Vortex Particle Methods for Periodic Channel Flow

J. J. MONAGHAN AND R. J. HUMBLE

Department of Mathematics, Monash University, Clayton, Victoria 3168, Australia

Received March 8, 1991; revised September 28, 1992

The boundary conditions for vortex particles in two-dimensional channel flow with periodic boundary conditions require infinitely many image vortices. By using the Poisson summation formula with Ewald's trick the effect of image vortices can be calculated efficiently. When combined with a tree code and appropriate smoothing, a fast robust algorithm can be designed. Applications to Kirchoff and Kida vortices are described. © 1993 Academic Press, Inc.

1. INTRODUCTION

Tropical cyclones can be viewed as patches of vorticity moving in a two-dimensional inviscid fluid (the barotropic plane) [15]. Similar models apply to the great spots on Jupiter and Neptune [13]. In order to follow the long term evolution of these systems numerical methods which transport vorticity accurately are required. Vortex particle methods [1, 4, 11, 14] are a natural candidate for this purpose, and they have the bonus that detailed Lagrangian information is given about the motion of parcels of fluid.

For numerical purposes the infinite barotropic plane must be replaced by a finite computational plane. Boundary conditions are then chosen to reduce edge effects. The typical configuration is a rectangular computational region with periodic boundary conditions on the east–west ends, and free-slip conditions on the north–south boundaries which are assumed impenetrable [15].

If vortex methods are to be applied to this configuration, infinitely many image vortices are required to satisfy the boundary conditions. Greengard [7] has considered a version of this problem with pure channel flow and has shown that it can be computed efficiently with a fast multipole moment algorithm. Greengard works with complex variables which enable the infinite terms from the north–south boundaries to be summed to a complex cosh function. This procedure could be extended to deal with the periodic east–west conditions. However, we wish to allow for smoothing of close interactions and this makes Greengard's method less attractive because some interactions must be treated separately. For the problem considered here, a flexible and efficient algorithm can be divised by using the Poisson summation formula and Ewald's [5] trick to tame the image vortices (Ewald's trick was used by Ewald to determine the lattice energy of a crystal). The end result is that the infinite sum is split into a rapidly convergent direct interaction and a rapidly convergent sum which represents the far field. By combining this with a tree code, the velocity of any vortex particle can be computed rapidly. Smoothing, with a local smoothing for each vortex particle, can be included easily.

In this paper details of the new method and applications to the Kirchoff and Kida vortices will be described.

2. THE VORTEX PARTICLE CONFIGURATION

We consider a set of N_p vortex particles in a rectangle with sides of length a parallel to the x axis (west–east) and sides of length b parallel to the y axis (south–north). The origin of the coordinate system is the lower left (southwest) corner. The system is periodic in the east–west direction. Each vortex particle has infinitely many images due both to the periodic condition and to the impenetrable north and south boundaries.

To satisfy the north and south boundary conditions a vortex particle with coordinates x, y and strength K requires image particles of the same strength with coordinates

$$x,\ y \pm 2nb, \qquad n = 1, 2, ..., \tag{2.1}$$

and image particles of opposite strength with coordinates

$$x, -y \pm 2nb, \qquad n = 0, 1, 2, ... \tag{2.2}$$

These images are generated by first finding the two images of the original particle in the north and south boundaries, then their images, and so on. In addition, each of the previous images has an infinite number of images from the periodic conditions. These images have x coordinates $x \pm ma$, $m = 1, 2, ...$

0021-9991/93 $5.00

The contribution to the velocity of the vortex particle with coordinates x_p, y_p from the vortex particle, with strength K at x, y and its images, is

$$\Delta v_x = -\frac{K}{2\pi}\sum_{m=-\infty}^{\infty}\sum_{n=-\infty}^{\infty}\frac{y_p-y-2nb}{(y_p-y-2nb)^2+(x_p-x-ma)^2}$$
$$+\frac{K}{2\pi}\sum_{m=-\infty}^{\infty}\sum_{n=-\infty}^{\infty}\frac{y_p+y-2nb}{(y_p+y-2nb)^2+(x_p-x-ma)^2} \tag{2.3}$$

and

$$\Delta v_y = \frac{K}{2\pi}\sum_{m=-\infty}^{\infty}\sum_{n=-\infty}^{\infty}\frac{x_p-x-ma}{(y_p-y-2nb)^2+(x_p-x-ma)^2}$$
$$-\frac{K}{2\pi}\sum_{m=-\infty}^{\infty}\sum_{n=-\infty}^{\infty}\frac{x_p-x-ma}{(y_p+y-2nb)^2+(x_p-x-ma)^2}. \tag{2.4}$$

The numerical problem is that the total velocity for the vortex article at x, y requires the evaluation of (2.3) and (2.4) for all the other vortex particles in the rectangle. Any one of the summations (m or n) can be completed analytically using Mittag–Leffler expansions. The remaining summation then creates awkward problems which are compounded when smoothing is included. We prefer to keep the functions simple and tackle the double summation directly. In the next two sections we show how this can be done efficiently.

