an approximant to be evaluated on the discrete spatial mesh.

2. *Interpolation*

In each of the three dimensions, we use quadratic spline interpolation to approximate $e^{ikx}$ in terms of $e^{ikn}$. In particular, $e^{ikx}$ is represented in the general interval $n - \frac{1}{2} \leqslant x \leqslant n + \frac{1}{2}$ as the superposition of three parabolic arcs as follows:

$$e^{ikx} \simeq S(k)[\tfrac{1}{2}(x - n + \tfrac{1}{2})^2\, e^{ik(n+1)} + (\tfrac{3}{4} - (x - n)^2)\, e^{ikn} + \tfrac{1}{2}(n + \tfrac{1}{2} - x)^2\, e^{ik(n-1)}]$$

or (3.3)

$$e^{ikx} \simeq S(k)\left[1 - \frac{(1 + 4x^2)}{2} \sin^2 \frac{k}{2} + ix \sin k\right].$$

Note that the choice $4/(3 + \cos k)$ for $S(k)$ would force agreement between function and approximant at $x = 0$. But since we know the function to be approximated throughout the interval in principle from its Fourier harmonics, we can make a less biased choice of $S(k)$. We can minimize the mean square error over the interval by

choice of $S(k)$. Using standard procedures, one finds that $S(k)\,P(x)$ deviates from $e^{ikx}$ with the least mean square error when

$$S(k) = \int_{-1/2}^{1/2} P(x)\, e^{-ikx}\, dx \Big/ \int_{-1/2}^{1/2} P^2(x)\, dx \tag{3.4}$$

and the mean square error is then

$$1 - \left(\int_{-1/2}^{1/2} P(x)\, e^{-ikx}\, dx\right)^2 \Big/ \int_{-1/2}^{1/2} P^2(x)\, dx.$$

Here $P(x)$ is the polynomial in brackets in (3.3). With our new choice of $S(k)$, we can define the second-order spline as that second-order polynomial which joins smoothly (up to the first derivative continuous) with the corresponding polynomial in the adjacent intervals and which departs from the given function with the least mean square error.

Equation (3.4) can be evaluated to yield

$$S(k) = \left(\frac{2}{k}\sin\frac{k}{2}\right)^3 \Big/ \left(1 - \sin^2\frac{k}{2} + \frac{2}{15}\sin^4\frac{k}{2}\right).$$

The expression for the mean square error now becomes

$$1 - \left(\frac{2}{k}\sin\frac{k}{2}\right)^6 \Big/ \left(1 - \sin^2\frac{k}{2} + \frac{2}{15}\sin^4\frac{k}{2}\right).$$

The square root of the mean square error is plotted versus $k$ in Fig. 1. One finds that for small $k$, the rms error is of order $k^3$, specifically the rms error is

$$\frac{k^3}{12(210)^{1/2}} \simeq .0058\, k^3.$$

As mentioned previously, the purpose of interpolation is, in fact, to distribute vorticity onto a grid. Figure 2 illustrates a one-dimensional model where three quadratic spline distributions of vorticity with different total amplitudes are shown to have their centers located at positions $a$, $b$, and $c$. It also shows that three nearest grid points to each of the centers will share the vorticity distribution according to the spline function weighting on them. In our example, grid points $N-2$ to $N$ will share the vortex "$a$" with grid point $N-1$ receiving the largest weight. Similarly points $N-1$ to $N+2$ will share the vortices "$b$" and "$c$." It is the array of this vorticity distribution that will be transformed to calculate the potential array and velocity field array. The quadratic spline weighting is superior to the zero-order weighting (NGP model) and first-order weighting (CIC model) in the sense of creating less field noise and resulting in smoother simulation functions [29, 30]. This is an obvious conclusion since vorticity is now distributed among three grid points instead of one

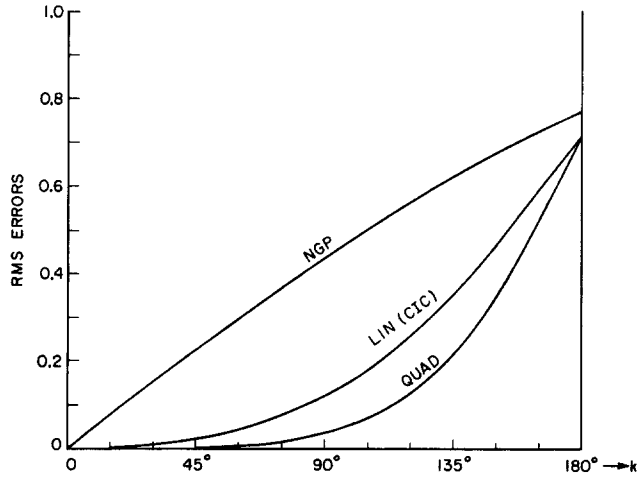



FIG. 1. Square root of the mean square error versus $k$ for nearest grid point, linear and quadratic interpolation of a pure harmonic.

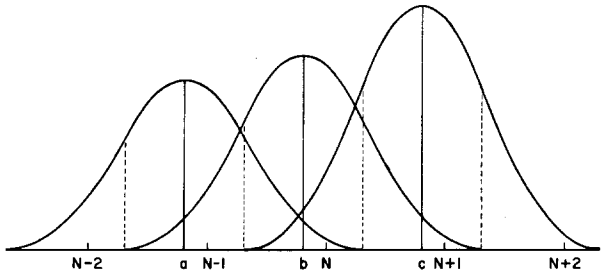



FIG. 2. Quadratic spline vorticity distribution of three typical vortices on their nearby grid points.

or two as in the other models and the interpolated distribution is quadratic rather than a piece-wise step function, or first-order linear function, with discontinuous derivatives. There is also a reduction of aliasing. In three dimensions, the three nearest grid points in each dimension (27 in all) will share the vorticity distribution according to the spline function weighting on them.

### 3. *Shape Factor*

The choice of the shape factor or filter function $G(\mathbf{r})$ [Eq. (2.6)], which gives the vorticity distribution within the core of each filament, is dictated by several considerations.

Firstly, it is intuitively obvious that low $|\mathbf{k}|$ harmonics are interpolated by a certain tabulation mesh better than high $|\mathbf{k}|$ harmonics. Aliasing sets a limit at $k_{\max} = \pi/\Delta$ for each component of $\mathbf{k}$ ($\Delta \equiv$ mesh spacing): any harmonic with a $k$-component higher than this will be misinterpreted by the interpolator as a corresponding lower harmonic with all $k$ components lying within the interval $(-\pi/\Delta, \pi/\Delta)$. In effect, the

interpolator will add a mixture of overtones to the approximation of a pure harmonic. A reduction of aliasing was already achieved by going to a higher-order interpolation than linear. Another way to suppress aliases is to deemphasize harmonics in the range near $k_{max}$ by filtering them out.

