

High Order Accurate Vortex Methods with Explicit Velocity Kernels

J. T. BEALE*

Department of Mathematics, Duke University, Durham, North Carolina 27706

AND

A. MAJDA[†]

Department of Mathematics, University of California, Berkeley, California 94720

Received October 19, 1983

Vortex methods of high order accuracy are developed for inviscid, incompressible fluid flow in two or three space dimensions. The velocity kernels are smooth functions given by simple, explicit formulas. Numerical results are given for test problems with exact solutions in two dimensions. It is found that the higher order methods yield a considerably more accurate representation of the velocity field than those of lower order for moderate integration times. On the other hand, the velocity field computed by the point vortex method has very poor accuracy at locations other than the particle trajectories. © 1985 Academic Press, Inc.

INTRODUCTION

Vortex methods offer a physically natural intuitively appealing means for simulating incompressible fluid flow at high Reynolds number. The use of a system of point vortices as a computational model for fluid flow began some time ago with the work of Rosenhead [17], who calculated the roll-up of a vortex sheet. Later, more refined calculations indicated that the point vortex method could lead to chaotic behavior rather than the expected roll-up. However, it was shown by Chorin and Bernard [7] and Kuwahara and Takami [13] that the roll-up was restored if the point vortices were replaced by vortices with finite cores or “blobs” of vorticity. (In the latter paper the blobs were allowed to expand in time to mimic viscosity.) A general vortex method for simulating high Reynolds number flow has been developed by Chorin and others over the last several years [5, 6, 14]. The fluid is represented by a collection of vortex blobs in the interior and sheets near the boundary, all of which are advected according to a computed velocity field.

* Supported by NSF Grant MCS-81-01639, ONR Contract N00014-76-C-0316, and A.R.O. Grant DAAG 29-80-C-0041.

[†] Supported by NSF Grant MCS-81-02360 and ARO Grant 483964-25530. Current address: Dept. of Mathematics, Princeton University, Princeton, New Jersey 08544.

New vorticity is created at the boundary to satisfy the no-slip condition, and a random walk of the vorticity elements simulates the effect of viscosity. A survey of the different vortex methods in current use may be found in [14].

We will be concerned here with the accuracy of vortex methods for smooth, inviscid flows in free space. In this case the velocity is determined from the vorticity by convolution with a singular velocity kernel. In the vortex blob algorithms the singular kernel is modified by convolution with a smooth approximation to the delta function. The discretization of the smoothed integral results in a system of ordinary differential equations for the paths of representative particles in the fluid.

Recently, a complete, rigorous theory of the accuracy, nonlinear stability, and convergence of vortex methods has been developed for smooth fluid flow in either two or three space dimensions [11, 2, 3]. One of the implications of this theory is that the choice of the smooth delta function determines the accuracy of the vortex blob algorithm for smooth flows, and that higher order methods can be designed by proper choice of this function. In Section 1 of this paper we produce a class of modified kernels of high order accuracy in two or three space dimensions which are given by simple, explicit formulas. With these choices, the vortex blob method can be implemented with essentially no more computational effort than would be necessary if the original kernel were used. In the different context of plasma physics, similar finite cores, or shape factors, have been used for density distributions [8, 9] and the significance of higher order kernels has been noted [10].

In the same section we also describe two 3-D vortex algorithms which have been developed recently. Each method has some similarity to an earlier method of Chorin [6], but there are significant differences among all three versions. The explicit higher order kernels can easily be implemented in any of these algorithms.

To test these methods we have performed a series of numerical experiments with exact solutions of the 2-D Euler equations. The examples used are steady flows generated by a radial distribution of vorticity. Since the vorticity function can be chosen so that substantial shearing takes place as time goes on, this class of problems seems to provide a significant test for a Lagrangian method.

The results of these experiments, reported in Section 2, verify the predicted order of accuracy for second or fourth order kernels for moderate integration times and also show that the higher order kernels can yield considerably smaller errors. For longer times the higher order accuracy deteriorates, but the errors do not grow catastrophically. The accuracy at later times is greatly improved by increasing the characteristic radius of the vortex blobs. In all cases the vortex blob methods produced comparable errors for the velocity on the particle trajectories or at regularly spaced locations. For the point vortex method the results were quite different. The velocities along the particle trajectories were reasonable, although not as accurate as those given by the blob methods. However, the accuracy of the computed velocity at other locations was very poor and unpredictable. This is a serious defect of the point vortex method, since in practical problems it is important to represent the velocity accurately at arbitrary points. For example, if boundaries are present, the velocity will need to be computed at the boundary.

Earlier tests of vortex blob methods for radial patches of vorticity were conducted by Hald and Del Prete [12] and related experiments were reported in [15], [16].

1. EXPLICIT SMOOTH KERNELS

Vortex methods are based on the fact that, for incompressible flow, the velocity is determined from the vorticity by a convolution,

$$u(z, t) = (K * \omega)(z, t) = \int K(z - z') \omega(z', t) dz'. \quad (1.1)$$

(We will use notation consistent with [2, 3]. This formula is interpreted differently in two or three dimensions, see below.) In the methods of [2, 3], as in earlier work, the velocity kernel K is replaced by

$$K_\delta = K * \psi_\delta, \quad \psi_\delta(z) = \delta^{-N} \psi(z/\delta). \quad (1.2)$$

Here $N = 2$ or 3 is the space dimension and δ is a parameter to be chosen in conjunction with the inner spacing h of the particles introduced at time zero. The smoothing of the kernel by the function ψ_δ can be interpreted as the approximation of the vorticity distribution by a sum of "blobs" of prescribed shape (see [11]).

We will choose the function ψ subject to the conditions

- (i) ψ is smooth and rapidly decreasing, i.e.,

$$|D^\beta \psi(z)| \leq C_\beta (1 + |z|^2)^{-j}$$

for every multi-index β and every integer j ;

(ii) $\int \psi(z) d^N z = 1$;

(iii) $\int z^\beta \psi(z) d^N z = 0, \quad 1 \leq |\beta| \leq m - 1,$

where m is an integer.

The results of [2, 3] imply that vortex methods satisfying (i)–(iii) converge provided the relation between δ and h is properly chosen. If $\delta = h^q$ (q any fixed number with $0 < q < 1$), the error is of the order of $\delta^m = h^{mq}$, i.e., the method is essentially m th order. Our object here is to choose ψ so that K_δ has a simple expression consistent with these requirements. As we shall see, choices of ψ with $m = 2$ easily lead to choices with $m \geq 4$.