3. EVALUATING THE IMAGE SUMMATION

The summations are of the form

$$\sum_n\sum_m f_{nm}, \tag{3.1}$$

which can be written

$$\sum_n\sum_m f_{nm}E_{nm}+\sum_n\sum_m f_{nm}(1-E_{nm}), \tag{3.2}$$

where E_{nm} is a function (E for Ewald) which is small for $|n|$, $|m|$ large. Typically $E_{nm}\to 0$ like a Gaussian as $|n|, |m|\to\infty$. The first summation in (3.2) is then easy to evaluate. The second summation can be transformed by the Poisson summation formula to give a rapidly convergent series.

The Poisson summation version of the second summation in (3.2) is

$$\sum_{p=-\infty}^{\infty}\sum_{s=-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}e^{-2\pi i(pn+sm)}f_{nm}\{1-E_{nm}\}\,dn\,dm. \tag{3.3}$$

The choice of E_{nm} is crucial. It must allow the first summation to be evaluated easily, and it must allow the Fourier transforms in (3.3) to be evaluated in a convenient form. A Gaussian E has the desired properties.

Consider the application of these ideas to the first summation in (2.3):

$$\sum_m\sum_n f_{nm} \tag{3.4}$$

with

$$f_{nm}=\frac{y_p-y-2nb}{(y_p-y-2nb)^2+(x_p-x-ma)^2}. \tag{3.5}$$

We take

$$E_{nm}=e^{-q^2\omega_{nm}^2}, \tag{3.6}$$

where $\omega_{nm}^2=(y_p-y-2nb)^2+(x_p-x-ma)^2$, and (3.4) becomes

$$\sum_m\sum_n\frac{(y_p-y-2nb)}{\omega_{nm}^2}\exp(-q^2\omega_{nm}^2)$$
$$+\sum_m\sum_n\frac{(y_p-y-2nb)}{\omega_{nm}^2}(1-\exp(-q^2\omega_{nm}^2)). \tag{3.7}$$

With the new variables

$$u=x_p-x-ma, \qquad v=y_p-y-2nb, \tag{3.8}$$

(3.3) becomes

$$\sum_p\sum_s\frac{1}{2ab}e^{-i\mathbf{k}\cdot(\mathbf{r}_p-\mathbf{r})}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}e^{i\mathbf{k}\cdot\mathbf{w}}\frac{v\,du\,dv}{u^2+v^2}(1-e^{-q^2(u^2+v^2)}), \tag{3.9}$$

where

$$\mathbf{k}=2\pi\left(\frac{s}{a},\frac{p}{2b}\right)$$
$$\mathbf{r}_p-\mathbf{r}=(x_p-x,\ y_p-y)$$
$$\mathbf{w}=(u,v).$$

The evaluation of the double integral is discussed in Appendix A. It vanishes if $p=s=0$. Otherwise it has the value

$$-\frac{2\pi}{i}\left(\frac{\pi p}{bk}\right)\exp(-k^2/(4q^2)). \tag{3.10}$$

Taking note of the p factor in (3.10), (3.9) becomes

$$\frac{\pi^2}{ab^2}\sum_p\sum_s \frac{p\sin \mathbf{k}\cdot(\mathbf{r}_p-\mathbf{r})}{k^2}\exp(-k^2/(4q^2)), \tag{3.11}$$

where $k^2=4\pi^2(s^2/a^2+p^2/(4b^2))$ and the term $p=s=0$ is excluded. With the appropriate choice of q^2 (see below) a sufficiently accurate evaluation of (3.11) only requires $|p|\leqslant 1$ and $|s|\leqslant 1$. In practice we do not evaluate (3.11) directly. Instead, since it represents the slowly varying far field component, we evaluate it on a coarse grid in the rectangle containing the vortex particles and then interpolate to the vortex particles.

The second summation in (2.3) only requires a change in the sign of y and $\mathbf{r}_p-\mathbf{r}$ in (3.11) is replaced by (x_p-x, y_p+y).

The first summation in (2.4) can be evaluated in the same way. The same E_{nm} leads to the Poisson summation

$$\frac{2\pi^2}{a^2b}\sum_p\sum_s \frac{s\sin(\mathbf{k}\cdot(\mathbf{r}_p-\mathbf{r}))}{k^2}\exp(-k^2/(4q^2)), \tag{3.12}$$

where all variables have the same meaning as in (3.9). The final expressions for Δv_x and Δv_y are given in Appendix B.