Secondly, there are good reasons for introducing shapes of the interacting elements even when no interpolation is used at all. It was pointed out [31] that, because high induced velocities occur when two vortices come close together, unless a finite core radius is used, the accuracy of the discrete vortex element does not improve as more and more elementary vortices are employed to represent a given vorticity field. Also, it is well known that nonlinear terms produce a cascade to higher harmonics that cannot be represented by the mesh. One way to suppress these harmonics is to use a cut-off or shape factor.

Finally, in our simulation so far, we use a cubical mesh system, that is $\Delta x = \Delta y = \Delta z = 1$. Therefore, we note that the shape factor $G(\mathbf{r})$ should be isotropic in space: It should know no coordinate axes. Hence we choose $\hat{G} = \hat{G}(|\mathbf{k}|)$ to be isotropic in $\mathbf{k}$-space, i.e., a function only of $|\mathbf{k}|$. (The compensating adjustment factors $S^2(k)$, on the contrary, must be functions of the components of $\mathbf{k}$.) Since one has an a priori freedom of choice as regards $G(\mathbf{r})$ or $\hat{G}(|\mathbf{k}|)$, one is at liberty to tailor $\hat{G}|\mathbf{k}|)$ so that it deemphasizes the poorly interpolated (badly aliased) harmonics near $|\mathbf{k}| = \pi$. A sharp cutoff in Fourier space would be undesirable (no matter how perfectly each harmonic is evaluated) because it surrounds the objects that interact via the field with lobes in real space. The lobes decay only weakly with distance, like $\Delta$ over the distance. If, instead, the spectrum is brought to zero more smoothly, say at least parabolically, such lobes become attenuated more strongly.

In light of the above considerations, a Gaussian profile for $\hat{G}(|\mathbf{k}|)$ seems a reasonable choice. In addition, Tung and Ting [32] and Saffman [33] found that the distribution of vorticity across the core of a viscous vortex ring with small cross section is Gaussian. Thus, we choose an approximately Gaussian profile, equivalent to a Gaussian shape in real space, but brought to a strict zero at some maximum $|\mathbf{k}|$.

To summarize the interpolation and shaping operations, we start with a vorticity field having singularities on each filament, Eq. (2.7). The use of a grid with spacing $\Delta$ and Fourier transform methods eliminate the singularities since only harmonic numbers up to $\pi/\Delta$ are recognized and the vortices are thus automatically broadened by what is approximately equivalent to a rectangular filter in Fourier space. In addition, further shaping of the vorticity distribution is achieved by explicit application of another filter whose representation in Fourier space is $\hat{G}(|\mathbf{k}|)$. The effective vorticity field $\boldsymbol{\omega}$ is then represented by a filtering operation on the singular $\tilde{\boldsymbol{\omega}}$ as shown by (2.6) where $G$ now indicates the transverse profile of the vortex filaments. We choose $\hat{G}(|\mathbf{k}|)$ to be almost Gaussian (coming to zero smoothly before the spectrum would cut off more drastically), and then each filament acquires a transverse profile in real space that is nearly Gaussian. The shape factor $G$ is applied again when one calculates the evolution of each space curve in time from the continuous velocity field $\mathbf{u}(\mathbf{r}, t)$, Eq. (2.8), with $\mathbf{u}$ determined from (2.4). In practice, this means applying the square of the transform of $G$ as a filter in Fourier space. The

use of the filter $\hat{G}$ provides damping of the high wavenumber components of the field, damping that would otherwise occur through subgrid scale dissipation or by viscous dissipation [23].

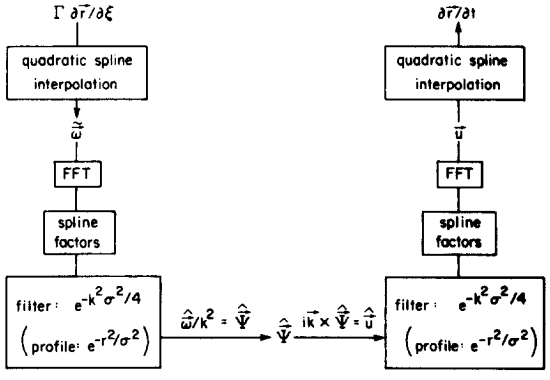
4. *Solving*

If we now consider Eq. (2.4) in Fourier space, the velocity field at the location of a "vortex-marker" is obtained by weighting the entries in the table of spline amplitudes with the spline weights. The latter are deduced from the relative position of a vortex in its interpolation cell. The spline amplitudes are obtained from the velocity harmonics by first multiplying with a factor $S(k)$ and two similar factors which have

The same factors appear again when the displacement of the $k$th vorticity harmonic is calculated. The interpolation can be done for the sum of all the harmonics, and the table into which one interpolates consists of the FFTs of the harmonics of vorticity, modified by the factors ensuring best mean square fit in each dimension.

In going from the table of spline amplitudes for vorticity to the table of spline amplitudes for velocity harmonics, one therefore has not only to perform a forward and backward FFT, with Eq. (2.4) in Fourier space in between, but one must also introduce the square of the spline fitting factors indicated previously.

Similarly, we mentioned that any vorticity shaping factor should be introduced both when the local velocity field action on the distributed vorticity cloud is evaluated and when its excitation of the vorticity harmonics is accumulated. In both cases, one could perform a convolution in real space, but it is much quicker to replace this by a multiplication in Fourier space. The square of the transform of the shape factor is therefore introduced along with the above mentioned spline fitting factors in the course of solving the equations for the velocity field in Fourier space. It is convenient to introduce the (squared) shape factor along with the inverse Poisson operator $1/|\mathbf{k}|^2$. In our present $16^3$ code there are only 64 possible different values of $|\mathbf{k}|^2$ in the sphere $|\mathbf{k}| \leqslant k_{\max}$, so any function of $|\mathbf{k}|$ can readily be pretabulated [39]. The schematic description given below summarizes the algorithm.



### 5. *Time-Stepping*

So far in the code, we have been using the well known leap-frog method with the first step generated by Euler's method. This method is unstable [34, 35] but was used mainly to test our code. In fact, we already noted the instability of this scheme in our two-ring experiments.

Nevertheless, used with an occasional forward Euler step, it seems to be possible to suppress the weak instability associated with leap-frog differencing [19, 36]. For the computed solutions described below we used an Euler step every 20 time steps.