Condition (i) implies that the Fourier transform of ψ , as well as ψ itself, is smooth and rapidly decreasing. We will always take $\psi = \psi(r)$, $r = |z|$. In this case

(iii) holds by symmetry for $|\beta|$ odd, so that m may be assumed even. For radial ψ , (iii) is always satisfied with $m=2$. This set of conditions is more stringent than those in the general theory of [2, 3]. (In the earlier language we are assuming ψ to be in the class $Fe S^{-\infty, m}$.) Condition (i) can be relaxed somewhat to allow a ψ which is not very smooth at $z=0$. Indeed, our simplest choice in three dimensions has this property. With a weaker condition replacing (i), a similar convergence result holds, but q is restricted to an interval $0 < q < q_0$ for some $q_0 < 1$. (See [2] for precise statements.)

Two-Dimensional Flows

In the 2-D case the vorticity is the scalar function $\omega = u_{2,x} - u_{1,y}$. The distinguishing property of two-dimensional flows is that the vorticity is conserved along particle paths,

$$\omega_t + u \cdot \nabla \omega = 0. \quad (1.3)$$

Suppose an initial velocity field is prescribed with vorticity ω_0 nonzero only within a bounded set. To simulate the flow, we cover this set with a square grid of size h and introduce a particle at the center of each square. We take the coordinates of a typical particle to be ih , where $i = (i_1, i_2)$ is a pair of integers; the i th particle is assigned the vorticity $\omega_i = \omega_0(ih)$. To compute approximate paths of particles, we discretize (1.1), with K replaced by K_δ , remembering (1.3), and arrive at the system of ordinary differential equations

$$\dot{\tilde{z}}_i = \sum_j K_\delta(\tilde{z}_i - \tilde{z}_j) \omega_j h^2, \quad \tilde{z}_i(0) = ih. \quad (1.4)$$

The area-preserving property of incompressible flow is used implicitly here. Once the \tilde{z}_i 's have been determined, an expression for the velocity can be obtained by setting

$$\tilde{u}^h(z, t) = \sum_j K_\delta(z - \tilde{z}_j) \omega_j h^2.$$

To apply this method it is best to have an explicit formula for K_δ . If G is the Green's function for $-\nabla^2$, $G(z) = -(2\pi)^{-1} \log r$, then with $z = (x, y)$

$$K(z) = (\partial_y, -\partial_x) G = \frac{(-y, x)}{2\pi r^2}.$$

A natural choice of ψ is the Gaussian $\psi^{(2)}(r) = e^{-r^2}/\pi$. The necessary conditions (i)–(iii) are satisfied with $m=2$. If $K_\delta = K * \psi_\delta$, then

$$K_\delta = (\partial_y, -\partial_x) G_\delta, \quad G_\delta = G * \psi_\delta. \quad (1.5)$$

Since ψ is radial, G_δ is also, and

$$\nabla^2 G_\delta = \nabla^2(G * \psi_\delta) = -\psi_\delta \quad \text{or} \quad \frac{1}{r} D_r(r D_r G_\delta) = -\psi_\delta(r) = -\frac{1}{\pi \delta^2} e^{-r^2/\delta^2}.$$

After integration we have

$$D_r G_\delta = \frac{1}{2\pi r} (e^{-r^2/\delta^2} - 1).$$

The constant of integration is determined by the fact that G_δ must be smooth. The corresponding velocity kernel K_δ may now be found from (1.5):

$$K_\delta^{(2)}(z) = \frac{(-y, x)}{2\pi r^2} (1 - e^{-r^2/\delta^2}).$$

The superscript (2) has been inserted to indicate the order of the kernel.

Next we will obtain a fourth order kernel by choosing $\psi = \psi^{(4)}$ as a combination of two Gaussians with different scalings,

$$\psi^{(4)}(r) = c_1 \psi^{(2)}(r) + c_2 \psi^{(2)}(r/a)$$

where a is arbitrary except that $a \neq 1$. To satisfy condition (ii) we must have

$$c_1 + a^2 c_2 = 1.$$

This leaves us with one constraint to impose conditions. Because of symmetry, condition (iii) will hold with $m = 4$ provided

$$\int \psi^{(4)}(r) r^2 \cdot r \, dr = 0.$$

This in turn holds if

$$c_1 + a^4 c_2 = 0,$$

and the two equations determine $\psi^{(4)}$ in terms of a . We can now find $K_\delta^{(4)}$ just as in the previous case:

$$K_\delta^{(4)}(z) = \frac{(-y, x)}{2\pi r^2} (1 - c_1 e^{-r^2/\delta^2} - c_2 a^2 e^{-r^2/a^2 \delta^2}).$$

For example, the choice $a^2 = 2$ leads to

$$\begin{aligned} K_\delta^{(4)}(z) &= \frac{(-y, x)}{2\pi r^2} (1 - 2e^{-r^2/\delta^2} + e^{-r^2/2\delta^2}) \\ &= \frac{(-y, x)}{2\pi r^2} (1 - e^{-r^2/2\delta^2})(1 + 2e^{-r^2/2\delta^2}). \end{aligned}$$

It should be clear that higher order kernels can be constructed by adding further terms with Gaussians of different scalings in the expression for ψ . A typical sixth order kernel is

$$K_{\delta}^{(6)}(z) = \frac{(-y, x)}{2\pi r^2} \left(1 - \frac{8}{3} e^{-r^2/\delta^2} + 2e^{-r^2/2\delta^2} - \frac{1}{3} e^{-r^2/4\delta^2} \right).$$

Of course, care must be taken in evaluating any version for r small, since the factor due to the smoothing vanishes at $z = 0$.

