JOURNAL OF COMPUTATIONAL PHYSICS 108, 84–94 (1993)

A Deterministic Vortex Method for the Navier–Stokes Equations

GIOVANNI RUSSO

Department of Mathematics, University of L'Aquila, L'Aquila, Italy

Received March 19, 1990; revised May 8, 1992

A new deterministic particle method is presented for treating the diffusion term in the vorticity formulation of the Navier–Stokes equations. It is based on the discrete approximation of the Laplacian on an irregular grid. The particles are convected according to the velocity field and their weight evolves according to the diffusion. The general properties of the method are analyzed. The method has been implemented in two dimensions in the case of an unbounded domain. The results of the method are studied numerically by comparison with an exact solution. © 1993 Academic Press, Inc.

1. INTRODUCTION

The vorticity formulation of the incompressible Euler equations has been extensively studied [11, 5, 27, 3, 8]. This formulation has the advantage that the pressure does not appear explicitly and need not be computed in order to describe the fluid motion. The evolution of the fluid is described in a Lagrangian form, in terms of the equations of motion of fluid particles. The initial vorticity is approximated by a set of N point vortices and the velocity of each vortex is computed as a function of the position and strength of the other vortices. This leads to a system of N ordinary differential equations (the point-vortex method). For theoretical and computational purposes the point vortices are often regularized by convolving them with a smooth function (the vortex-blob method [10, 3]). This procedure gives better numerical results and convergence properties than the point-vortex method [21, 16].

A natural way of introducing the effect of diffusion in this formulation is the addition of a Wiener process to the motion of each vortex. This *random vortex method* was first introduced by Chorin [10]. Its properties have been extensively studied in the literature [26, 20]. The method is simple, easy to implement, and can be used with complicated geometries. Boundary conditions can be imposed by creating the proper amount of vorticity at the wall. Recent developments and improvements of the method are discussed in [9, 25].

This method suffers, however, from several drawbacks. The accuracy is very poor because of the random fluctuations and the error usually decreases as $1/\sqrt{N}$. There are cases in which the results are poor even for large N. This happens, for example, when there is a strong cancellation between vortices of opposite signs. In that case the net signal can be much smaller than the noise due to the fluctuations!

For these reasons it is desirable to have a deterministic way of treating the diffusion term in the Navier Stokes equations, maintaining the advantages of a vorticity formulation. Some of the previous attempts were unsuccessful, as was pointed out in [22]. An interesting formulation is the *fractional step method* proposed by Cottet *et al.* [15, 17]. The evolution of the system in a time interval Δt is obtained in two steps. In the first step the velocity of the vortices is reconstructed via a *vortex-blob method* and the particles are advected according to their velocity. In the second step each point vortex is spread into a gaussian of size $\sqrt{2\nu\,\Delta t}$. The vorticity is now a smooth function which is then approximated in terms of δ-functions centered at the same point locations, by adjusting the strength of the vortices. This procedure leads to a consistent method. In a subsequent paper the same author treats the problem of the boundary conditions [13].

Another approach has been considered in [18]. A vortex-blob method is used and the contribution of the diffusion is obtained by differentiating the *cut-off function* and approximating the diffusion term as a sum of δ-functions, by means of a quadrature formula. The method is consistent and the stability has been proven for the heat equation, provided the cutoff function has a positive Fourier transform. The rate of convergence depends on the choice of the cutoff function.

In this paper we use a different approach, which is based on a discrete approximation of differential operators on an irregular grid. The particles are advected with the fluid velocity. The diffusion equation for the vorticity is solved on the grid formed by the particles. The method is based on the discrete approximation of a Laplacian on an irregular grid and can be generalized to three dimensions.

We remark here that this is a *free Lagrangian* method, and therefore it is adaptive. This is an advantage for the

0021-9991/93 $5.00

long-time behavior. With the usual Lagrangian schemes a regridding is required after a certain time [3, 14].

The plan of the paper is the following: in the next section we review the vorticity formulation of the Navier–Stokes equations and we deduce the evolution equations for vortex strengths and positions for the numerical method; the third section describes the implementation of this method using the *Voronoi diagram* for computing the discrete operators on an irregular grid; the fourth section describes the general properties of the method; the fifth section shows numerical results; finally, in the last section we draw some conclusions.

2. DERIVATION OF THE METHOD

In this section we review the vorticity formulation of the Navier–Stokes equation in two and three dimensions. Let us consider first the two-dimensional case. The equation for the vorticity in two dimensions is given by

$$\frac{\partial \omega}{\partial t} + (\mathbf{u} \cdot \nabla)\omega = \nu \, \Delta\omega, \tag{1}$$

where $\mathbf{u} = (u_1, u_2)$ is the fluid velocity field and $\omega \equiv \partial_x u_2 - \partial_y u_1$ is the vorticity. The velocity field can be reconstructed in terms of the vorticity

$$\mathbf{u} = \int \mathbf{K}(\mathbf{x} - \mathbf{x}') \, \omega(\mathbf{x}', t) \, d\mathbf{x}', \tag{2}$$

where

$$\mathbf{K} = \frac{1}{2\pi} \frac{1}{|\mathbf{x}|^2} \begin{pmatrix} -y \\ x \end{pmatrix}.$$

We shall consider a "particle approximation" of the solution a distribution of the form

$$\omega^{(N)} = \sum_{i=1}^{N} \Gamma_i \delta(\mathbf{x} - \mathbf{x}_i), \tag{3}$$

which represents a sum of N point vortices centered at the points $\mathbf{x}_i$, each of strength Γ_i.

We shall assume that the strength and location of the point vortices depends on time and we shall derive an evolution equation for $\Gamma_i(t)$ and $\mathbf{x}_i(t)$. If we insert (3) into (1) we obtain

$$\text{l.h.s.} = \sum_{i=1}^{N} \left[\dot{\Gamma}_i \delta_{\mathbf{x}_i}(\mathbf{x}) - \Gamma_i \dot{\mathbf{x}}_i \cdot \nabla\delta_{\mathbf{x}_i}(\mathbf{x}) + \Gamma_i \mathbf{u}(\mathbf{x}) \cdot \nabla\delta_{\mathbf{x}_i}(\mathbf{x}) \right], \tag{4}$$

$$\text{r.h.s.} = \nu \sum_{i=1}^{N} \Gamma_i \, \Delta\delta_{\mathbf{x}_i}(\mathbf{x}), \tag{5}$$

where $\delta_{\mathbf{x}_i} := \delta(\mathbf{x} - \mathbf{x}_i)$. Here we denote by $\mathbf{u}_i$ a suitable reconstruction of the fluid velocity due to the point vortices. It can be obtained either by a point-vortex method,

$$\mathbf{u}_i = \sum_{j \neq i} \Gamma_j \mathbf{K}(\mathbf{x}_i - \mathbf{x}_j), \tag{6}$$

or by a vortex-blob method,

$$\mathbf{u}_i^{\varepsilon} = \sum_{j=1}^{N} \Gamma_j \mathbf{K}_{\varepsilon}(\mathbf{x}_i - \mathbf{x}_j), \tag{7}$$

where $\mathbf{K}_\varepsilon = \mathbf{K} * g_\varepsilon$ is a "mollified" kernel and the cutoff function $g_\varepsilon(\mathbf{x}) = 1/\varepsilon^2 g(\mathbf{x}/\varepsilon)$ is some smooth function with certain regularity properties [3]. Other techniques that can be used for the reconstruction of the velocity field at the particle location include the method of local corrections [2] and the vortex-in-cell method [12].