## 4. Results of the Computer Experiments

The numerical solution of the scheme described in the previous sections for different initial conditions were carried out on the CDC-7600 computer at NASA–Ames Research Center. The computing time per computational time step to move a vortex made of $m$ markers was approximately $0.34 + m/5000$ CPU sec. All calculations were done on a $16^3$ mesh and with leap-frog stepping in time that involves only one evaluation of the derivative per step [39].

A first experiment was done on a single vortex ring of radius $R$ about the $z$ axis. Its center is initially located at (8, 8, 8) in our mesh and thereafter moves along the $z$ axis. The circulation is $\Gamma = 2$ and a fixed $\Delta t = 0.03$ is used. Each ring is discretized by 360 markers ($m = 360$). Therefore the maximum error in the discretized representation of $\mathbf{r}(\xi)$ (see Section 3.1) is $(1 - \cos(\pi/360))\, R = (0.00004)\, R$. In particular, we investigated the initial speed of the vortex ring as a function of radius and position

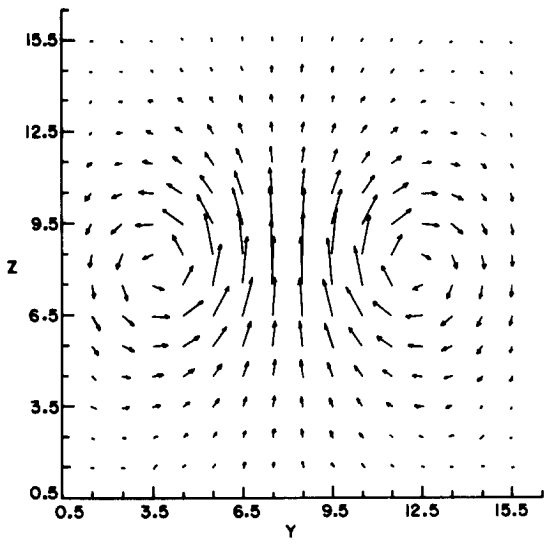



FIG. 3. Projection on ($y$–$z$) plane ($x = 6.5$) of the velocity field in the middle of the cells for a single vortex ring ($R = 4$) centered at (8, 8, 8) in a $16 \times 16 \times 16$ mesh.

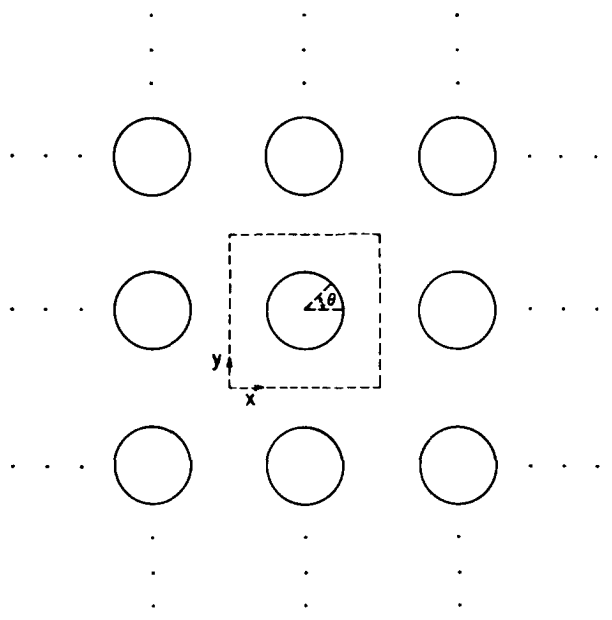



FIG. 4. Periodic array of single vortex rings, all having the same circulation.

around the ring and the time evolution of the shape of the ring. Figure 3 shows the initial velocity field projected on the $(y-z)$-plane at $x=6.5$.

To check the accuracy of our mesh technique we also computed the initial speed of the vortex ring using a continuum or Green's function approach. Since the filter we use in the mesh method is approximately Gaussian we consider a single vortex ring of Gaussian cross section. Following the procedure as defined by (2.6)–(2.8), we obtain as its filtered velocity of translation in free space

$$\frac{\partial \mathbf{r}}{\partial t}=\frac{\Gamma}{4\pi}\oint \frac{f(a)}{|\mathbf{r}-\mathbf{r}'|^3}(\mathbf{r}-\mathbf{r}')\times\frac{\partial \mathbf{r}'}{\partial \xi}\,d\xi, \tag{4.1}$$

where $a=|\mathbf{r}-\mathbf{r}'|/2^{1/2}\sigma$, $\sigma$ is the radius of the cross section or the width of the Gaussian filter and $f(a)=2\pi^{-1/2}ae^{-a^2}-\operatorname{erf}(a)$. If $\sigma^2 \ll R^2$, (4.1) can be approximated to yield

$$\frac{\partial \mathbf{r}}{\partial t}\approx\frac{\Gamma}{4\pi R}\left[\ln\left(\frac{8R}{\sigma}\right)-C\right]\mathbf{e}_z,$$

where $C=1.058$ and $\mathbf{e}_z$ is the unit vector in the direction of translation $z$ [37][1]. To (4.1), we then add the Biot–Savart contributions of the periodic images.

[1] Note that the actual speed of a thin vortex ring with a gaussian distribution of vorticity has been calculated by Saffman [33] and is given by the above formula but with $C=0.558$. The difference is due to the fact that Saffman's result is based on the collective motion of an infinite number of vortex tubes with internal interaction between the filaments whereas our result represents the speed of a single computational ring filament.

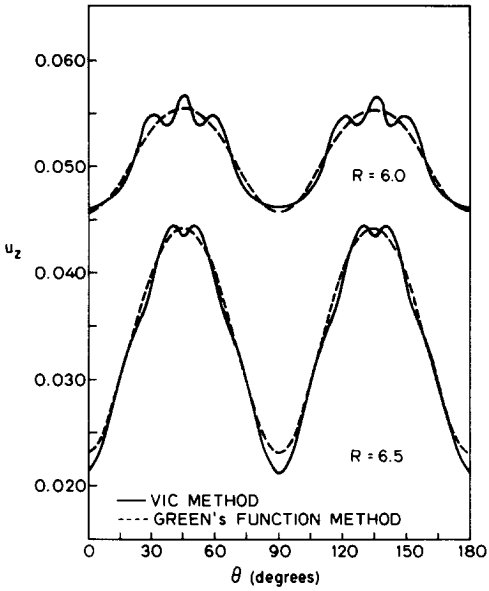



FIG. 5. Velocity of translation versus the angle $\theta$ around the ring for a periodic array of single vortex rings. Radii $R = 6.0$ and $R = 6.5$ are shown. $\Gamma = 2.0$.