Even simpler high order kernels can be obtained by choosing ψ in the form $\psi(r) = P(r) e^{-r^2}$, where P is a polynomial in even powers of r . Then as in the argument above

$$\begin{aligned} rD_r G_{\delta}(r) &= -\delta^{-2} \int rP(r/\delta) e^{-r^2/\delta^2} dr \\ &= (2\pi)^{-1} \{Q(r/\delta) e^{-r^2/\delta^2} - Q(0)\}, \end{aligned}$$

where Q is another even polynomial of the same degree. For condition (ii) to hold we must have $rD_r G_{\delta} \rightarrow -1/2\pi$ as $r \rightarrow \infty$, so that $Q(0) = 1$. Finally

$$D_r G_{\delta}(z) = \frac{1}{2\pi r} (Q(r/\delta) e^{-r^2/\delta^2} - 1),$$

and according to (1.5)

$$K_{\delta}(z) = K(z)(1 - Q(r/\delta) e^{-r^2/\delta^2}).$$

To satisfy the moment conditions (iii) we need to have

$$\int_0^{\infty} r^{2j} P(r) e^{-r^2} r dr = 0, \quad 1 \leq j \leq (m-2)/2,$$

or after integrating by parts,

$$\int_0^{\infty} r^{2j-1} Q(r) e^{-r^2} dr = 0.$$

The moment conditions are thus reduced to linear equations for the coefficients of Q . For a kernel of order m there are $(m-2)/2$ conditions, and a polynomial of degree $m-2$ is sufficient. The first few such kernels are

$$K^{(m)}(z) = \frac{(-y, x)}{2\pi r^2} \{1 - Q^{(m)}(r/\delta) e^{-r^2/\delta^2}\}$$

with

$$\begin{aligned} Q^{(4)}(r) &= 1 - r^2, \\ Q^{(6)}(r) &= 1 - 2r^2 + r^4/2, \\ Q^{(8)}(r) &= 1 - 3r^2 + 3r^4/2 - r^6/6. \end{aligned}$$

The polynomials which appear are Laguerre polynomials of r^2 , normalized so that the constant term is 1.

Three-Dimensional flows

In the three-dimensional case the vorticity $\omega = \nabla \times u$ is a vector quantity, and the velocity is expressed in the form (1.1) by the Biot–Savart law. We will approximate the particle paths by an analogue of (1.4), but now the vorticity must be updated as well as the position. There are two natural ways to describe the evolution of vorticity, and convergent methods can be based on either. The more familiar form is the direct generalization of (1.3),

$$\omega_t + u \cdot \nabla \omega = \omega \cdot \nabla u. \quad (1.6)$$

The method of [2], however, uses a Lagrangian expression for the vorticity due to Cauchy. Let α be the Lagrangian position and $\Phi^t: \alpha \rightarrow z$ the coordinate mapping induced by the flow. Then

$$\omega(z, t) = \nabla \Phi^t(\alpha) \cdot \omega_0(\alpha) \quad (1.7)$$

where ω_0 is the initial vorticity and $z = \Phi^t(\alpha)$. Thus the vorticity is carried along particle paths but distorted by the Jacobian matrix of the flow. The gradient with respect to the Lagrangian variable α of the Jacobian matrix can easily be implemented by a difference operator on the initial grid applied to z . We are thus led to the system of ordinary differential equations for the particle trajectories $\{\tilde{z}_i\}$,

$$\dot{\tilde{z}}_i = \sum_j K_{ij}(\tilde{z}_i - \tilde{z}_j) \tilde{\omega}_j(t) h^3, \quad \tilde{z}_i(0) = ih \quad (1.8)$$

with

$$\tilde{\omega}_i = \nabla_i^h \tilde{z}_i \cdot \omega_0(ih). \quad (1.9)$$

Here ∇^h is an anti-symmetric difference operator whose order is at least the intended order of accuracy. (See [2] for details.) In [2] we evaluated the vorticities $\tilde{\omega}_i$ by solving a spatially discretized version of the time derivative of (1.7). C. Greengard has pointed out that the simpler formulation just described is equivalent to our original one.

An alternative method can be based on (1.6). In this case we solve a coupled

system of ordinary differential equations for the $\{\tilde{z}_i\}$ and $\{\tilde{\omega}_i\}$, consisting of (1.8) and a discrete version of (1.6):

$$\frac{d\tilde{\omega}_i}{dt} = \tilde{\omega}_i \cdot \sum \nabla K_\delta(\tilde{z}_i - \tilde{z}_j) \tilde{\omega}_j h^3.$$

Here the gradient is to be computed analytically once an explicit expression for K_δ is known. Such a method was proposed by C. Anderson. It can be shown to converge provided the order of accuracy of the smooth kernel is at least 4. The convergence proof will be given in [4]. The first method has the virtue of simplicity and requires less computation than the second. However, the second has the important advantage that it does not make explicit reference to the initial grid, i.e., no information has to be carried along about the original particle configuration. Thus the second method could more easily be combined with methods for representing boundary effects. The method used by Chòrín [6] is similar to these two but not identical with either.

The three-dimensional realization of (1.1) is

$$u = K * \omega = \nabla \times (G * \omega),$$

where G is the Green's function for $-\nabla^2$, $G = 1/4\pi r$, and the convolution is componentwise. As before we set $G_\delta = G * \psi_\delta$ and $K_\delta = K * \psi_\delta$. It is easy to see that

$$(K_\delta * \omega)(z) = \int \frac{\partial G_\delta}{\partial r}(|z - z'|) \frac{z - z'}{|z - z'|} \times \omega(z') dz'$$

or more briefly

$$K_\delta(z) = \frac{\partial G_\delta}{\partial r}(|z|) \frac{z}{|z|}.$$

Thus K_δ will have a simple expression provided $\partial G_\delta/\partial r$ does so.

In this case it is less clear how to choose ψ than in the two-dimensional case, and it is best to proceed in the opposite direction from before. For simplicity we assume at first that $\delta = 1$. We suppose that

$$\frac{\partial G_1}{\partial r} = -\frac{f(r)}{4\pi r^2}$$

with f to be determined; this form is convenient since we expect

$$\frac{\partial G_1}{\partial r} \sim \frac{\partial G}{\partial r} = -\frac{1}{4\pi r^2} \quad \text{as } r \rightarrow \infty.$$

Then

$$-\psi = \nabla^2 G_1 = r^{-2} D_r \{r^2 D_r G_1\}$$

so that

$$\psi(r) = \frac{f'(r)}{4\pi r^2}. \quad (1.10)$$

We can now list the conditions which must be satisfied by f so that conditions (i)–(iii) hold for ψ . Since ψ should be smooth we require

(C1) $f'(r)$ is a smooth function of r^2 ;

(C2) $f(r) = 0(r^3)$ as $r \rightarrow 0$.

It is easily seen that the total weight of ψ is $f(\infty)$, and our last condition is therefore

(C3) $f(r) \rightarrow 1$ as $r \rightarrow \infty$, and f' is rapidly decreasing.