A reasonable choice for q^2 is to choose it so that there is a balance between the convergence of the n, m summation and the p, s summation. Each is largely controlled by exponential terms of the same type. Requiring the rate of falloff to be the same, we obtain

$$\frac{1}{4q^2}\left(4\pi^2\left(\frac{1}{a^2}+\frac{1}{4b^2}\right)\right)=q^2(a^2+4b^2) \tag{3.13}$$

$$\Rightarrow q^2=\frac{\pi}{2ab}. \tag{3.14}$$

Typically we use rectangles with $a=2b$ and the terms decrease as $\exp(-2\pi)$. The actual situation is complicated by particles with small or large differences in $|y_p-y|$ and $|x_p-x|$. Taking note of the fact that we smooth close interactions, satisfactory accuracy is achieved with $n, m=1, 0, 1$.

4. EVALUATING THE FINAL SUMMATIONS

The evaluation of the summations over m, n with E_{nm} converge rapidly. In practice this means that satisfactory accuracy is achieved with $n, m=0, \pm 1$. However, each vortex particle then has 14 images (two image vortices in each cell above and below; these are then repeated left and right according to the periodicity requirement making 12, then two from the left and right repeats of the original vortex particle).

We evaluate the contributions of these image particles using a quad tree ([2, 8] describe the three-dimensional equivalent). The particles and their images are used to calculate the node properties on the tree. This process is fast and is proportional to $N_p n_i$, where n_i is number of images (14) per particle.

The tree was constructed by covering the vortex particles and their images by a square. This was divided into four cells. Each of these cells was divided into four cells and so on, down to the chosen deepest level.

The node properties of the cells include $\sum_j K_j$, $\sum_j x_j' K_j$, $\sum_j y_j' K_j$, $\sum_j x_j'^2 K_j$, $\sum_j y_j'^2 K_j$, and $\sum_j x_j' y_j' K_j$ which enable the interaction to be replaced by moments about the cell centre (thus $x_j'=x_j-x_c$, where x_c is the x coordinate of the cell centre and j denotes a vortex particle or image in the cell). The expansion is about the cell centre rather than the centre of vorticity since this quantity, unlike the centre of mass, can be outside the cell, and even at infinity, because the vorticity strengths can be positive or negative. The expansions required are given in Appendix C.

Once the tree is constructed the contribution to the velocity can be found by traversing the tree. Since this is only done for the vortex particles, the presence of 14 images per particle has only a tiny influence on the computational time.

The algorithm runs significantly faster if the tree is traversed with a group of particles. This can be done by choosing a group to be the particles in a cell at the deepest level and using a link list to access particles. The tree is then traversed by each link-list cell in turn. Decisions about opening nodes are made using the distance between the centre of the link-list cell and the node cell.

As mentioned earlier, the far field (Poisson sum) terms are calculated on a grid (10 × 10 is sufficient), then interpolated to the particles. For the calculations described here, bi-linear interpolation was used.

5. SMOOTHING

Smoothing the interaction gives better results, because the vortices then more accurately approximate a smooth vorticity distribution [11]. The direct interaction between two vortices in the $x-y$ plane, one at the origin with strength K, and one at r, gives the second particle a velocity

$$\mathbf{v}=\frac{K}{2\pi}\frac{\hat{\mathbf{z}}\times\mathbf{r}}{r^2}, \tag{5.1}$$

The smoothed interaction replaces $1/r^2$ by

$$\frac{1}{r^2}\int_0^r Wr\,dr, \tag{5.2}$$

where W is a function which is zero for $r>h$. We can take

W to be the piecewise continuous (C^2) function based on a spline (see [12] for the three-dimensional versions) given by

$$h^2 W(r, h) = \frac{15}{7\pi} \begin{cases} \frac{2}{3} - g^2 + \frac{1}{2} g^3; & 0 \leqslant g \leqslant 1 \\ \frac{1}{6}(2-g)^3; & 1 \leqslant g \leqslant 2 \\ 0; & g > 2, \end{cases} \tag{5.3}$$

where $g = r/h$. This function has compact support so there is no smoothing of the interaction for $r > 2h$. The smoothed interaction factor

$$S = \int_0^r Wr \, dr$$

is easily found to be

$$\frac{30}{7} \begin{cases} \frac{1}{3} g^2 - \frac{1}{4} g^4 + \frac{1}{10} g^5; & 0 \leqslant g \leqslant 1 \\ \frac{2}{3} g^2 - \frac{2}{3} g^3 + \frac{1}{4} g^4 - \frac{1}{30} g^5 - \frac{1}{30}; & 1 \leqslant g \leqslant 2 \\ \frac{7}{30}; & g \geqslant 2. \end{cases} \tag{5.4}$$

In practice (5.4) is not evaluated directly. A table of the function (5.2) is calculated and when the smoothed interaction is required it is interpolated from the table. To speed up the calculations, the exponentials in the n, m summation are also interpolated from a table.