All these methods are extensively described in the context of vortex methods for the Euler equations [3, 8]. In the next section we shall briefly describe the techniques used in our implementation. Our concern now is the derivation of the evolution equations for the diffusive term. We therefore assume that $\mathbf{u}_i$ is a given functional of the position and of the strength of the vortices.

The two sides, l.h.s. and r.h.s., cannot match as identities in the δ-function and its derivatives. This means that a distribution of the form (3) cannot be a weak solution of the Navier–Stokes equations. This is obvious because the diffusion has the effect of spreading the point vortices. We look instead for an approximate solution of Eq. (1). An approximate matching of (4) and (5) is obtained by a discrete approximation of the gradient and the Laplacian of a δ-function on an irregular grid.

This approximation is interpreted in the weak sense and is obtained in the following way: let us introduce the bracket $\langle \, , \, \rangle$ that defines a distribution as a continuous linear functional on a certain function space $\mathscr{D}$. Then we have, $\forall \phi \in \mathscr{D}$,

$$\langle \delta_{\bar{\mathbf{x}}}, \phi \rangle = \phi(\bar{\mathbf{x}}),$$

$$\langle \Delta\delta_{\bar{\mathbf{x}}}, \phi \rangle = \Delta\phi(\bar{\mathbf{x}}).$$

Let us approximate the derivatives of the function $\phi(\mathbf{x})$ in $\mathbf{x}_i$ by a linear combination of values of $\phi(\mathbf{x})$ at the other points of the set:

$$\Delta\phi(\mathbf{x}_i) \approx \sum_j \beta_{ij} \phi(\mathbf{x}_j). \tag{8}$$

The coefficients β_{ij} are, of course, not uniquely defined and depend on the particular discretization of the Laplacian. They are functions of the positions of the points.

From Eq. (8) and the definition of δ it follows that

$$\langle \Delta\delta_{\mathbf{x}_i}, \phi\rangle = \Delta\phi(\mathbf{x}_i) \approx \sum_j \beta_{ij}\phi(\mathbf{x}_j) = \sum_j \beta_{ij}\langle \delta_{\mathbf{x}_j}, \phi\rangle = \left\langle \sum_j \beta_{ij}\delta_{\mathbf{x}_j}, \phi\right\rangle.$$

Therefore the discrete approximation of the Laplacian of a δ-function is given by

$$\Delta\delta_{\mathbf{x}_i}(\mathbf{x}) \approx \sum_j \beta_{ij}\delta_{\mathbf{x}_j}(\mathbf{x}). \tag{9}$$

Let us make a brief digression on the relation between particle methods and quadrature formulas. Suppose we want to solve numerically the heat equation

$$\frac{\partial f}{\partial t} = \Delta f$$

on an irregular grid defined by a set of points $S := \{\mathbf{x}_i, i = 1, ..., N\}$. We can discretize $f(\mathbf{x}, t)$ either using an approximation of its values on the grid points,

$$f_i(t) \approx f(\mathbf{x}_i, t),$$

or by giving the weights of a *particle approximation* of $f(\mathbf{x}, t)$,

$$f(\mathbf{x}, t) \approx \sum_{i=1}^{N} w_i(t)\,\delta(\mathbf{x} - \mathbf{x}_i). \tag{10}$$

The relation between w_i and f_i is the following: let $\{p_i, i = 1, ..., N\}$ denote the weights of an accurate quadrature formula whose nodes are the points $\{\mathbf{x}_i\}$; then the following approximation holds:

$$\int f(\mathbf{x})\, g(\mathbf{x})\, d\mathbf{x} \approx \sum_{i=1}^{N} f(\mathbf{x}_i)\, g(\mathbf{x}_i)\, p_i \approx \sum_{i=1}^{N} f_i\, g(\mathbf{x}_i)\, p_i. \tag{11}$$

On the other hand, Eq. (10) implies

$$\int f(\mathbf{x})\, g(\mathbf{x})\, d\mathbf{x} \approx \sum_{i=1}^{N} w_i\, g(\mathbf{x}_i). \tag{12}$$

Comparing (11) to (12) it is natural to assume the relation

$$w_i = p_i f_i. \tag{13}$$

The evolution equations for f_i are then given by

$$\dot{f}_i = \sum_j \beta_{ij} f_j, \tag{14}$$

while the evolution equations for w_i are

$$\dot{w}_i = \sum_j w_j \beta_{ji}; \tag{15}$$

that is, the system describing the evolution of the weights is the dual of the one describing the evolution of the function. Comparing (14) to (15) and making use of (13), we obtain the following relations between the weights of the quadrature formula and the coefficients of the discrete Laplacian:

$$p_i\beta_{ij} = p_j\beta_{ji} =: \tilde{\beta}_{ij},$$

where $\tilde{\beta}_{ij} = \tilde{\beta}_{ji}$. The weights p_i have a geometrical meaning: they are proportional to the area (volume in 3D) associated to each point $\mathbf{x}_i$. If the points are uniformly distributed, as in the case of a regular grid, then the weights are constant and the matrix β_{ij} is symmetric. We will derive some specific expressions for β_{ij} and p_i in the next section.

Let us return to the derivation of the evolution equations. Using the approximation (9) in (5), it is possible to match the expressions (4) and (5). The resulting evolution equations are:

$$\dot{\mathbf{x}}_i = \mathbf{u}_i^{\varepsilon}, \qquad \dot{\Gamma}_i = \nu \sum_j \beta_{ji}\Gamma_j. \tag{16}$$

Observe that the vortices are advected according to the fluid velocity and their strength changes according to diffusion. Note also that the (weak) consistency of the approximation of the diffusion term is a consequence of the consistency of the discrete approximation of the differential operator.