If (C1)–(C3) hold, then ψ , defined by (1.10), satisfies (i)–(iii) with $m = 2$. Two choices of f meeting our requirements are

$$f(r) = 1 - e^{-r^3}, \quad f(r) = \tanh r^3,$$

corresponding to $\psi(r) = (3/4\pi) e^{-r^3}$ and $\psi(r) = (3/4\pi) \operatorname{sech}^2 r^3$, respectively. The first choice is simpler and is analogous to the Gaussian function in two dimensions. Actually, in this case (C1) does not hold in the strictest sense at the origin because $f(r)$ has terms r^6 and higher in the Taylor expansion. However, the general theory of [2, 3] applies to this choice, and we do not expect the difference to be significant in practice. Having chosen f , we define ψ_δ from (2) and reverse our steps to find

$$\frac{\partial G_\delta}{\partial r} = -\frac{f(r/\delta)}{4\pi r^2}$$

so that

$$K_\delta(z) = -\frac{1}{4\pi r^3} f\left(\frac{r}{\delta}\right) z.$$

The kernels just constructed are second order accurate with respect to δ . If f' is arbitrarily smooth, as is true for $f(r) = \tanh r^3$, then we have convergence with $\delta = h^q$, $q = 1 - \varepsilon$, and the errors are essentially second order in h as well. For the “cubic Gaussian,” the results of [2] require $q < \frac{5}{8}$, so that the order of convergence in h is $\frac{5}{4} - \varepsilon$. (See Theorem 1 in [2] and the remarks following Theorem 2; the number M is 6 in this case. The predicted order of convergence is not sharp and can be improved at least to $\frac{12}{7} - \varepsilon$.)

To obtain kernels with $m = 4$ we can combine two different scalings as before. Let $\psi^{(4)}(r) = c_1 \psi(r) + c_2 \psi(r/a)$, where ψ is one of the two choices specified above. The

conditions that $\psi^{(4)}$ have weight one and satisfy the second order moment conditions lead to the equations

$$\begin{aligned}c_1 + a^3 c_2 &= 1 \\c_1 + a^6 c_2 &= 0.\end{aligned}$$

Again reversing the steps we have

$$\begin{aligned}\psi_\delta^{(4)}(r) &= \frac{1}{4\pi r^2 \delta} \left\{ c_1 f' \left(\frac{r}{\delta} \right) + c_2 a^2 f' \left(\frac{r}{a\delta} \right) \right\} \\ \frac{\partial G_\delta}{\partial r}(r) &= -\frac{1}{4\pi r^2} \left\{ c_1 f \left(\frac{r}{\delta} \right) + c_2 a^3 f \left(\frac{r}{a\delta} \right) \right\} \\ K_\delta(z) &= -\frac{1}{4\pi r^3} \left\{ c_1 f \left(\frac{r}{\delta} \right) + c_2 a^3 f \left(\frac{r}{a\delta} \right) \right\} z.\end{aligned}$$

For example, with $f(r) = 1 - e^{-r^3}$ or $\tanh r^3$, it would be convenient to choose $a^{-3} = 2$. In the first case, this gives

$$K_\delta(z) = -\frac{1}{4\pi r^3} \{ 1 - c_1 e^{-r^3/\delta^3} - c_2 e^{-2r^3/\delta^3} \} z$$

so that only one exponentiation is necessary.

An alternative method can be used to produce kernels of fourth order in δ from the one of second order already obtained. Suppose a function $f(r)$ has been found as above meeting conditions (C1)–(C3). We will choose

$$f_4(r) = c_1 f(r) + c_2 r f'(r) \quad (1.11)$$

with appropriate constants and check that the corresponding kernel

$$K_\delta^{(4)}(z) = -\frac{1}{4\pi r^3} f_4 \left(\frac{r}{\delta} \right) z$$

is fourth order.

If f satisfies (C1) and (C2), then f_4 does also, and (C3) will hold provided $c_1 = 1$. We need to impose the moment condition (iii) with $|\beta| = 2$. The correspondence between f_4 and $\psi^{(4)}$ is given by (1.10), and we can convert the condition on $\psi^{(4)}$ to a similar one for f_4 ,

$$\begin{aligned}0 &= 4\pi \int_0^\infty \psi^{(4)}(r) r^2 \cdot r^2 dr = \int_0^\infty f_4'(r) r^2 dr \\ &= \int_0^\infty g_4'(r) r^2 dr = -2 \int_0^\infty g_4(r) r dr,\end{aligned}$$

where $g_4(r) = f_4(r) - 1$, so that $g_4(\infty) = 0$. If we now substitute from (1.11) with $f = 1 + g$ and $c_1 = 1$, our condition becomes

$$\begin{aligned} 0 &= \int_0^\infty \{g(r)r + c_2 g'(r)r^2\} dr \\ &= (1 - 2c_2) \int_0^\infty g(r)r dr, \end{aligned}$$

after integrating by parts. This holds if $c_2 = \frac{1}{2}$, and therefore the choice

$$f_4(r) = f(r) + \frac{1}{2}rf'(r)$$

satisfies all our requirements.

We can apply this result directly to the two choices of f made above. If $f(r) = 1 - e^{-r^3}$, then

$$f_4(r) = 1 + (-1 + \frac{3}{2}r^3) e^{-r^3}.$$

For $f(r) = \tanh r^3$, we have

$$f_4(r) = \tanh r^3 + \frac{3}{2}r^3 \operatorname{sech}^2 r^3$$

or

$$f_4(r) = T + \frac{3}{2}r^3(1 - T^2), \quad T = \tanh r^3.$$

Again, either of these two methods can be extended to produce higher order kernels. Similar arguments could be used in the 2-D case and would lead to the Gaussian kernels found before and additional ones as well, all of the form

$$K_\delta(z) = \frac{(-y, x)}{2\pi r^2} f\left(\frac{r}{\delta}\right).$$

For example, in analogy with the second choice above, the function

$$f(r) = \tanh r^2$$

gives a second order kernel. As in the 3-D case we can add a term of the form $rf'(r)$ to obtain a fourth order kernel

$$\begin{aligned} f_4(r) &= \tanh r^2 + r^2 \operatorname{sech}^2 r^2 \\ &= T + r^2(1 - T^2), \quad T = \tanh r^2. \end{aligned}$$

Although convergence proofs for vortex methods have been given only in the circumstances of [2, 3], the smooth kernels obtained here could be applied to other versions of these methods, for example, the three-dimensional method of [6]. It should be noted, however, that for a vorticity distribution on a set of lower dimen-

sion, such as an interface between two potential flows, the formulas for modified kernels are somewhat different. Smooth kernels for this case will be developed in other work [1].