6. INVARIANTS

For a set of N interacting point vortices with strength K_j, $j = 1, 2, ..., N$, it is easy to show [3] that the quantities

$$\sum_j K_j, \sum_j x_j K_j, \sum y_j K_j, \sum K_j(\{x_j - X\}^2 + \{y_j - Y\}^2) \tag{6.1}$$

and

$$\sum_{i \neq j} \sum K_i K_j \log((x_i - x_j)^2 + (y_i - y_j)^2) \tag{6.2}$$

are invariant, where $X = \sum K_j x_j / \sum K_j$ and $Y = \sum K_j y_j / \sum K_j$. The smoothed interaction also preserves five invariants. The first four are identical to (6.1). The invariant (6.2) arises because (5.1) can be written

$$\mathbf{v} = \frac{K}{2\pi} \hat{z} \times \nabla \Psi(r),$$

where $\Psi(r) = \ln(r)$. When smoothing is used

$$\Psi(r) = \int \frac{S(r)}{r} dr, \tag{6.3}$$

and in the invariant (6.2) the log function is replaced by (6.3).

These invariants were used in tests of the tree code on sets of vortex particles in a region with no boundaries. When the tree code parameters were set to ensure the maximum error in v was $\leqslant 0.6\,\%$, the errors in the invariants was $< 1\,\%$.

7. TIME STEPPING

Any convenient time stepping can be used. For the calculations described here, a predictor–corrector method with an Euler predictor to the midpoint, and a corrector step using the midpoint velocity, was used. The error per step δt is then $\propto (\delta t)^3$.

For the applications considered here, where the particles start with minimum separation d, the time step was chosen according to

$$\delta t = 0.125\, d^2/K,$$

where K is the maximum vortex particle strength. Because the strength is $\propto \zeta d^2$, where ζ is the vorticity, the time step is independent of d and, therefore, for the problems described in Section 9, it is independent of the number of vortex particles.

8. COMPARATIVE TIMINGS

Our tree code (T) was tested by applying it to a set of particles with no boundaries and comparing the results with direct summation (D) and the fast multipole moment technique (F) [6]. For 2000 particles the times, in seconds, on a Silicon Graphics Iris for the calculation of all velocities were D : 40 s, T : 15 s, F : 7 s. For 10,000 particles the times were D : 894 s, T : 112 s, F : 57 s. The time scales with particle N as N^2 for D and $N \ln N$ for T and F. The latter should scale as N [6] but overhead may be corrupting the estimate. The first conclusion is that the fast multipole moment technique is the fastest. However, the tree code is very flexible; it can be easily adjusted to include our exponential E factors, and it can be easily adjusted to include a different smoothing length for each particle. For that reason we prefer it. Furthermore, it can be easily applied to three-dimensional (vortex tube) simulations.

It may be of interest to note the times that the full image program with 660 particles took to calculate all velocities for one step on the following computers:

Iris 4D/20:	10.4 s
Dec 2100:	9.0 s
Dec 3100:	6.7 s
Vax 6410:	8.9 s.

The efficiency of the tree code depends on the particle configuration (clumped or uniformly spread), the rule for opening a node, and the maximum depth. We set the maximum depth (level) of the tree to be seven and the rule for *not* opening a node is that its half width H and its distance D from the point whose $\mathbf{v}$ is required satisfies

$$D > \sqrt{13}\, H,$$

giving errors of 0.6% when calculated using the zeroth, linear, and quadratic terms in the node expansion (see Appendix C). By adjusting the condition on D any desired accuracy can be achieved. The error varies as $(H/D)^3$. We have found that at the 0.6% level is satisfactory for the examples treated here (see Section 9). Note that it is an advantage to include the quadratic terms, but experiments showed that including the next terms in the expansion did not repay the extra work.

9. TESTS

The simplest test of the tree code is to apply it to a small number of vortex particles in the absence of boundaries. A single particle should remain at rest; a pair of vortex particles can be set up so that they move in a binary orbit or move off in straight lines. The tree code was found to satisfy these tests accurately.

A similar simple test, but with a boundary, is to place a particle close to a boundary It should move parallel to the boundary.

A detailed comparison with direct summation shows that the tree code with the $D > \sqrt{13}\, H$ rule has errors ~0.6%.