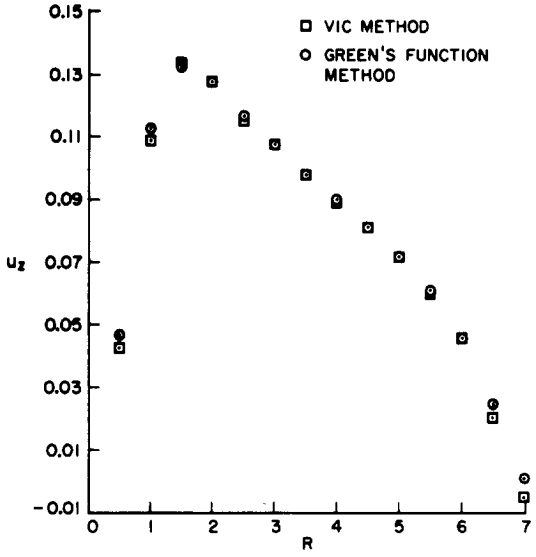



FIG. 6. Velocity of translation ($\theta = 0°$) versus radius for a periodic array of single vortex rings. $\Gamma = 2.0$.

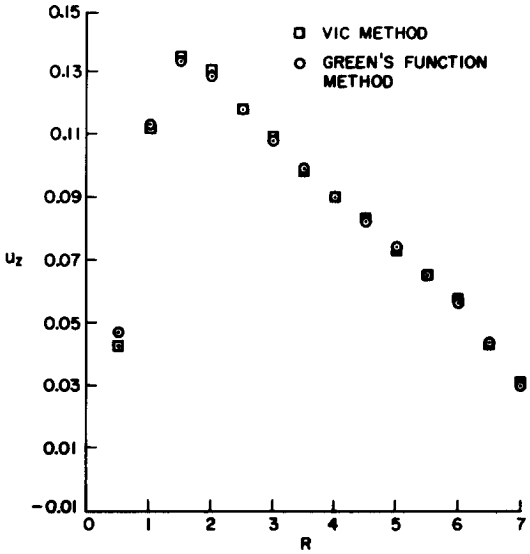



FIG. 7. Velocity of translation ($\theta = 45°$) versus radius for a periodic array of a single vortex rings. $\Gamma = 2$.

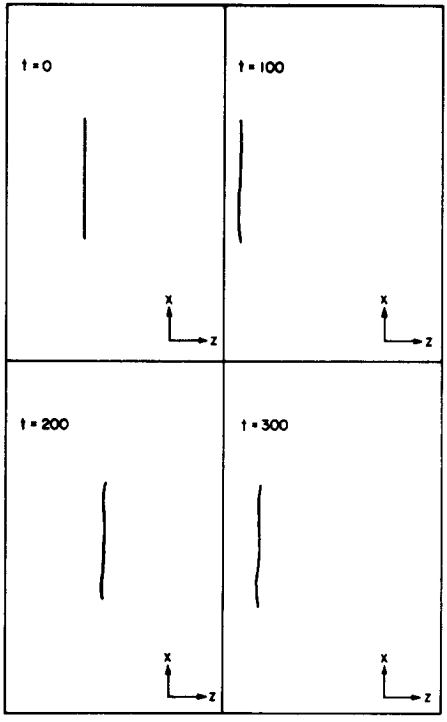



FIG. 8. Displacement of a single vortex ring ($R = 4$) in a periodic space at four instants. $\Gamma = 2$.

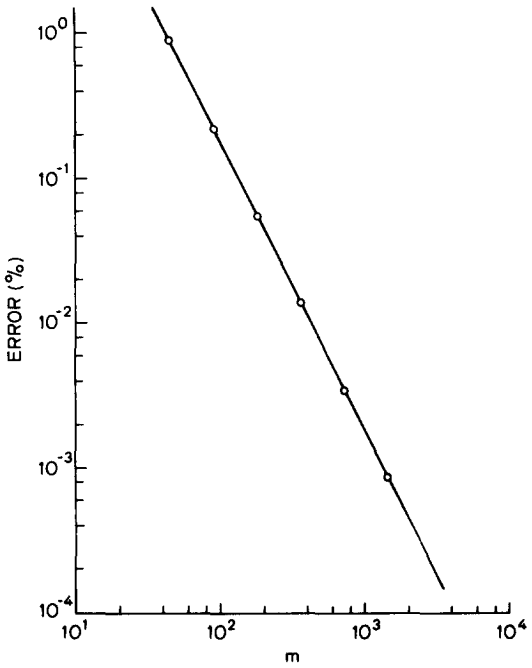



FIG. 9. Percentage of error in the velocity of translation ($\theta = 0$) versus $m$ for a periodic array of single vortex rings. $R = 4$, $\Gamma = 2$.

Remember that the vortex-in-cell (VIC) method used here implies periodic boundary conditions in each of the three dimensions. Thus the initial velocity will not be constant around the circumference of the ring (see Fig. 4). In general the influence of neighboring rings in the same plane is to decrease the local velocity of translation, close neighbors (along the axes) having a stronger influence than neighbors farther away (along the diagonals). This trend is shown in Fig. 5 where $u_z$ is seen to have maxima at $\theta = 45°$, 135°,..., and minima at $\theta = 0°$, 90°,.... Note also the very good agreement of the present VIC calculation with the exact results. The effects of the $16 \times 16 \times 16$ mesh are seen to produce only a small-amplitude high-frequency error.

The Gaussian width used in the Green's function calculations was chosen to give the best fit to the vortex-in-cell results and was found to be $\sigma^2 = 1.1$ times the cell area. This is in good agreement with a theoretical estimate of $\sigma^2 = 12/\pi^2 \simeq 1.2$ based on a Gaussian fit to the low $|\mathbf{k}|$ behavior of our filter. Recall that our filter is not strictly Gaussian but is brought smoothly to zero at $|\mathbf{k}| = \pi$.