2. NUMERICAL EXPERIMENTS FOR TWO-DIMENSIONAL FLOWS

Here we describe the result of numerical experiments which illustrate the higher order (super-quadratic) accuracy of the explicit kernels derived in the first section for two dimensional flows and moderate integration times. We also report on calculations which illustrate the large inaccuracy of the point vortex method for smooth fluid flows. In addition, we describe numerical results on the behavior of all these vortex methods for long time intervals of integration. Numerical experiments describing the accuracy of a variety of low-order accurate vortex methods for moderate integration times are given by Hald and Del Prete in [12], while the work in [15] tests some aspects of the accuracy of vortex methods for inviscid shear layers.

A useful class of test problems can be obtained by choosing a radial distribution of vorticity, $\omega = \omega(r)$. The corresponding solution of the 2-D Euler equations is a rotating steady flow with a simple exact expression for the velocity field,

$$u(x, y) = (u_1, u_2) = \frac{1}{2\pi r^2} (-y, x) \int_0^r s\omega(s) ds. \quad (2.1)$$

The particle paths are circles about the origin. Since the vorticity distribution is arbitrary, the angular velocity can vary with the radius, and substantial shearing can take place as time goes on. Thus we have a natural way to test the accuracy of the method in a highly distorted geometry.

In the numerical studies to be described, we compared calculated solutions with the exact solution of (2.1) for several choices of $\omega(r)$. This class of test problems was also used in [12]. We always chose $\omega(r)$ as a polynomial in r for $r \leq 1$ and zero for $r > 1$ since in these cases the formulae in (2.1) become especially simple. Since the support of the vorticity $\omega(r)$ is confined to $r < 1$, we let \mathcal{U} denote the normalized mean velocity for $r < 1$, i.e.,

$$\mathcal{U}^2 = \frac{1}{\pi} \iint_{r < 1} |u|^2 dx dy = 2 \int_0^1 |u|^2 r dr. \quad (2.2)$$

To discretize the initial data, we used a square grid centered at the origin with spacing h in each coordinate direction and we took the initial particle locations to be the centers of these squares. We always assigned the exact value of the vorticity at the center of each square multiplied by h^2 as the initial vorticity distribution associated with the given particle (see the discussion above (1.4) from Section 1).

the vorticity to distribute the particles in a special configuration to minimize the absolute errors reported below.

The particle locations $\{\tilde{z}_i\}$ computed by a vortex method determine the numerical velocity field according to the expression

$$u_{\text{comp}}(z, t) = u^h(z, t) = \sum_k K_\delta(z - \tilde{z}_k) \omega_k h^2$$

with (2.3)

$$u_{j,\text{comp}} = \sum_k K_\delta(\tilde{z}_j - \tilde{z}_k) \omega_k h^2$$

the associated discrete velocity field computed at the particles.

We utilized two different nondimensional ways for measuring the error. First, if N particles are used in the calculation, the mean square average error in their velocities is E_{part} with

$$(E_{\text{part}})^2 = \frac{1}{N} \sum_{j=1}^N |u_{j,\text{comp}} - u_{j,\text{exact}}|^2$$
(2.4)

and the nondimensional

$$\begin{aligned} &\text{relative mean square error in velocity} \\ &\text{at the particles is } e_{\text{part}} = E_{\text{part}}/\mathcal{U}. \end{aligned}$$
(2.5)

In the second measure of error, we monitor the computed velocity field from (2.3) along the ray $0 < x < 1$ and $y = 0$. By assuming that the error along this ray is typical (from the rotational symmetry of the exact solution this is a plausible assumption), the mean square average error over $r < 1$ is

$$\frac{1}{\pi} \cdot 2\pi \int_0^1 |u_{\text{comp}}(x, 0) - u_{\text{exact}}(x, 0)|^2 x \, dx.$$

We evaluate the above integral by sampling the computed velocity at the ten points along the ray, $x = j/10$, $j = 1, \dots, 10$, $y = 0$ and using the trapezoidal rule to determine the error E_{ray} , given by

$$E_{\text{ray}}^2 = 2 \sum_{j=1}^{10} \left| \mathbf{u}_{\text{comp}} \left(\frac{j}{10} \right) - \mathbf{u}_{\text{exact}} \left(\frac{j}{10} \right) \right|^2 \frac{j}{100} f_j,$$
(2.6)

where $f_j = 1$ for $j = 1, \dots, 9$ and $f_{10} = \frac{1}{2}$. With E_{ray} defined in (2.6), the nondimensional

$$\text{relative mean square error in ray velocity is } e_{\text{ray}} = E_{\text{ray}}/\mathcal{U}. \quad (2.7)$$

We have introduced this second more severe way of measuring the error since in most practical problems, the accuracy in velocity outside the computed particle

locations is also very important. For example, if boundaries are present the value of u_{comp} off the particles are needed to generate the irrotational flow to satisfy the boundary conditions. Furthermore, in many problems involving shear layers, the propagation of secondary vortices is affected by the accuracy of the velocity field generated by the background vorticity.

We tested vortex methods using the higher order Gaussian kernels

$$K_{\delta}^{(m)}(z) = K(z) \left(1 - Q_m \left(\frac{r}{\delta} \right) e^{-r^2/\delta^2} \right)$$

with orders $m = 2, 4, 6, 8$ and described in Section 1. We denote by $m = 0$ the point vortex method, in which (2.3) and the following formula are replaced by

$$u_{\text{comp}} = \sum_k K(z - \tilde{z}_k) \omega_k h^2,$$

$$u_{j,\text{comp}} = \sum_{k \neq j} K(\tilde{z}_j - \tilde{z}_k) \omega_k h^2.$$

Numerical experiments with the other kernels developed in Section 1 through scaling are given by Perlman in [16]. With these preliminaries, we now describe the numerical results.

The Accuracy of Vortex Methods for Moderate Times

In our first experiment we chose a relatively smooth positive vorticity distribution,

$$\omega(r) = \begin{cases} (1-r^2)^3, & r < 1, \\ 0, & r > 1. \end{cases} \quad (2.8)$$

This vorticity distribution has three bounded derivatives. In Table I we report the relative velocity error at the particles, e_{part} , and the relative velocity error along the ray, e_{ray} , for a 16×16 rectangular mesh covering the initial vorticity (208 particles with nonzero vorticity) for a sample of times up to $T = 12$. The significance of this time is that

$T = 12$ corresponds approximately to the shortest period of rotations for the eddy associated with (2.8) (the eddy turn around time).