10. APPLICATIONS

A demanding test is to apply the algorithm to the Kirchoff vortex. This vortex is an ellipse of constant vor-

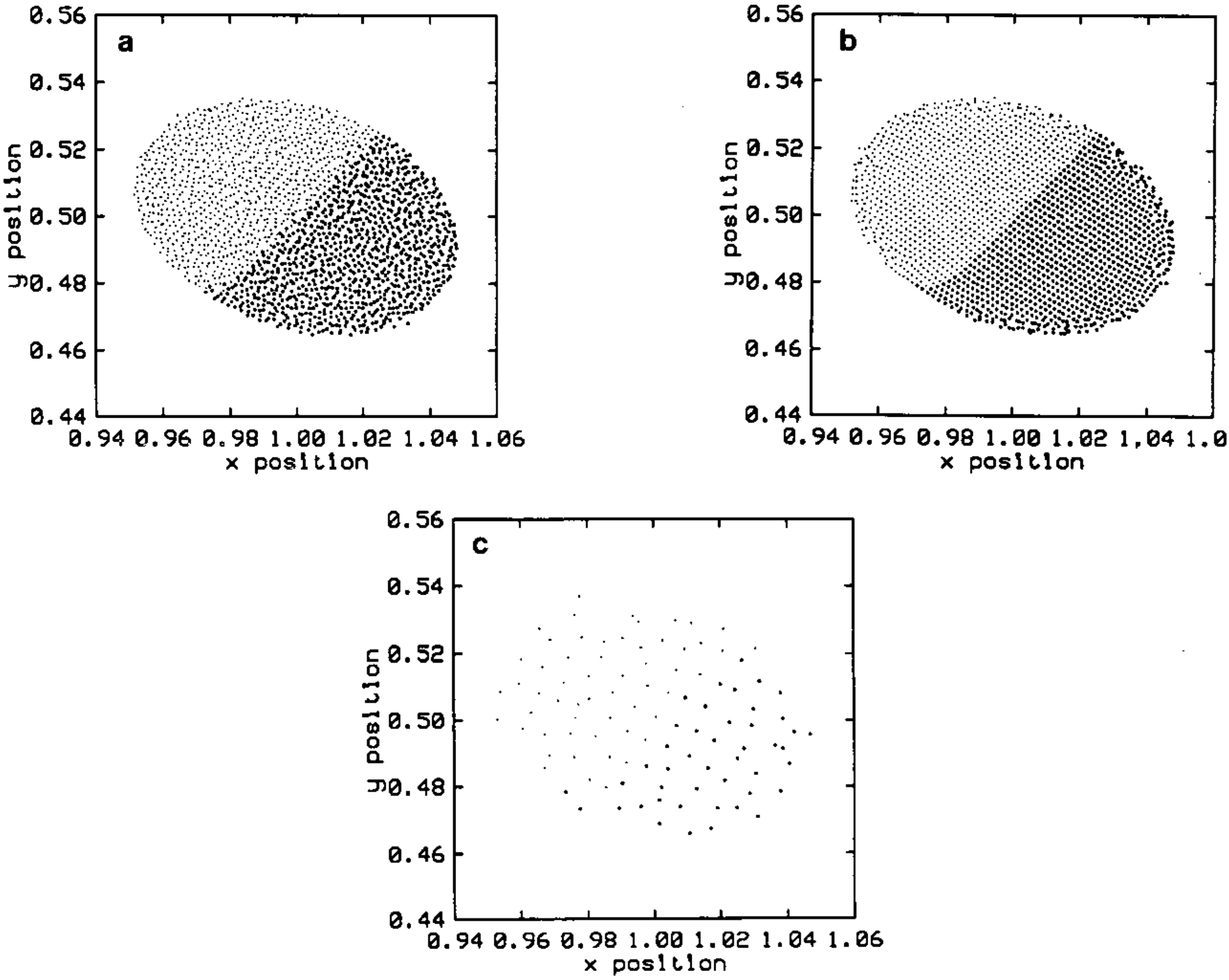

FIG. 1. (a) Kirchoff vortex with no boundaries. 1700 particles and no smoothing. See text for further details. The ellipse was orientated initially so that the major axis was parallel to the x axis. The darker dots denote particles initially in the right half of the ellipse. (b) Kirchoff vortex with no boundaries. 1700 particles and smoothing. (c) Kirchoff vortex with no boundaries. 100 particles and smoothing.

ticity ζ in a background with zero vorticity. In the absence of boundaries the ellipse should rotate with constant angular velocity [10, Art. 159]

$$\omega = \frac{AB\zeta}{(A+B)^2}, \tag{10.1}$$

where A and B are the semi-major and minor axes, respectively. If the vortex is placed in a region of the kind considered in this paper the boundaries will have an effect on the motion of the vortex, but the effect will be small if the region is large compared to the vortex.

Figure 1 shows the motion of a Kirchoff vortex in the absence of boundaries. The axis ratio is 1.5, $\zeta = 2$, $A = 0.05$, and $\omega = 0.48$, so that the period is 13.09. The calculations used 1700 particles. Vortex particles were placed initially on a grid with square cells of area σ. Only the vortices within the ellipse were accepted. The grid therefore produces small stable perturbations on the boundary of the ellipse. The left frame is from a calculation without smoothing, and the right frame is from a calculation with smoothing. It is clear that smoothing keeps the particles more orderly. Because it is intended to apply the technique to cyclonic structures, which may only involve a small number of particles, the vortex motion was simulated with just 100 particles. The results are shown in the third frame of Fig. 1. In this case the edge perturbations are larger because the ellipse is only crudely modelled by particles taken from a grid of square cells. Despite this, the motion is very similar to that of the same vortex with 1700 particles.