This method can be extended to the 3D Navier–Stokes equations. In this case ω is a vector and the equations become

$$\frac{\partial\omega}{\partial t} + (\mathbf{u}\cdot\nabla)\omega = (\omega\cdot\nabla)\mathbf{u} + \nu\,\Delta\omega. \tag{17}$$

The velocity field can be reconstructed from the vorticity. In unbounded space, for example, we have

$$\mathbf{u}(\mathbf{x}, t) = \int \mathbf{K}(\mathbf{x} - \mathbf{x}')\cdot\omega(\mathbf{x}', t)\, d\mathbf{x}',$$

where the matrix $\mathbf{K}(\mathbf{x})$ is given by

$$\mathbf{K}(\mathbf{x}) = -\frac{1}{4\pi\,|\mathbf{x}|^3}\begin{pmatrix} 0 & z & -y \\ -z & 0 & x \\ y & -x & 0 \end{pmatrix}.$$

Let us look for an approximate solution of Eq. (17) of the form

$$\omega^{(N)}(\mathbf{x}, t) = \sum_{i=1}^{N} \mathbf{\Gamma}_i(t)\, \delta(\mathbf{x} - \mathbf{x}_i(t)), \tag{18}$$

where $\mathbf{\Gamma}_i$ are 3D vectors. A consistent scheme for the Euler equations is obtained by a vortex-blob method, which is obtained by regularizing the kernel $\mathbf{K}$ with a smooth function $g_\varepsilon(\mathbf{x}) = (1/\varepsilon^3)\, g(\mathbf{x}/\varepsilon)$ [3]. The velocity field is therefore reconstructed according to

$$\mathbf{u}^\varepsilon(\mathbf{x}) = \sum_{j=1}^{N} \mathbf{K}_\varepsilon(\mathbf{x} - \mathbf{x}_j) \cdot \mathbf{\Gamma}_j,$$

where $\mathbf{K}_\varepsilon = \mathbf{K} * g_\varepsilon(\mathbf{x})$. Substituting (18) in (17) we obtain

$$\text{l.h.s.} = \sum_{i=1}^{N} [\dot{\mathbf{\Gamma}}_i \delta(\mathbf{x} - \mathbf{x}_i) + \mathbf{\Gamma}_i(\mathbf{u}(\mathbf{x}) - \dot{\mathbf{x}}_i) \cdot \nabla\delta(\mathbf{x} - \mathbf{x}_i)], \tag{19}$$

$$\text{r.h.s.} = \sum_{i=1}^{N} \left[\delta(\mathbf{x} - \mathbf{x}_i)(\mathbf{\Gamma}_i \cdot \nabla) \sum_j \mathbf{K}_\varepsilon(\mathbf{x} - \mathbf{x}_j) \cdot \mathbf{\Gamma}_j + \nu \mathbf{\Gamma}_{ii}\, \Delta\delta(\mathbf{x} - \mathbf{x}_i) \right]. \tag{20}$$

Approximating the Laplacian by (9) we can again match l.h.s. and r.h.s., obtaining the evolution equations

$$\begin{aligned} \dot{\mathbf{x}}_i &= \mathbf{u}_i^\varepsilon, \\ \dot{\mathbf{\Gamma}}_i &= (\mathbf{\Gamma}_i \cdot \nabla) \sum_j \mathbf{K}_\varepsilon(\mathbf{x}_i - \mathbf{x}_j) \cdot \mathbf{\Gamma}_j + \nu \sum_j \beta_{ij} \mathbf{\Gamma}_j. \end{aligned} \tag{21}$$

3. IMPLEMENTATION

In this section we describe the implementation of the method for 2D problems in unbounded domain. We compute the advection velocity $\mathbf{u}$ at the particle locations using a vortex-blob method [28]. We make use of the fast multipole method (FMM) [23] for the evaluation of the sum (7). For a point-vortex method, this reduces the computational cost to $O(N)$, with a constant that depends on the tolerance required. In single precision the FMM is faster than the direct evaluation for N larger than about 200.

To construct the discrete approximation of the Laplacian we make use of the *Voronoi diagram* associated with the particle positions. Given a set $\mathscr{S}$ of points in the plane, the Voronoi diagram associates each point $\mathbf{x}_i \in \mathscr{S}$ with a convex polygon P_i defined as

$$P_i = \{\mathbf{x} \in \mathbb{R}^2 : |\mathbf{x} - \mathbf{x}_i| \leqslant |\mathbf{x} - \mathbf{x}_j|,\ \forall \mathbf{x}_j \in \mathscr{S}\}.$$

Voronoi diagrams have been extensively used as a computational tool in fluid dynamics calculations [19], as well as for constructing triangulations used in finite element methods [24] (see Fig. 1). Fast algorithms have been developed for constructing these diagrams that make free Lagrangian techniques more attractive and competitive [31, 1]. In particular there are algorithms that construct Voronoi diagrams in $O(N \log N)$ operations [31] and algorithms that are parallelizable [1].

The concept of a Voronoi diagram can be extended to any number of dimensions and can be generalized in various ways. A survey on Voronoi diagrams is given in [4].

We make use of algorithms for the construction of Voronoi diagrams that have been implemented in portable Fortran by Peskin and Börgers [6, 7]. These algorithms compute the diagram in $O(N^2)$ operations, but update it in only $O(N)$ operations, under the assumption that the topology of the diagram does not undergo dramatic change. They are particularly suited for Lagrangian computations, in which the number of points is constant and the positions of the points change only slightly at each time step.

Voronoi diagrams can be used to construct discrete approximations of differential operators. Starting from the continuous relations:

$$\int_P \Delta\phi\, d\mathbf{x} = \int_{\partial P} \frac{\partial\phi}{\partial\mathbf{n}}\, ds,$$

$$\int_P \nabla \cdot \mathbf{u}\, d\mathbf{x} = \int_{\partial P} \mathbf{u} \cdot \mathbf{n}\, ds,$$

$$\int_P \nabla\phi\, d\mathbf{x} = \int_{\partial P} \phi\mathbf{n}\, ds,$$

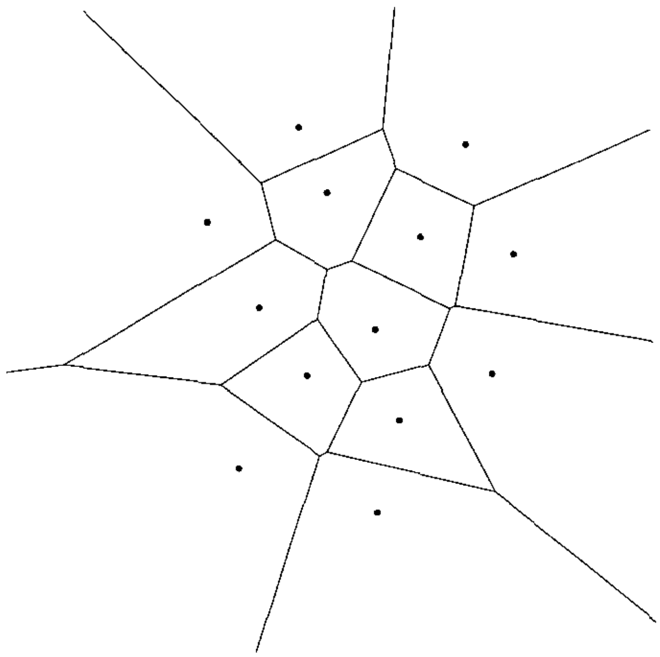

FIG. 1. Voronoi diagram associated with a distribution of points.