The classical fourth order Runge–Kutta method with time step $t = 1$ was used in the time discretization of the vortex methods in the numerical results described in Table I as well as all other results reported below. When we halved the time step in any case we found that there was no significant change in the accuracy indicating that the dominant errors were due to spatial discretization. In Fig. 1, we have illustrated the distortion by the exact solution generated by $\omega(r)$ in (2.8) of the

TABLE I^a

Time	m :	0-Point	2	4	6	8
		vortices				
	$C_m = \delta/h$:		1	2	2.5	2.5
$T=0$	e_{part}	0.009	0.027	0.012	0.0054	0.0015
	e_{ray}	0.021	0.028	0.012	0.0053	0.0015
$T=3$	e_{part}	0.013	0.027	0.012	0.0054	0.0016
	e_{ray}	0.128	0.028	0.012	0.0053	0.0015
$T=6$	e_{part}	0.033	0.028	0.012	0.0054	0.0017
	e_{ray}	0.159	0.028	0.012	0.0053	0.0016
$T=9$	e_{part}	0.050	0.030	0.013	0.0060	0.0046
	e_{ray}	0.111	0.029	0.012	0.0053	0.0040
$T=12$	e_{part}	0.051	0.034	0.014	0.0077	0.0086
	e_{ray}	0.366	0.033	0.014	0.0086	0.0111

^a $\omega(r) = (1 - r^2)^3$, $h = 0.125$, 208 particles with nonzero vorticity.

squares at time $t=0$ used in the initial discretization. These graphics indicate that there is already substantial distortion and shearing beginning near the time $T=6$ and increasing rapidly with time; thus, the numerical solution of this problem up to $T=12$ is a good test for the fact that errors generated by the finite cores in vortex methods ignore the distortion of the actual underlying fluid blobs.

The first important conclusion from Table I regards the point vortex method. While the velocity errors on the particles themselves stay within (perhaps acceptable) errors of 5% of the average exact velocity until $T=12$, the errors in the velocity field along a ray have grown to 36% of the average velocity by the time $T=12$! This indicates that the velocity field representation by point vortices can be wildly inaccurate even for smooth flows where there is a complete unambiguous theory of existence, uniqueness, and stability for solutions of the 2-D ideal fluid equations [2]. The significance of this observation is that the point vortex method is unreliable for problems such as those involving boundaries or smooth inviscid shear layers where one needs accurate values of the velocity off the particle trajectories. In the experiments with longer times of integration, these errors for the point vortex method continue to grow as time increases.

On the other hand, we have found in all our experiments that the smoothed core vortex methods give comparable errors for the particle velocities and the velocities along the ray. It is evident from Table I that both errors are much smaller than the particle velocity errors of the point vortex method, except in the second order case initially. For $m=2$ the ray errors range to 3.3%, while for the orders $m=4, 6, 8$, these errors vary respectively from 1.2 to 0.16% at one-half the eddy turn-around time and grow to a size from 1.5% to 1% at the time $T=12$. Furthermore, at all but the final time, the absolute errors are smaller as the order m increases.

To determinate the order of accuracy for the methods with $m=2, 4, 6, 8$, we have

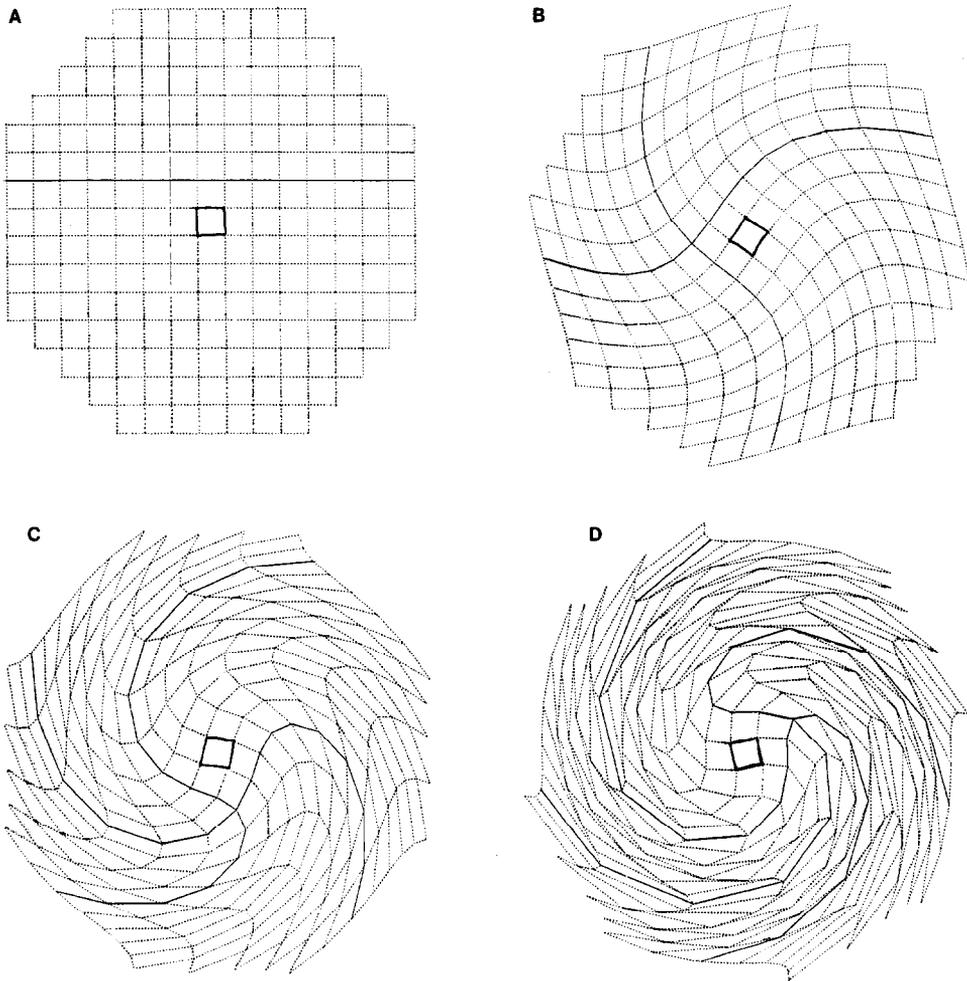


FIG. 1. The distortion of the initial rectangles used in discretization by the exact fluid flow generated by (2.8) at the successive times $T = 0, 2, 6, 10$.

also used a 20×20 rectangular grid to discretize the initial data, resulting in 316 particles with nonzero vorticity. We determined the parameter δ on this mesh by $\delta = C_m h^{3/4}$, where C_m is the number determined by the choice of δ and h in Table I. If the vorticity distribution is smooth enough and if h is small enough so that the convergence theory from [2] applies, the predicted theoretical rate of convergence for the m th order kernel is $O(h^{mq})$, where $mq \equiv \frac{3}{4}m$, $m = 2, 4, 6, 8$.