Figure 2 shows a frame taken from a simulation of the motion of the Kirchoff ellipse with boundaries of the type considered in this paper ($a = b = 1$). This calculation used 778 particles. The results are very similar to those found without boundaries, showing that the boundary conditions on the computational boundary simulate an infinite sea satisfactorily.

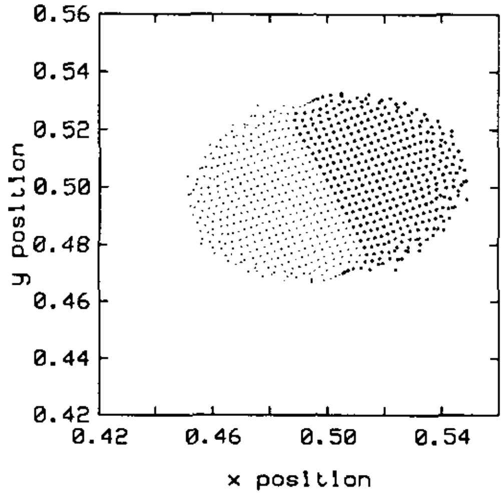

FIG. 2. Kirchoff vortex using the image method with $a = b = 1$.

The Kida vortex is an elliptical patch of vortex in a shear-flow background. Kida showed that the equations could be reduced to a simple Hamiltonian system for the parameters and orientation of the ellipse. The ellipse remains an ellipse, but the semi-major and minor axes, and the orientation, change with time.

The shear flow is a region of constant vorticity. In this case particles are given the appropriate vorticity for the ellipse or the shear flow, depending on their initial position. Because the shear flow carries particles out of the computational zone the east–west periodic conditions require that identical particles reappear through the opposite side. Twenty-five hundred particles were placed in the computational box ($a = b = 1$). The vorticity in the ellipse was one and the vorticity outside was 0.1.

In Fig. 3 the motion of a Kida vortex system [9] is illustrated by showing the velocity vector for each particle. The shear flow is zero along the central line ($y = 0.5$) and

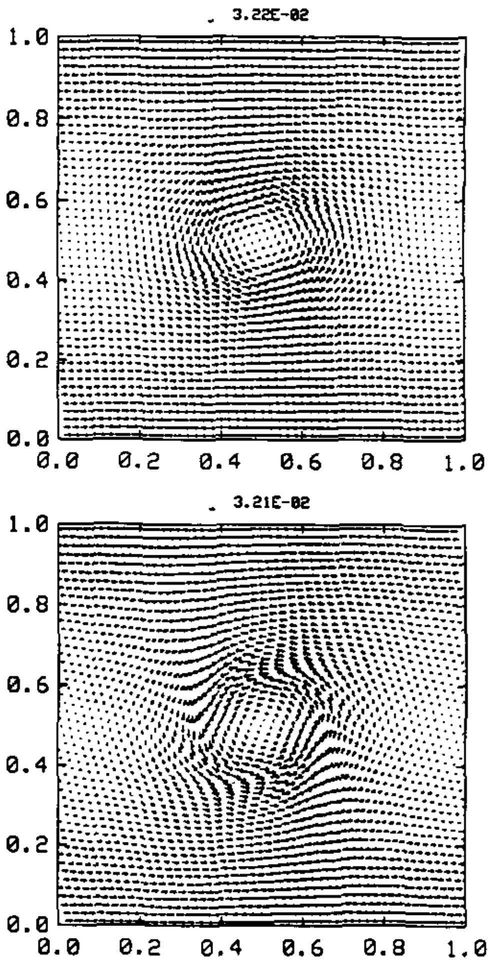

FIG. 3. Elliptical vortex patch in a shear flow (Kida vortex) with images, 2500 particles and smoothing.

should be parallel to the upper and lower boundaries near those boundaries. The vortex particle flow clearly satisfies those conditions. In this case the elliptical patch is stretched by the shear flow, but maintains its elliptical form in agreement with Kida's equations. A more detailed discussion of the Kida vortex will be given elsewhere in connection with the application of this method to cyclones.

11. CONCLUSION

The results of the tests and calculations with Kirchoff and Kida vortices show that the combination of tree code and Ewald's method gives an efficient algorithm. No attempt was made to fine-tune the algorithm with respect to smoothing length, levels in the tree, calculation of the far field on the grid, or time stepping. As a consequence, higher speed can be expected. The method has great flexibility and can be extended to handle the case where the smoothing of the vorticity uses an h for each particle. In this way a wide range in vortex particle number density can be accommodated as in the related case of self-gravitating particles [8].