one defines the discrete Laplacian, L, discrete divergence, D, and discrete gradient, G, according to

$$A_i L\phi(\mathbf{x}_i) = \sum_{j\neq i} \frac{\phi_j - \phi_i}{|\mathbf{x}_j - \mathbf{x}_i|} l_{ij},$$
$$A_i D\mathbf{u}(\mathbf{x}_i) = \sum_{j\neq i} \frac{\mathbf{u}_i + \mathbf{u}_j}{2} \cdot \frac{\mathbf{x}_j - \mathbf{x}_i}{|\mathbf{x}_j - \mathbf{x}_i|} l_{ij}, \qquad (22)$$
$$A_i G\phi(\mathbf{x}_i) = \sum_{j\neq i} \frac{\phi_i + \phi_j}{2} \frac{\mathbf{x}_j - \mathbf{x}_i}{|\mathbf{x}_j - \mathbf{x}_i|} l_{ij},$$

where A_i is the area of the polygon associated with point $\mathbf{x}_i$, l_{ij} is the length of the edge corresponding to points $\mathbf{x}_i$ and $\mathbf{x}_j$, and $\phi_i := \phi(\mathbf{x}_i)$.

From the definition of the discrete Laplacian it follows that

$$\beta_{ij} = \begin{cases} \dfrac{1}{A_i} \dfrac{l_{ij}}{|\mathbf{x}_j - \mathbf{x}_i|}, & j \neq i, \\ -\dfrac{1}{A_i} \displaystyle\sum_{k\neq i} \dfrac{l_{ik}}{|\mathbf{x}_k - \mathbf{x}_i|}, & j = i. \end{cases} \qquad (23)$$

Note that $\beta_{ij} = \tilde{\beta}_{ij}/A_i$ with $\tilde{\beta}_{ij} = \tilde{\beta}_{ji}$ and A_i can be used as the weight of a quadrature formula whose nodes are at the particle location, in agreement with the discussion of the previous section. Note also that the coefficients β_{ij} are non-zero only if the points i and j are neighbors in the diagram. Therefore the number of non-zero coefficients is $O(N)$. Once the diagram is constructed the contribution of diffusion in Eqs. (16) can be evaluated in $O(N)$ operations.

Initial conditions. The values of Γ_i and $\mathbf{x}_i$ at time $t = 0$ are obtained from the initial conditions. There are several techniques for obtaining the initial positions and strenghts of the vortices [3]. We consider an initial value problem $\omega(\mathbf{x}, 0) = \omega_0(\mathbf{x})$, where $\omega_0(\mathbf{x})$ has (numerically) compact support. We distribute the points on a regular mesh of grid size h and the initial values of Γ_i are given by

$$\Gamma_i = \omega_0(\mathbf{x}_i)\, h^2. \qquad (24)$$

We place vortices with zero weight around the support of ω_0, in order to take into account the spreading of the numerical support of the vorticity. It would be desirable to have an adaptive algorithm that adds new vortices when needed and removes them where they are superfluous. An adaptive scheme for the heat equation on a triangulated mesh has been proposed in [32].

4. GENERAL PROPERTIES

In this section we shall consider some properties of the method that are direct consequences of the properties of the discrete Laplacian L.

Consistency. The discrete Laplacian L is weakly consistent to first order with the continuous Laplacian Δ, but it is not pointwise consistent. This statement is proved in the case of periodic boundary conditions in [6, 7].

Conservation of vorticity. The total vorticity $\Gamma := \sum_{i=1}^{N} \Gamma_i$ is conserved by scheme (16); from the expression for β_{ij}, it follows that

$$\frac{d}{dt} \sum_i \Gamma_i = \nu \sum_i \sum_j \beta_{ji} \Gamma_j = \sum_j \left(\sum_i \beta_{ji} \right) \Gamma_i = 0. \qquad (25)$$

Energy stability. The energy of a solution of the discrete diffusion equation on a Voronoi mesh can only decrease. To show this, let us consider the system (15), where the coefficient β_{ij} are given by (23):

$$\dot{\Gamma}_i = \sum_j \beta_{ji} \Gamma_j.$$

Let

$$\mathscr{E} = \sum_i \frac{\Gamma_i^2}{A_i}.$$

Then

$$\dot{\mathscr{E}} = \sum_i \sum_j \Gamma_i \Gamma_j \beta_{ji}/A_i < 0$$

because the matrix β_{ij}/A_j is negative definite. This statement is proved in [30].

Angular moment. The main result of this section is the proof that this scheme gives the exact evolution law for the angular moment of the vorticity. This is an important result, since it indicates that there is no spurious numerical diffusion in the method, and therefore the scheme seems suitable for slightly viscous flow.

We start by proving a remarkable property of the discrete Laplacian. For the continuous Laplacian the following relation holds:

$$\int_{\mathbb{R}^2} |\mathbf{x}^2|\, \Delta\phi(\mathbf{x})\, d\mathbf{x} = 4 \int_{\mathbb{R}^2} \phi(\mathbf{x})\, d\mathbf{x}$$

for any arbitrary function $\phi(\mathbf{x}) \in C^2(\mathbb{R}^2)$ with compact support. The same property holds for the discrete Laplacian, i.e.,

$$\sum_j^{*} |\mathbf{x}_j|^2\, L\phi(\mathbf{x}_j)\, A_j = 4 \sum_j^{*} \phi(\mathbf{x}_j)\, A_j, \qquad (26)$$

where $\phi(\mathbf{x})$ is an arbitrary function whose support does not contain nodes of the diagram associated with polygons of infinite area and the * indicates that the sum is extended only to points in the support of ϕ. In order to prove Eq. (26) we make use of the following lemma.