In Table II we record the actual rate of convergence computed by comparing these two calculations. We have only displayed the convergence rate measured by the relative velocity error at the particles, although the rates for the velocity along

TABLE II

Time	m :	2	4	6	8
		Theoretical order			
All times		1.5	3.0	4.5	6.0
		Computed order			
$T=0$		1.40	2.59	3.38	3.57
$T=6$		1.43	2.57	3.35	3.64
$T=12$		1.63	2.40	2.22	1.21

the ray are similar. Computed convergence rates by comparison with 16×16 and 20×20 meshes initially.

We observe that the predicted rate of convergence is within 0.1 of the computed rate for $m=2$ and within 0.6 of the actual convergence rate for $m=4$. This is

that h is small enough and $\omega(r)$ is smooth enough for the asymptotic convergence theory in [3] to apply for $m=2$ and almost so for $m=4$. However, for $m=6, 8$, the convergence rate deviates appreciably from the theoretical prediction.

This is not very surprising since for the theory in [3] to apply for $m=6, 8$ with the predicted theoretical convergence rate, $\omega(r)$ needs to have at least 14 bounded derivatives whereas $\omega(r)$ in (2.8) has only three. Nevertheless, the absolute errors with the 6th and 8th order methods are substantially smaller than the errors with $m=2$ and slightly smaller than those for $m=4$ for these moderate integration times. Furthermore, the numerical tests for $m=4, 6, 8$ clearly indicate superquadratic convergence without extensive additional computational labor beyond that required for low order accurate vortex methods. In all cases the errors with the larger number of particles were considerably smaller than those in Table I. For example, with $m=4$, e_{ray} reached a maximum of 0.008 at $T=12$.

Next, we report the results for other patches of vorticity. In one case, we chose the vorticity to change sign,

$$\begin{aligned} \omega(r) &= (1-r)^2(1-2r)(1+4r), & r < 1, \\ &= 0, & r > 1. \end{aligned} \quad (2.9)$$

The last factor is included to preserve smoothness at the origin. This vorticity distribution is less smooth and the velocity profile is not monotone for $r < 1$, so that this problem is a more severe test than the one in (2.8). In Table III, we record the results of numerical experiments using a 16×16 mesh (208 particles with nonzero vorticity) to discretize the initial vorticity distribution. Here we only record the relative error in the ray velocity.

We observe that as before the point vortex method is extremely unreliable and the errors rise to over 50% of the amplitude of the average velocity at $T=9$. This error does fall to about 10% at $T=12$; however, these errors oscillate and again

TABLE III
Relative Error in Velocity along the Rays, e_{ray}^a

	$m:$	0	2	4	6	8
Time	$\delta/h:$		1	2	2.5	2.5
$T=0$		0.036	0.073	0.059	0.036	0.012
$T=3$		0.487	0.073	0.059	0.036	0.012
$T=6$		0.187	0.073	0.059	0.036	0.012
$T=9$		0.511	0.075	0.060	0.038	0.015
$T=12$		0.096	0.074	0.056	0.032	0.019

^a $\omega(r) = (1-r)^2(1-2r)(1+4r)$ and $h = 0.125$.

grow to 55% by $T = 13$. The absolute errors for $m = 2, 4, 6, 8$ in this case behave as in Table I; however, the magnitude of the errors is somewhat larger.

Finally, we mention that we have also done numerical experiments for the discontinuous patch of vorticity,

$$\begin{aligned}\omega(r) &= 1, & r < 1, \\ &= 0, & r > 0.\end{aligned}$$

The particles in this case rotate as a rigid body, and most of the error is due to the crude initial discretization with a rectangular grid for this discontinuous initial data. We expected and observed no difficulties with the point vortex method in this case as a representation of the particle velocities. The higher order methods treated the discontinuous initial data in a stable fashion and all methods gave relative errors of 1–2% in the relative particle velocities with 208 particles. Negligible error growth in time was observed. We mention this experiment here primarily to point out that the vortex methods using higher order kernels computed this discontinuous initial data in a stable accurate fashion even though the theory from [3] does not apply to discontinuous initial data.

The Accuracy of Vortex Methods for Large Integration Times

With the test problems defined in (2.8) and (2.9) we report the results of large time integration of the vortex methods with $m = 4, 8$ up to approximately 4.5 eddy turn-around times, i.e., $T = 54$ —in fact, if one were to attempt a graphical display of the distortion of the initial grid up to this analogous to the displays in Fig. 1, the grid squares would be distorted beyond recognition. For (2.8) we used a 16×16 grid with 208 particles with nonzero vorticity as before. In Table IV we record the relative error in the velocity at the particles at intervals of one eddy turn-around time. For comparison, we also display in this table large time runs with a very simple rezoning strategy which we describe in detail later.

Before discussing the results presented in Table IV, we mention a computational trend which we have not tabulated. With any of the smooth velocity kernels with

TABLE IV
Relative Error in Velocity at the Particles, e_{part}^a

	$m:$	4	8	8 rezoned
Time	$C_m = \delta \cdot h:$	2.5	4	2
$T = 0$		0.026	0.0148	0.00055
$T = 12$		0.026	0.0149	0.0014 (276) ^b
$T = 24$		0.042	0.0312	0.0021 (317) ^b
$T = 36$		0.028	0.0183	0.0030 (376) ^b
$T = 48$		0.028	0.0183	
Maximum error in e_{part}		4.7% $0 \leq T \leq 54$	3.5% $0 \leq T \leq 54$	0.3% $0 \leq T \leq 36$

^a Long time integration with $h = 0.125$, $\omega(r) = (1 - r^2)^3$ and 208 particles initially.

^b The method in the last column uses new particles and the new number of particles is also given (in parentheses).