APPENDIX A

The evaluation of

$$I=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{i\mathbf{k}\cdot\mathbf{w}}[1-e^{-q^2(u^2+v^2)}]\,\frac{v\,dv\,du}{u^2+v^2}. \tag{A.1}$$

Note, first, that if $k=0$ then the integral vanishes because the integrand is an odd function of v. We assume then that $k\neq 0$.

We let $u=w\cos\theta$, $v=w\sin\theta$ and $\mathbf{k}=(k\cos\alpha,\ k\sin\alpha)$. Then

$$I=\int_0^{\infty}\int_0^{2\pi} e^{ikw\cos(\alpha-\theta)}\sin\theta[1-\exp(-q^2w^2)]\,dw\,d\theta. \tag{A.2}$$

Let $\phi=\theta-\alpha$ and the integral over θ becomes

$$\sin(\alpha)\int_0^{2\pi} e^{ikw\cos(\phi)}\cos(\phi)\,d\phi=\frac{2\pi\sin(\alpha)}{ik}\frac{\partial}{\partial w}J_0(kw)\,dw, \tag{A.3}$$

which vanishes if $k=0$. Substituting (3) into (2) we obtain

$$I=\frac{2\pi\sin(\alpha)}{ik}\int_0^{\infty}[1-\exp(-q^2w^2)]\frac{\partial}{\partial w}J_0(kw)\,dw. \tag{A.4}$$

Integration by parts (assuming $k\neq 0$) gives

$$I=-\frac{4\pi\sin(\alpha)}{ik}q^2\int_0^{\infty}J_0(kw)\,we^{-q^2w^2}\,dw. \tag{A.5}$$

The remaining integral is known and the final result is

$$I=-\frac{2\pi\sin(\alpha)}{ik}\exp(-k^2/(4q^2)). \tag{A.6}$$

The Δv_y contribution requires an integral which is similar to I except that the n in the numerator is replaced by u. The resulting integral is

$$-\frac{2\pi\cos(\alpha)}{ik}\exp(-k^2/(4q^2)). \tag{A.7}$$

Note that $\sin\alpha=\pi p/(bk)$ and $\cos(\alpha)=2\pi s/(ak)$.

APPENDIX B: EXPRESSIONS FOR THE VELOCITY CONTRIBUTIONS

The contributing particle is at (x, y). The particle whose velocity we require is at (x_p, y_p). Let

$$\begin{aligned} w_{nm}^2(\pm) &:= (y_p\pm y-2nb)^2+(x_p-x-ma)^2,\\ g_{nm}(\pm) &:= s_{nm}(\pm)/\omega_{nm}^2(\pm), \end{aligned}$$

where s_{nm} is the smoothing function for the w_{nm} (see Section 5, where, in (5.4), r is replaced by ω_{nm}),

$$E_{nm}(\pm):=\exp(-q^2\omega_{nm}^2(\pm)).$$

Then with

$$\begin{aligned} \mathbf{r}_p-\mathbf{r}' &:= (x_p-x,\ y_p+y)\\ \mathbf{r}_p-\mathbf{r} &:= (x_p-x,\ y_p-y)\\ \mathbf{k} &= 2\pi\left(\frac{s}{a},\frac{p}{(2b)}\right), \end{aligned}$$

we find

$$\begin{aligned} \Delta v_x = {} & -\frac{K}{2\pi}\sum_m\sum_n (y_p-y-2nb)\,g_{nm}(-)\,E_{nm}(-)\\ & +\frac{K}{2\pi}\sum_m\sum_n (y_p+y-2nb)\,g_{nm}(+)\,E_{nm}(+)\\ & -\left(\frac{K}{2\pi}\right)\left(\frac{\pi^2}{ab^2}\right)\sum_p\sum_s [\sin\mathbf{k}\cdot(\mathbf{r}_p-\mathbf{r}')\\ & -\sin\mathbf{k}\cdot(\mathbf{r}_p-\mathbf{r})]\,\frac{p}{k^2}e^{-k^2/4q^2} \end{aligned}$$

$$\Delta v_y = \frac{K}{2\pi} \sum_m \sum_n (x_p - x - ma)\, g_{nm}(-)\, E_{nm}(-)$$
$$- \frac{K}{2\pi} \sum_m \sum_n (x_p - x - ma)\, g_{nm}(+)\, E_{nm}(+)$$
$$- \left(\frac{K}{2\pi}\right)\left(\frac{2\pi^2}{ab}\right) \sum_p \sum_s [\sin(\mathbf{k} \cdot (\mathbf{r}_p - \mathbf{r}'))$$
$$- \sin(\mathbf{k} \cdot (\mathbf{r}_p - \mathbf{r}))] \frac{s}{k^2} e^{-k^2/4q^2}.$$

APPENDIX C

This appendix gives the interaction term in the m, n summation in terms of an expansion. For the tree code this expansion involves moments about the centre of a cell.