In Fig. 6 and 7, the minimum ($\theta = 0°$) and maximum ($\theta = 45°$) velocities, respectively, are shown for different ring radii $R$. The agreement is very good over the entire range of ring radii. Figure 8 shows the lateral vortex profile in the $(x - z)$ plane at four instants. Some small deformation at later times is noticeable due to the nonsym-

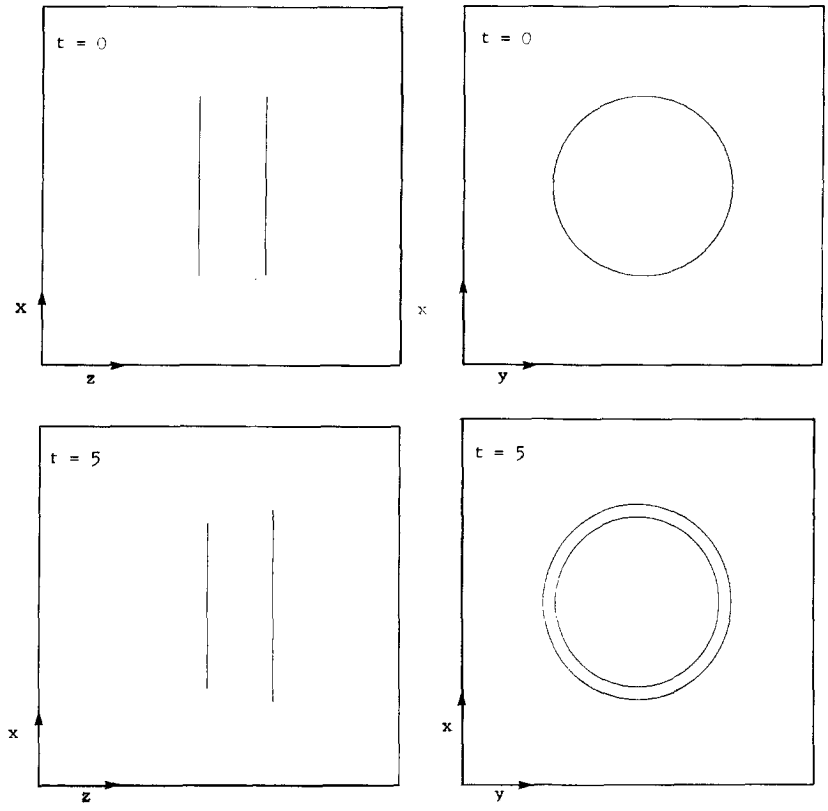



FIG. 10. Displacement of a pair of vortex rings ($R = 4$) in a periodic space at ten instants. $\Gamma = 2$.

metric influence of images. We also investigated the error due to the discretization of the ring as discussed in Section 3.1. In Fig. 9, we show the percentage of error in the velocity of translation ($\theta = 0$) due to this effect as a function of the number of markers $m$ used. Note a $1/m^2$ dependence and the small error the discretization used in this study, 0.0138% for $m = 360$.

A second test was done on a set of two vortex rings of radius $R = 4$ about the $z$ axis. Their centers are initially located at (8, 8, 7) and (8, 8, 10) in the mesh and thereafter move also along the $z$ axis. Both have the same circulation $\Gamma = 2$. As before $\Delta t = 0.03$ and $m = 360$ for each ring.

We know that two similar vortex rings at some distance apart on a common axis of symmetry will do the following: The velocity field associated with the rear vortex ring has a radially outward component at the position of the front ring and so the radius of the front ring gradually increases (with $\Gamma$ constant). This leads to a decrease

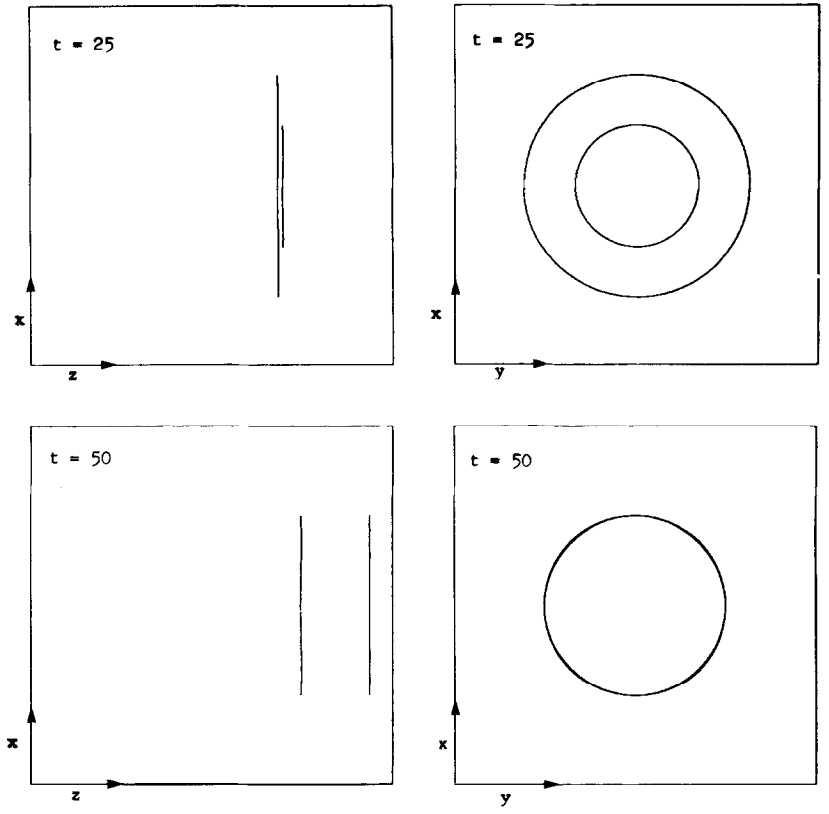



FIG. 11. Continuation of Fig. 10.

in its velocity of translation, and there is a corresponding increase in the velocity of translation of the rear vortex which ultimately passes through the larger vortex and in turn becomes the front vortex. The process is then repeated. Indeed, we observed that process. The last five figures (Figs. 10–14) show the displacement in the $(x–z)$ plane and the $(x–y)$ plane at 10 instants. We see the rings going through each other repeatedly and the build-up of distortions due to the influence of images.

To test our scheme for translation invariance, we recomputed the above two-ring case with the ring centers each displaced initially by $(\frac{1}{2}, \frac{1}{2}, \frac{1}{2})$ from the first calculation. Figures 15 and 16 give the comparison of the results at $t = 100$ and $t = 150$. At $t = 100$, differences between the two cases are barely perceptible while at $t = 150$, the side views in particular show noticeable differences, undoubtedly due to the accumulation of very small differences at each time step.