LEMMA 1. *Let P_k be a Voronoi polygon with finite area A_k. It then follows that*

$$c_k := \sum_{j=1}^{N} \frac{l_{jk}}{|\mathbf{x}_j - \mathbf{x}_k|} (|\mathbf{x}_j|^2 - |\mathbf{x}_k|^2) = 4A_k. \tag{27}$$

Proof. Let us perform a coordinate transformation,

$$\mathbf{x}_j = \mathbf{a} + \mathbf{z}_j,$$

where $\mathbf{a}$ is a constant vector. Then

$$c_k = \sum_{j=1}^{N} \frac{l_{jk}}{|\mathbf{z}_j - \mathbf{z}_k|} (2\mathbf{a} \cdot (\mathbf{z}_j - \mathbf{z}_k) + |\mathbf{z}_j|^2 - |\mathbf{z}_k|^2).$$

Let us choose $\mathbf{a} = \mathbf{x}_k$. Then $\mathbf{z}_k = 0$ and it follows that

$$c_k = 2 \sum_{j=1}^{N} l_{jk} \mathbf{a} \cdot \frac{\mathbf{z}_j}{|\mathbf{z}_j|} + \sum_{j=1}^{N} l_{jk} |\mathbf{z}_j|.$$

The first sum is zero, since it is the flux of the constant vector $\mathbf{a}$ across the border of the polygon. Each term in the second sum is equal to four times the area of the triangle with one edge of length l_{jk} and the opposite vertex in $\mathbf{x}_k$. The sum over j of the areas of such triangles is A_k, therefore $c_k = 4A_k$. Q.E.D.

We are now able to prove the following.

THEOREM 1. *Let $\phi(\mathbf{x}) \in C^2(\mathbb{R}^2)$ and let us assume that the support Ω of ϕ contains only points of $\mathscr{S}$ associated with polygons of finite area. Then the following property holds:*

$$\sum_j^{*} |\mathbf{x}_j|^2 L\phi(\mathbf{x}_j) A_j = 4 \sum_j^{*} \phi(\mathbf{x}_j) A_j. \tag{28}$$

Proof. From the definition of L,

$$\begin{aligned} M_2 &:= \sum_j^{*} |\mathbf{x}_j|^2 L\phi(\mathbf{x}_j) A_j \\ &= \sum_j^{*} |\mathbf{x}_j|^2 \sum_{k \neq j}^{*} \frac{\phi(\mathbf{x}_k) - \phi(\mathbf{x}_j)}{|\mathbf{x}_k - \mathbf{x}_j|} l_{jk} \\ &= \sum_j^{*} |\mathbf{x}_j|^2 \sum_{k \neq j}^{*} \frac{l_{jk}}{|\mathbf{x}_j - \mathbf{x}_k|} \phi(\mathbf{x}_k) \\ &\quad - \sum_j^{*} \sum_{k \neq j}^{*} |\mathbf{x}_j|^2 \frac{l_{jk}}{|\mathbf{x}_j - \mathbf{x}_k|} \phi(\mathbf{x}_j). \end{aligned}$$

By exchanging the order of summation and relabeling the indices in the second sum one has

$$M_2 = \sum_k^{*} \phi(\mathbf{x}_k) \sum_{j \neq k}^{*} \frac{l_{jk}}{|\mathbf{x}_j - \mathbf{x}_k|} (|\mathbf{x}_j|^2 - |\mathbf{x}_k|^2).$$

From Lemma 1 the proof follows. Q.E.D.

A property of the Navier–Stokes equations is that the second moment of the vorticity distribution is linear in time. More precisely,

$$\mathscr{L}(t) := \frac{\int_{\mathbb{R}^2} |\mathbf{x}|^2 \omega(\mathbf{x}, t)\, d\mathbf{x}}{\int_{\mathbb{R}^2} \omega(\mathbf{x}, t)\, d\mathbf{x}} = \mathscr{L}(0) + 4\nu t.$$

This property is maintained by the vortex scheme that we propose. Let us define the discrete second moment as

$$\sigma^2(t) := \frac{\sum_{i=1}^{N} \Gamma_i(t)\, |\mathbf{x}_i(t)|^2}{\sum_{i=1}^{N} \Gamma_i(t)}, \tag{29}$$

and let us consider the system of Eqs. (16), where a vortex-blob method is used in the computation of the velocity,

$$\begin{aligned} \dot{\mathbf{x}}_i &= \mathbf{u}_i^{\varepsilon}, \\ \dot{\Gamma}_i &= \nu \sum_j \beta_{ji} \Gamma_j. \end{aligned} \tag{30}$$

The conservation property of the scheme is expressed by the following.

THEOREM 2. *For the scheme* (30) *the second moment satisfies the relation*

$$\sigma^2(t) = \sigma(0)^2 + 4\nu t. \tag{31}$$

Proof. First we observe that from (25), $\Gamma := \sum_{i=1}^{N} \Gamma_i$ is constant. Then let us take the time derivative of σ^2,

$$\frac{d\sigma^2}{dt} = \frac{1}{\Gamma} \sum_{i=1}^{N} \dot{\Gamma}_i |\mathbf{x}_i|^2 + \frac{2}{\Gamma} \sum_{i=1}^{N} \Gamma_i \mathbf{x}_i \cdot \mathbf{u}_i^{\varepsilon}. \tag{32}$$

From the second equation of (30) we have

$$\sum_{i=1}^{N} \dot{\Gamma}_i |\mathbf{x}_i|^2 = \nu \sum_{i=1}^{N} |\mathbf{x}_i|^2 L\phi(\mathbf{x}_i) A_i,$$

with $\phi(\mathbf{x}_i) = \Gamma_i / A_i$. From Theorem 1 it follows that

$$\sum_{i=1}^{N} |\mathbf{x}_i|^2 L\phi(\mathbf{x}_i) A_i = 4 \sum_{i=1}^{N} \phi(\mathbf{x}_i) A_i = 4 \sum_{i=1}^{N} \Gamma_i = 4\Gamma;$$

therefore

$$\sum_{i=1}^{N} \dot{\Gamma}_i |\mathbf{x}_i|^2 = 4\nu\Gamma. \tag{33}$$

We now prove that for a vortex-blob method,

$$B := \sum_{i=1}^{N} \Gamma_i \mathbf{x}_i \cdot \mathbf{u}_i^\varepsilon = 0. \tag{34}$$

The velocity $\mathbf{u}^\varepsilon$ is given by

$$\mathbf{u}_i^\varepsilon = \sum_{j=1}^{N} \mathbf{K}_\varepsilon(\mathbf{x}_i - \mathbf{x}_j)\, \Gamma_j.$$