Let $\mathbf{r}_p$, $\mathbf{R}$ be the position of the particle and the cell centre (node), respectively. Let $\mathbf{r}_j$ be the position of a particle (either real or image) in the cell. The contribution to the velocity of p is given by

$$\mathbf{v}_p = \frac{1}{2\pi} \sum_j K_j \frac{\hat{\mathbf{z}} \times (\mathbf{r}_p - \mathbf{r}_j)}{|\mathbf{r}_p - \mathbf{r}_j|^2} \exp(-q^2\, |\mathbf{r}_p - \mathbf{r}_j|^2);$$

let $\mathbf{r}_j = \mathbf{R} + \mathbf{r}'_j$ and let $\mathbf{D} = \mathbf{r}_p - \mathbf{R} \equiv (D_x, D_y)$. Defining

$$S = \sum_j K_j; \qquad S_y = \sum_j y'_j K_j; \qquad S_{xy} = \sum_j x'_j\, y'_j K_j$$
$$S_x = \sum_j x'_j K_j; \qquad S_{xx} = \sum_j x'^2_j K_j; \qquad S_{yy} = \sum_j y'^2_j K_j$$
$$c_1 = 2\left(q^2 + \frac{1}{D^2}\right) \qquad c_2 = \frac{4}{D^4} + 2(q^2)^2 + \frac{4q^2}{D^2}$$
$$f_1 = e^{-q^2 D^2} \frac{1}{2\pi D^2} \qquad b_1 = S_{xx} D_x + S_{xy} D_y$$
$$f_2 = c_1 (D_x D_x + D_y D_y) \quad b_2 = S_{yy} D_y + S_{xy} D_x$$

and expanding about D gives $\mathbf{v}_p = (v_x, v_y)$ in the forms:

zeroth and linear terms retained,

$$v_x = -f_1 D_y (S + f_2)$$
$$v_y = +f_1 D_x (S + f_2)$$

zeroth, linear, and quadratic terms retained,

$$v_x = -f_1 (D_y (S + f_2 + F_3) - S_y - c_1 b_2)$$
$$v_y = +f_1 (S D_x + f_2 D_x - S_x),$$

where $f_3 = c_2 (b_1 D_x + b_2 D_y) - \frac{1}{2} C_1 (S_{xx} + S_{yy})$.

We give the two approximations to v_x, v_y here because it is useful to compare the advantage of higher accuracy (which does not require opening up so many nodes) with the disadvantage of extra work.

REFERENCES

1. C. Anderson and C. Greengard, On vortex methods, *SIAM J. Numer. Anal.* **22,** 417 (1985).
2. J. Barnes and P. Hut, A hierarchical $O(N \log N)$ force calculation algorithm, *Nature* **324,** 446 (1986).
3. G. Batchelor, *An Introduction to Fluid Dynamics* (Cambridge Univ. Press, Cambridge, UK, 1967).
4. A. J. Chorin, Numerical study of slightly viscous flow, *J. Fluid. Mech.* **57,** 785 (1973).
5. P. P. Ewald, *Ann. Phys.* (*N.Y*) **64,** 253 (1921).
6. L. Greengard and V. Rokhlin, A fast algorithm for particle simulation, *J. Comput. Phys.* **73,** 325 (1987).
7. L. Greengard, Potential flow in channels, *SIAM J. Sci. Stat. Comput.* **11,** 603 (1990).
8. L. Hernquist and N. Katz, Tree SPH: A unification of SPH with the hierarchical tree method, *Astrophys. J.* (*Suppl.*) **70,** 419 (1989).
9. S. Kida, Motion of an elliptic vortex in uniform shear flow, *J. Phys. Soc. Japan* **50,** 3517 (1981).
10. H. Lamb, *Hydrodynamics* (Cambridge, Univ. Press, Cambridge, UK, 1932).
11. A. Leonard, Vortex methods for flow simulation, *J. Comput. Phys.* **37,** 289 (1980).
12. J. J. Monaghan and J. C. Lattanzio, A refined particle method for astrophysical problems, *Astrophys. Appl.* **149,** 135 (1985).
13. L. M. Polvani, J. Wisdom, E. DeJong, and A. P. Ingersoll, Simple dynamical models of Neptune's great dark spot, *Science* **249,** 1393 (1990).
14. T. Sarpkaya, Computational methods with vortices, *J. Fluids Eng.* **111,** 5 (1988).
15. R. K. Smith, W. Ulrich, and G. Dietachmayer, A numerical study of tropical cyclone motion using a barotropic model, *Q.J.R. Met. Soc.* **116,** 337 (1990).