$m = 2, 4, 6, 8$, we have found that decreasing δ with the same h but always enforcing $\delta \geq h$ decreases the error for times up to roughly one turn-around time at the expense of larger errors for longer times. For example, a choice of δ as in Table I would lead to errors up to about 8% of the average velocity at much later times with 208 particles. Nevertheless, extremely rapid error growth in time was not observed. In particular, for $m = 4$ and $m = 8$ we have used larger values of C_m in Table IV than in Table I and the results there illustrate this trend. The reader can check that the errors at $T = 12$ are larger in Table IV for $m = 4, 8$, however, the maximum error of 3.5% for $0 \leq t \leq 54$ with $\delta = 4h$ and $m = 8$ would be more than doubled with the choice $\delta = 2.5h$. A possible explanation of this phenomenon is that the discretization of the velocity integral at large times, which is inherently poor because of the irregular particle arrangement, is improved by increasing the radius of smoothing so that the integral is less singular.

The trend regarding the deterioration of accuracy in the point vortex method described earlier for moderate times becomes even more exaggerated at larger times. We have found errors in the ray velocity of over 100% at times as small as $T = 19$. Furthermore, the error in the relative velocity at the particles for the point vortex method, while better behaved than the error measured at the rays, grows more rapidly than the error with any of the smoothed kernels.

We have also performed similar long runs for the vorticity distribution of (2.9) with a change of sign. In this case we used a 20×20 grid (316 particles). With $m = 4$ and $\delta/h = 2.5$, the maximum relative error in particle velocity for $T \leq 50$ was 7.7%; for $m = 8$ and $\delta/h = 4$, it was 5.6%. This vorticity patch fails to satisfy the classical Rayleigh stability criterion. It is unclear whether the larger errors in this case are a manifestation of instabilities in the steady flow. Further numerical experiments would be necessary to determine this.

The vorticity distribution with the form in (2.8) generates an asymptotically

stable solution of the 2-D Euler equations satisfying linearized stability criteria and the superficial reason for the deteriorating accuracy of vortex methods for large times is the neglect of core distortion. One simple remedy to overcome this difficulty is the following one: After a fixed number of time steps one takes the computed vorticity, redistributes this vorticity on a rectangular mesh, and restarts the calculation with this new vorticity distribution. The vortex method gives a natural way to implement this strategy. The computed particles \tilde{z}_j define a continuous vorticity distribution by

$$\omega_{\text{comp}}(z, t) = \sum_j \psi_j(z - \tilde{z}_j) \omega_j h^2.$$

One can evaluate this function explicitly to assign new vortices on a rectangular grid in the restarting procedure. Since the ψ_δ functions in our case are Gaussians, and do not have compact support, we only treat as nonzero those values on a rectangular mesh with the same h covering four times the initial area (32×32 in our example) with $|\omega_{\text{comp}}| > \theta_0$ where θ_0 is a given tolerance. It is very important that this rezoning strategy is not implemented very often since this most likely introduces diffusive error, and vortex methods without the rezoning have the desirable feature that they have essentially no numerical viscosity.

In the last column of Table IV, we report the results of implementing the rezoning strategy every five time steps with $\Delta t \equiv 1$ and $\theta_0 = 0.0005$. (Here the velocity errors were computed at initial particle locations rather than current ones.) This method keeps the error an order of magnitude below that given without the rezoning for times as large as $T = 36$ but it has the disadvantage that 168 additional particles were added in the course of the calculation so that the computational labor has gone up substantially. A similar redistribution procedure for the vorticity patch of (2.9) led to errors under 2% for $T \leq 40$ with the number of particles growing to 364. The authors do not advocate this rezoning strategy as a general procedure because these steady exact solutions are very special and further extensive testing is necessary; however, the results here indicate the feasibility of such an approach for long time integration with vortex methods.

CONCLUSIONS

The authors have developed simple explicit formulae for high order accurate kernels for vortex methods in two and three dimensions. A series of numerical experiments using exact steady smooth solutions of the 2-D Euler equations indicates the following facts: (1) Pure point vortex methods are unpredictably inaccurate in representing the velocity field off the particle trajectories in smooth flows. (2) For moderate integration times, the higher order smooth kernels yield considerably smaller relative errors than lower order vortex methods. The methods with higher order kernels also exhibit super-quadratic convergence under

refinement without significant increase in the computational labor when compared with low-order vortex methods. (3) For long integration times, the higher accuracy of vortex methods deteriorates without exhibiting catastrophic large-time numerical instability, and simple rezoning strategies can minimize this effect.

ACKNOWLEDGMENTS

The authors thank Alexandre Chorin, Ole Hald, Christ Anderson, and Claude Greengard for their interest in this project, their helpful suggestions, and constructive comments. The calculations reported here were done on a VAX computer at Lawrence Berkeley Laboratory.

REFERENCES

1. J. T. BEALE AND G. BAKER, to appear.
2. J. T. BEAL AND A. MAJDA, *Math. Comput.* **39** (1982), 1.
3. J. T. BEAL AND A. MAJDA, *Math. Comput.* **39** (1982), 29.
4. J. T. BEALE AND A. MAJDA, Stretching of vorticity for 3-D vortex methods, to appear.
5. A. J. CHORIN, *J. Fluid Mech.* **57** (1973), 785.
6. A. J. CHORIN, *SIAM J. Sci. Stat. Comput.* **1** (1980), 1.
7. A. J. CHORIN AND P. BERNARD, *J. Comput. Phys.* **13** (1973), 423.
8. J. DAWSON, *Rev. Mod. Phys.* **55** (1983), 403.
9. J. EASTWOOD AND R. W. HOCKNEY, "Computer Simulation Using Particles," McGraw-Hill, New York, 1981.
10. R. A. GINGOLD AND J. J. MONAGHAN, *J. Comput. Phys.* **46** (1982), 429.
11. O. HALD, *SIAM J. Numer. Anal.* **16** (1979), 726.
12. O. HALD AND V. M. DEL PRETE, *Math. Comput.* **32** (1978) 791
13. K. KAWABATA AND H. TAKAMI, *J. Phys. Soc. Japan* **34** (1972), 211.
14. A. LEONARD, *J. Comput. Phys.* **37** (1980), 289.
15. Y. NAKAMURA, A. LEONARD, AND P. SPALART, "Vortex Simulation of an Inviscid Shear Layer," AIIAA/ASME Third Joint Thermophysics, Fluids, Plasma, and Heat Transfer Conference, 1982.
16. M. PERLMAN, Ph. D. thesis, Univ. of California, Berkeley, 1984.
17. L. ROSENHEAD, *Proc. R. Soc. London Ser. A* **134** (1932), 170.