Therefore one finds

$$B = \sum_{i,j} \Gamma_i \Gamma_j \mathbf{x}_i \cdot \mathbf{K}_\varepsilon(\mathbf{x}_i - \mathbf{x}_j)$$

$$= \sum_{i,j} \Gamma_j \Gamma_i \mathbf{x}_j \cdot \mathbf{K}_\varepsilon(\mathbf{x}_j - \mathbf{x}_i).$$

For a radially symmetric core (i) $\mathbf{K}_\varepsilon(\mathbf{x}) = -\mathbf{K}_\varepsilon(-\mathbf{x})$ and (ii) $\mathbf{x} \cdot \mathbf{K}_\varepsilon(\mathbf{x}) = 0$. Making use of (i) we have

$$B = -\sum_{i,j} \Gamma_i \Gamma_j \mathbf{x}_j \cdot \mathbf{K}_\varepsilon(\mathbf{x}_i - \mathbf{x}_j),$$

and summing the first expression for B it follows that

$$B = \tfrac{1}{2} \sum_{i,j} \Gamma_i \Gamma_j (\mathbf{x}_i - \mathbf{x}_j) \cdot \mathbf{K}_\varepsilon(\mathbf{x}_i - \mathbf{x}_j).$$

By property (ii), each term in the summation is zero.

Substituting (33) and (34) in (32) it follows that

$$\frac{d\sigma^2}{dt} = 4\nu,$$

which is equivalent to (31). Q.E.D.

Stability considerations. Explicit schemes for the solution of the diffusion equation on a Voronoi mesh suffer a restriction on the time step due to stability conditions. In our implementation we used a forward-Euler scheme in the diffusion step. A sufficient condition for the stability is given by [30]

$$\nu\, \Delta t \leqslant \frac{2}{\max_i \sum_{j \neq i} (1 + \sqrt{A_i/A_j})\, \beta_{ij}}.$$

This condition was obtained by application of the Gershgorin theorem to the discrete Laplacian and is valid also on a moving grid of points. When applied to the case of a square grid it reduces to the well-known condition

$$\nu\, \Delta t \leqslant \tfrac{1}{2} h^2.$$

5. NUMERICAL RESULTS

In this section we test the method by comparing the numerical results with an exact solution of the Navier–Stokes equation. We consider an initial value problem with

$$\omega_0(x, y) = \exp[-12(x^2 + y^2)].$$

The period of rotation of a particle at the origin is $T = 4\pi$. We integrate the system up to time $t_{\max} = 3T$. As a measure of the error we consider the quantity

$$e_\infty(t) = \frac{\max |\mathbf{u}_i^\varepsilon(t) - \mathbf{u}(\mathbf{x}_i, t)|}{\max |\mathbf{u}(\mathbf{x}_i, t)|},$$

where $\mathbf{u}_i^\varepsilon(t)$ is the numerical value of the velocity of particle i and $\mathbf{u}(\mathbf{x}_i, t)$ is the exact velocity. We use this norm because the velocity is the physical quantity one wants to compute and this type of error can be used both for vortex-blob methods and for point-vortex methods. We do not use the L_1 norm in the velocity error, since the velocity field is not L_1 and the numerical evaluation of the error on a finite domain would depend on the computational domain.

We use a Runge–Kutta scheme of fourth order for the convection step, and a forward-Euler scheme for the diffusion step. A step size of $\Delta t = 0.05$ is used throughout all the calculations. Except when otherwise stated, we use a

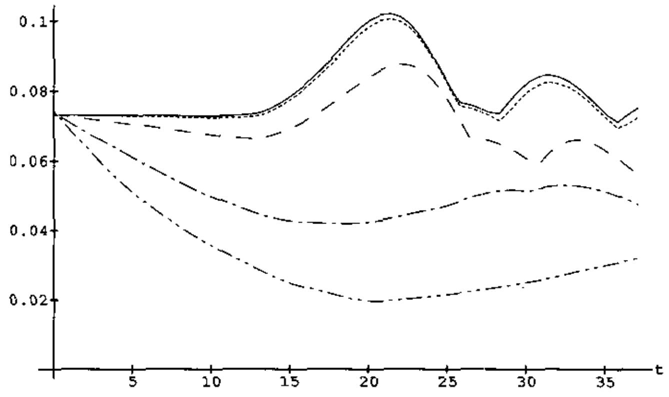

FIG. 2. Dependence of the error on the viscosity. L_∞ relative error in the velocity as a function of time; $h = 0.1$, $\varepsilon = h^q$, with $q = 0.75$. Viscosity: $\nu = 0$ (continuous line), 10^{-5} (short dash), 10^{-4} (long dash), 5×10^{-4} (dot–dash), 10^{-3} (dot–dot–dash).

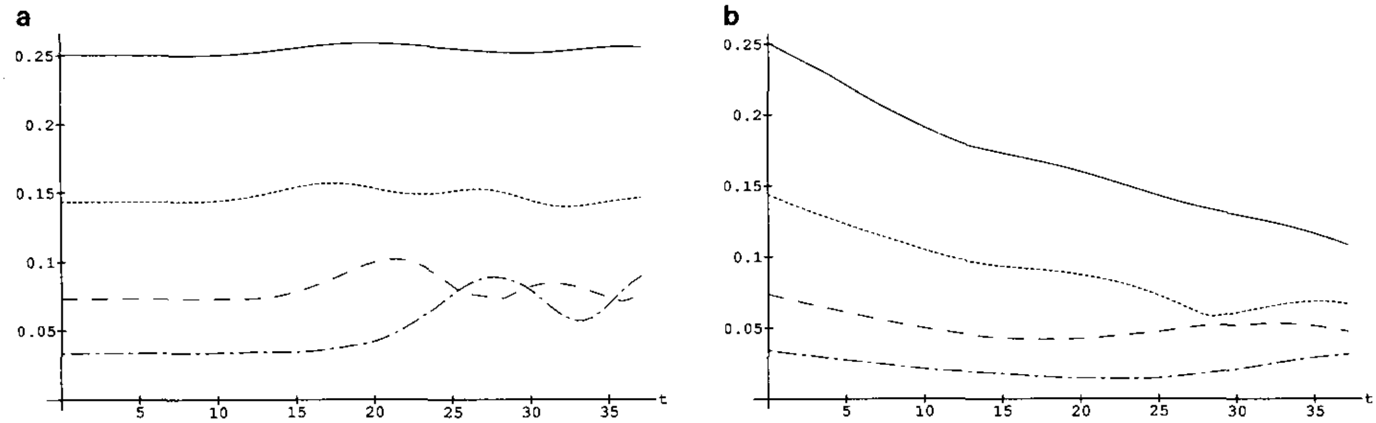

FIG. 3. Dependence of the error on the space discretization. L_∞ relative error in the velocity as a function of time: $h = 0.2$ (continuous line), 0.14 (short dash), 0.1 (long dash), 0.07 (dot–dash); $q = 0.75$. (a) Euler equations. (b) Viscosity $\nu = 5 \times 10^{-4}$.