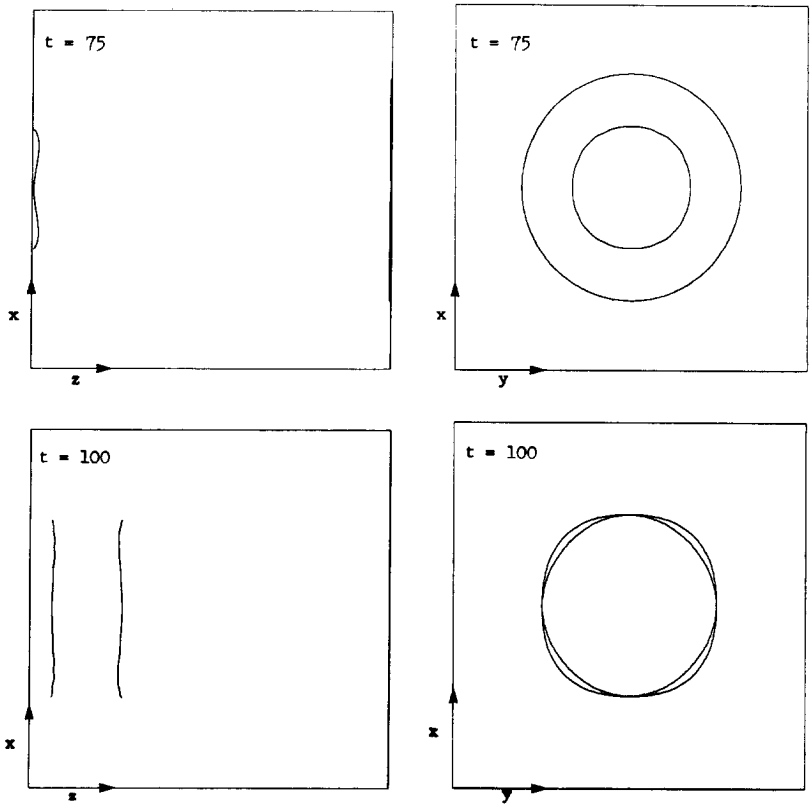



FIG. 12. Continuation of Fig. 11.

## 5. CONCLUSIONS

A new vortex-in-cell method for the numerical simulation of three-dimensional incompressible flows has been developed. The computational elements are vortex filaments whose motion is given by the velocity field the filaments themselves produce. High efficiency is obtained by creating a mesh record of the vorticity field, integrating Poisson's equation via fast Fourier transforms to obtain the velocity field on the mesh, and then interpolating velocities back onto the filaments. High accuracy is obtained by a carefully constructed quadratic spline interpolation scheme and by filtering in wavenumber space to impose a Gaussian vorticity distribution within the core of each filament. The filtering process also provides subgrid dissipation.

As shown in the computed motion of vortex rings, our Lagrangian treatment of the vorticity allows the convection of flow structures over long distances without

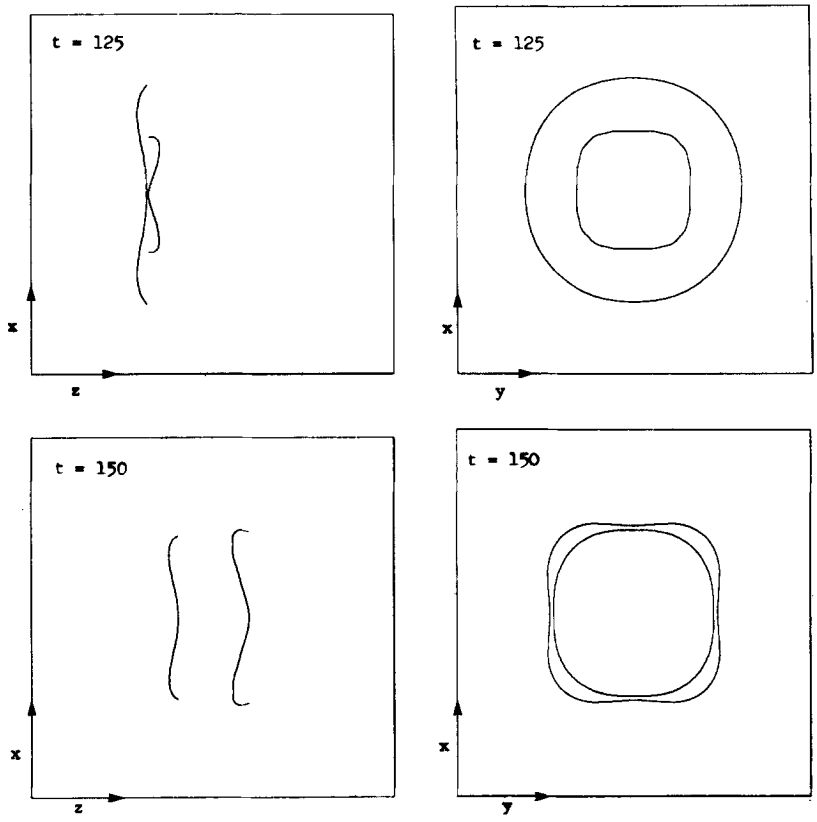



FIG. 13. Continuation of Fig. 12.

diffusive effects caused by differencing of convective derivatives. In testing the numerical method, no undesirable grid effects or numerical instabilities were found.

## 6. APPENDIX

On one hand, using the series expansion for sine, we have letting $\mathbf{k} \cdot [(\mathbf{r}_j - \mathbf{r}_{j-1})/2] = \varepsilon_j$,

$$\frac{\sin \varepsilon_j}{\varepsilon_j} = 1 - \frac{1}{6}\varepsilon_j^2 + \frac{1}{120}\varepsilon_j^4 - \cdots .$$

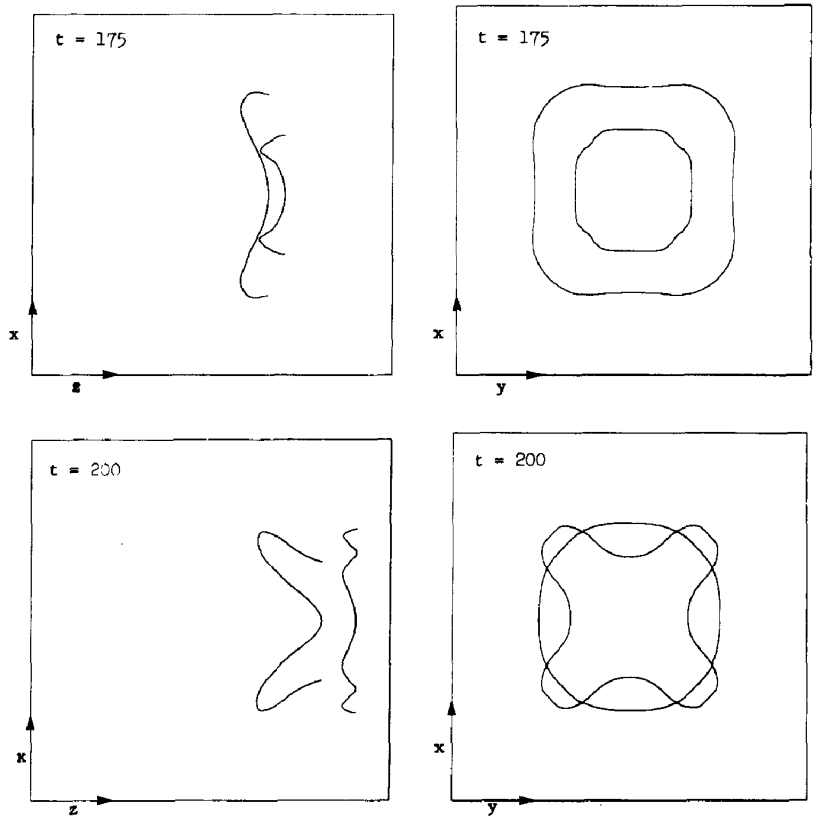



FIG. 14. Continuation of Fig. 13.