vortex-blob method for the reconstruction of the velocity field with a Gaussian cutoff function of fourth order,

$$g(\mathbf{x}) = \frac{1}{\pi}\left(2e^{-|\mathbf{x}|^2} - \frac{1}{2}e^{-|\mathbf{x}|^2/2}\right).$$

The size of the core is related to the grid size by

$$\varepsilon = h^q.$$

A detailed numerical study of the vortex-blob method for the Euler equations is reported in [28]. We use a rather small value of q, $q = 0.75$, in most of our calculations. This value is not optimal for short times, but it provides a rather uniform accuracy over a few rotation periods for the Euler equations.

We want to study the dependence of the error of the method on the space discretization h and on the viscosity ν.

In Fig. 2 we show the error as a function of time for various values of the viscosity. The grid size is $h = 0.1$. The fact that the error decreases with time is a consequence of the large value of q. For a fixed h, the consistency error has a minimum for an optimal value of ε, which balances the discretization error and the moment error [3]. Because of stability considerations, however, it is better to use cores with a size larger than the optimal. This guarantees a uniform accuracy over longer times. Due to diffusion, the vorticity distribution spreads, and the relative size of the core becomes smaller. This is the reason the error decreases. This effect is not present when the point vortex method is used (see Fig. 5). It is evident that the method behaves like the vortex-blob method as the viscosity decreases to zero. This is obviously an essential feature if the method is to be useful for flows with small viscosity.

Figure 3a shows the error for the vortex-blob method applied to the Euler equations, while Fig. 3b shows the result of the present method with a viscosity $\nu = 0.0005$. During the simulation, the width of the vorticity distribution roughly doubles.

Figure 4 shows the result of the computation obtained

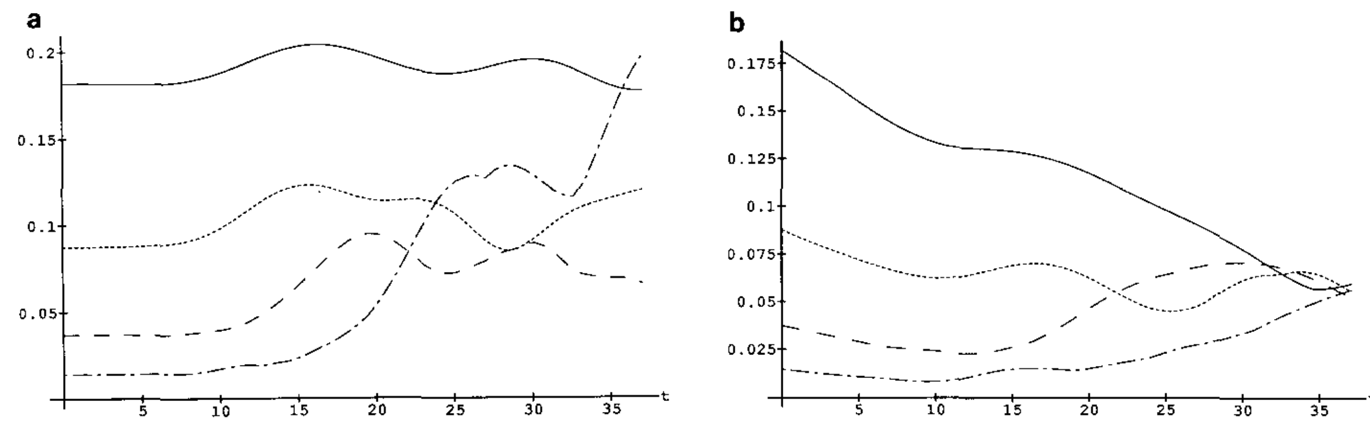

FIG. 4. Dependence of the error on the space discretization. L_∞ relative error in the velocity as a function of time: $h = 0.2$ (continuous line), 0.14 (short dash), 0.1 (long dash), 0.07 (dot–dash); $q = 0.85$. (a) Euler equations. (b) Viscosity $\nu = 5 \times 10^{-4}$.

581/108/1-7

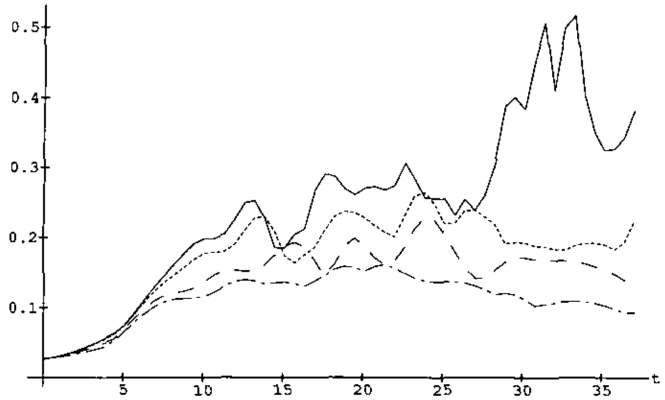

FIG. 5. Point vortex method. Dependence of the error on the viscosity. L_∞ relative error in the velocity as a function of time: $h = 0.1$. Viscosity: 0 (continuous ine); 10^{-4} (short dash); 5×10^{-4} (long dash); 10^{-3} (dot–dash).

with $q = 0.85$. As is evident, the main source of error is in the convective term. For comparison, Fig. 5 shows the result of a point-vortex method.

Comparison with the Random Vortex Method

We compare our method with the random vortex method. The initial condition is computed according to Eq. (24). During the time step Δt the circulation associated with each vorte is unchanged.