On the other hand, we obtained in (3.1)

$$\hat{\boldsymbol{\omega}}(\mathbf{k}) = \Gamma \sum_{j=1}^{m} (\mathbf{r}_j - \mathbf{r}_{j-1}) I_j,$$

where $I_j$ stands for $\int_0^1 \exp(-i\mathbf{k} \cdot [\xi \mathbf{r}_j + (1-\xi)\mathbf{r}_{j-1}])\, d\xi$. A Gaussian type integration [28] can be used to evaluate $I_j$, keeping the approximation down to the sum of two exponentials. Using the zeros of the second-order Legendre polynomial:

$$P_2(x) = \tfrac{1}{2}(3x^2 - 1) = 0,$$

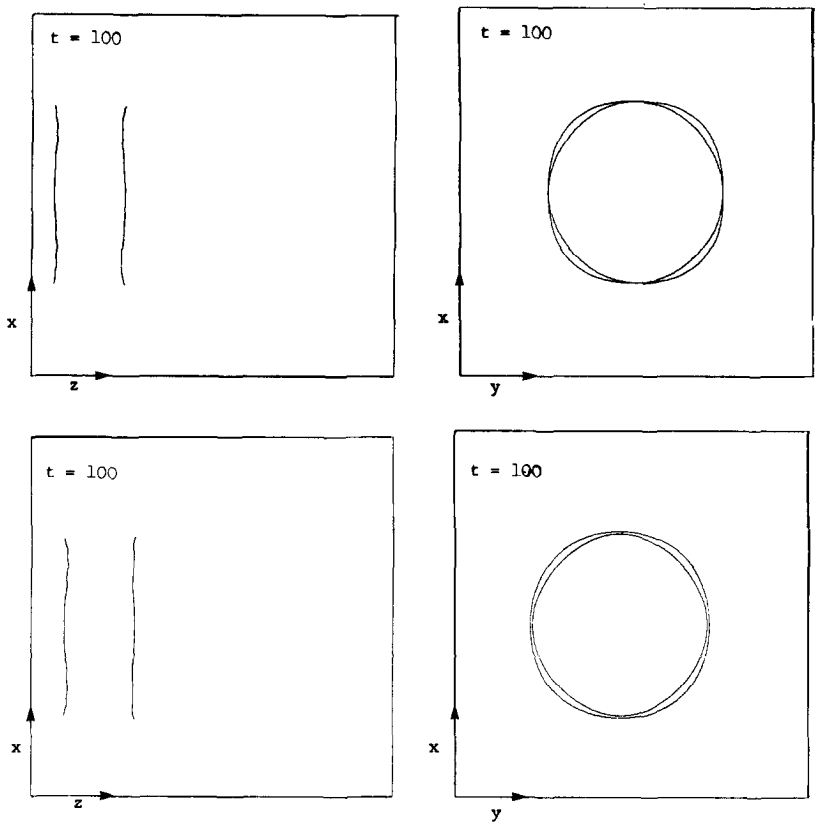



FIG. 15. Comparison in the displacement of a pair of vortex rings ($R = 4$) in a periodic space. Top: rings with centers at (8, 8, 7) and (8, 8, 10) initially. Bottom: rings with centers at (7.5, 7.5, 7.5) and (7.5, 7.5, 10.5) initially.

which implies $x_1 = +3^{-1/2}$ and $x_2 = -3^{-1/2}$, we obtain

$$I_j = \tfrac{1}{2}\{\exp(-i\mathbf{k}\cdot\tfrac{1}{2}[(1+3^{-1/2})\,\mathbf{r}_j + (1-3^{-1/2})\,\mathbf{r}_{j-1}]) + \exp(-i\mathbf{k}\cdot\tfrac{1}{2}[(1-3^{-1/2})\,\mathbf{r}_j + (1+3^{-1/2})\,\mathbf{r}_{j-1}])\} + \text{Error},$$

where

$$|\text{Error}| = \left|\frac{[\mathbf{k}\cdot(\mathbf{r}_j - \mathbf{r}_{j-1})]^4}{135}\exp(-i\mathbf{k}\cdot[\xi\mathbf{r}_j + (1-\xi)\,\mathbf{r}_{j-1}])\right|, \qquad 0 < \xi < 1,$$

$$= \frac{16\varepsilon_j^4}{135}.$$

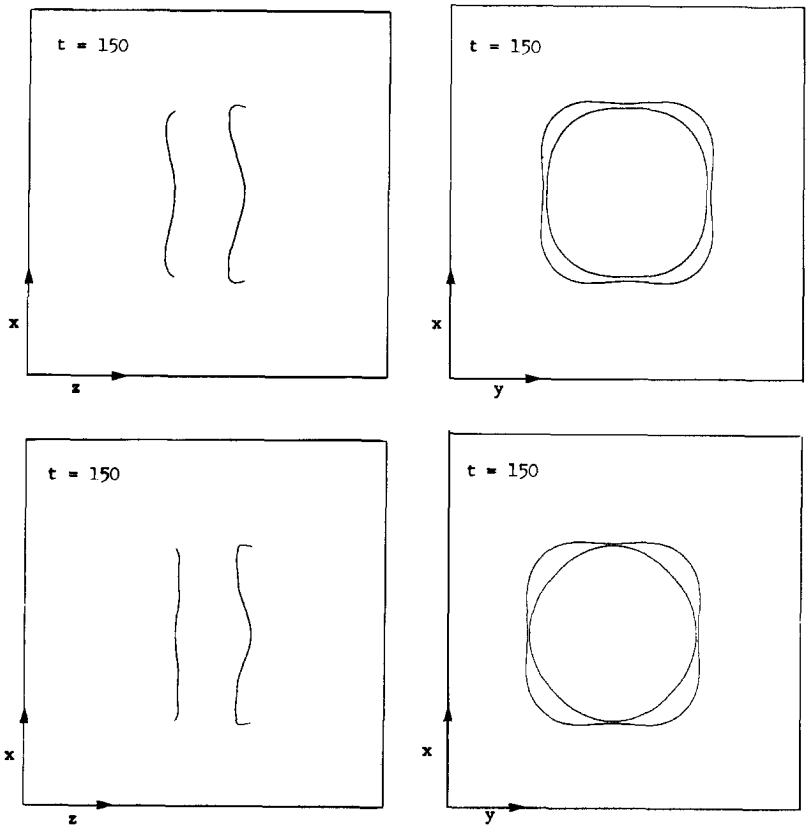



FIG. 16. Same as Fig. 15 at a later time.

This accuracy to the third order in $\varepsilon_j$ is shown in the resulting approximation of

$$\frac{\sin \varepsilon_j}{\varepsilon_j} \simeq \frac{e^{-\epsilon_j 3^{-1/2}} + e^{i\epsilon_j 3^{-1/2}}}{2}$$

$$= \cos\left(\frac{\varepsilon_j}{3^{1/2}}\right) = 1 - \frac{\varepsilon_j^2}{6} + \frac{\varepsilon_j^4}{216} - \cdots .$$

## ACKNOWLEDGMENT

This work was initially supported by a NASA Interchange Agreement with the Ames Research Center and subsequently by the Office of Naval Research.

## REFERENCES


1. L. ROSENHEAD, *Proc. Roy. Soc. A* **134** (1931), 170–192.
2. P. T. FINK AND W. K. SOH, "Calculation of Vortex Sheets in Unsteady Flow and Applications in Ship Hydrodynamics," Univ. New South Wales Report, NAV/ARCH 74/1, 1974.
3. B. B. KADOMTSEV AND D. P. KOSTOMAROV, *Phys. Fluids* **15** (1972), 1–3.
4. K. KUWAHARA AND H. TAKAMI, *J. Phys. Soc. Japan* **34** (1973), 247–253.
5. J. P. CHRISTIANSEN AND N. J. ZABUSKY, *J. Fluid Mech.* **61** (1973), 219–243.
6. A. J. CHORIN, *J. Fluid Mech.* **57** (1973), 785–796.
7. W. T. ASHURST, *in* "Turbulent Shear Flows I" (F. Durst *et al.*, Eds.) Springer–Verlag, Berlin, 1979, pp. 402–413.
8. W. T. ASHURST, "Vortex Dynamic Calculation of Fluid Motion in a Four Stroke Piston 'Cylinder'—Planar and Axisymmetric Geometry," SAND79-8229, Sandia Labs., Livermore, Calif., 1978.
9. A. I. SHESTAKOV, "Numerical Solution of the Navier–Stokes Equations at High Reynolds Numbers," Lecture Notes in Physics No. 59, Springer–Verlag, New York, 1976, pp. 404–409.
10. R. S. ROGALLO, "Proceedings of the Second Computational Fluid Dynamics Conference," AIAA, Vol. 67, 1975.
11. J. A. L. THOMSON AND J. C. S. MENG, "Studies of Free Buoyant and Shear Flows by the Vortex-in-Cell Method," Lecture Notes in Physics No. 35, Springer–Verlag, New York, 1974, pp. 401–416.
12. J. C. S. MENG AND J. A. L. THOMSON, *J. Fluid Mech.* **84** (1978), 433–453.
13. F. H. ABERNATHY AND R. E. KRONAUER, *J. Fluid Mech.* **13** (1962), 1–20.
14. A. MICHALKE, *Ing. Arch.* **33** (1964), 264–276.
15. E. ACTON, *J. Fluid Mech.* **76** (1976), 561–592.
16. J. P. CHRISTIANSEN, *J. Comput. Phys.* **13** (1973), 363–379.
17. H. AREF AND E. D. SIGGIA, "Vortex Dynamics of the Turbulent Shear Layer," Lab. of Atomic and Solid State Physics, Cornell Univ., Ithaca, N. Y., 1979.
18. S. S. WANG, "Grid-Insensitive Computer Simulation of the Kelvin–Helmholtz Instability and Shear Flow Turbulence," Ph. D. thesis, Stanford University Institute for Plasma Research Report 710, 1977.
19. A. LEONARD, "Numerical Simulation of Interacting, Three-Dimensional Vortex Filaments," Lecture Notes in Physics No. 35, Springer–Verlag, New York, 1975, pp. 245–250.
20. A. LEONARD, "Simulation of Three-Dimensional Separated Flows with Vortex Filaments," Lecture Notes in Physics No. 59, Springer–Verlag, New York, 1976, pp. 280–284.
21. J. C. S. MENG, *J. Fluid Mech.* **84** (1978), 455–469.
22. A. LEONARD, *in* "Second Symposium on Turbulent Shear Flows," Imperial College, London, 1979.
23. J. H. FERZIGER, *AIAA J.* **15** (1977), 1261–1267.
24. S. A. ORSZAG AND G. S. PATTERSON, *Phys. Rev. Lett.* **28** (1972), 76–79.
25. S. A. ORSZAG AND D. S. RAILA, *Phys. Fluids***16** (1973), 172–173.
26. D. G. FOX AND S. A. ORSZAG, *Phys. Fluids* **16** (1973), 169–171.
27. S. A. ORSZAG, *Studies in Appl. Math.* **50** (1971), 293–397.
28. M. ABRAMOVITZ AND S. A. STEGUN, "Handbook of Mathematical Functions," Dover, New York, 1968.
29. O. BUNEMAN, *J. Comput. Phys.* **11** (1973), 250.
30. O. BUNEMAN, "Variationally Optimized, Grid-Insensitive Vortex Tracing," Lecture Notes in Physics, No. 35, Springer–Verlag, New York, 1974, pp. 111–115.
31. A. J. CHORIN AND P. S. BERNARD. *J. Comput. Phys.* **13** (1973), 423.
32. C. TUNG AND L. TING, *Phys. Fluids* **10** (1967), 901–910.
33. P. G. SAFFMAN, *Studies in Appl. Math.* **44** (1970), 371.
34. P. HENRICI, "Discrete Variable Methods in Ordinary Differential Equations," Wiley, New York, 1962.
35. R. W. HAMMING, "Numerical Methods for Scientists and Engineers," 2nd ed., McGraw–Hill, New York, 1973.

36. D. K. LILLY, *Mon. Weather Rev.* **93** (1965), 11–26.
37. B. COUËT, "Evolution of Turbulence by Three-Dimensional Numerical Particle-Vortex Tracing," Ph. D. thesis, Stanford University Institute for Plasma Research 793, 1979.
38. B. COUËT AND A. LEONARD, Exact extension to the infinite domain for the vortex-in-cell method, submitted to *SIAM J. Sci. Satist. Comput.* In this paper, the periodicity condition is relaxed.
39. B. COUËT AND A. LEONARD, *in* "Second Symposium on Turbulent Shear Flows," Imperial College, London, 1979. This paper described results obtained with a $32^3$ code.