Following Roberts [30a], we compare the exact and computed value of the relative second moment of the vorticity distribution

$$\mathscr{L}(t) = \frac{\int_{\mathbb{R}^2} |\mathbf{x}|^2 \, \omega(\mathbf{x}, t) \, d\mathbf{x}}{\int_{\mathbb{R}^2} \omega(\mathbf{x}, t) \, d\mathbf{x}},$$

which is given by

$$\mathscr{L}(t) = \mathscr{L}(0) + 4\nu t.$$

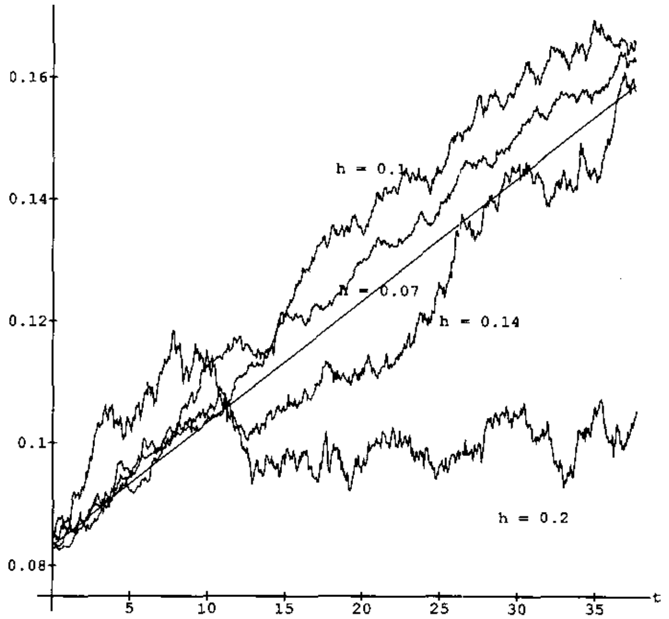

FIG. 6. Random vortex method. Angular moment vs time. Viscosity $\nu = 5 \times 10^{-4}$. Straight line, exact solution.

In the numerical scheme $\mathscr{L}(t)$ is approximated by the quantity (29)

$$\sigma^2(t) = \frac{1}{\Gamma} \sum_{i=1}^{N} \Gamma_i \, |\mathbf{x}_i|^2.$$

In Fig. 6 we show the exact and computed values of $\mathscr{L}(t)$. The angular moment computed with the random vortex method fluctuates around the exact value. The fluctuations decrease as the number of vortices increases. The deterministic scheme that we propose gives the exact evolution law for the angular moment; therefore the result is indistinguishable from the exact solution. In Fig. 7 we show the L_1 relative error in the vorticity distribution for the ran-

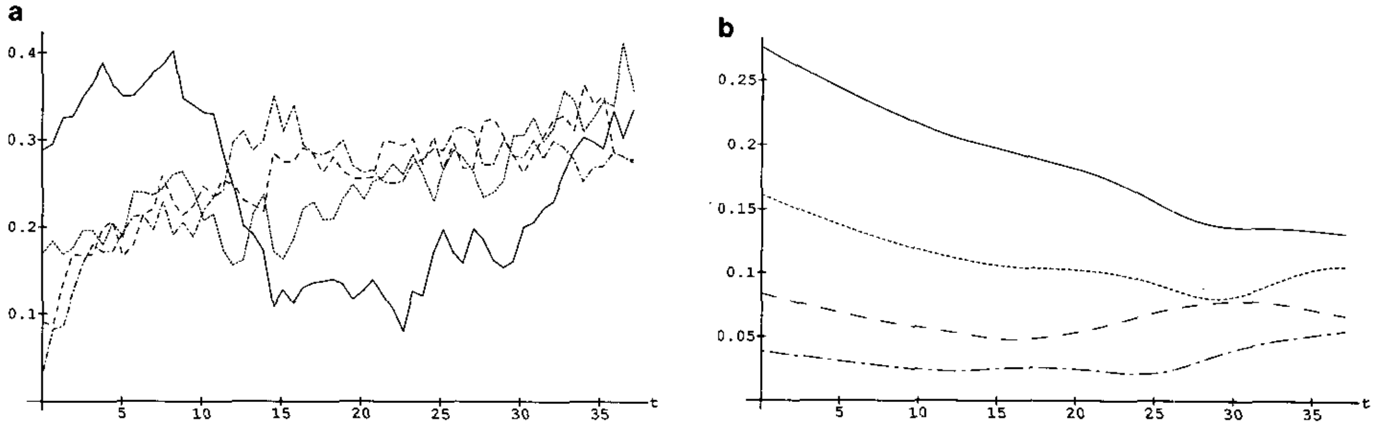

FIG. 7. L_1 relative error in the vorticity for different values of h: $h = 0.2$ (continuous line); 0.14 (short dash); 0.1 (long dash); 0.07 (dot–dash). Viscosity $\nu = 5 \times 10^{-4}$. (a) Random vortex method. (b) Deterministic method.

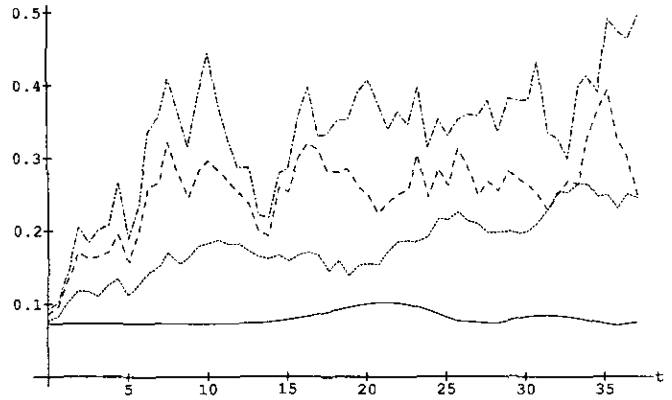

FIG. 8. L_∞ relative error in the velocity as a function of time: $h = 0.1$, $\nu = 0$ (continuous line), 10^{-4} (short dash), 5×10^{-4} (long dash), 10^{-3} (dot–dash).

dom vortex method and for the deterministic schemes. This indicator is probably not very good for determining the accuracy of the random vortex method. The random fluctuations are so large that the improvement with the increasing resolution is not evident. The use of e_∞ gives even worse results. In Fig. 8 we show the dependence of e_∞ on different values of the viscosity. From the comparison of this figure with Fig. 2, it is evident that the proposed deterministic scheme is far more accurate than the random vortex method.

Timing

In Table I we report the time used by the different parts of the program in a typical calculation. The computation was performed with a SUN sparc 2.

The diffusion step includes the computation and updating of the Voronoi diagram. Most of the time is spent in the computation of the velocity field. The time spent in the diffusion step is a negligible fraction of the total computation time.

TABLE I

Partial and Total Computation Times (Seconds)

Number of vortices	177	349	709	1425
Velocity field	0.39	0.96	2.29	6.26
Diffusion step	0.077	0.165	0.387	0.95
Convection step	1.62	3.94	9.49	25.9
Time/step	1.62	4.16	10.41	29.8
Total diffusion time	58	182	474	1190
Total convection time	1149	2973	7156	19560
Total time	1120	3136	7877	22594

6. CONCLUSIONS

We propose a new deterministic vortex method for the solution of the incompressible Navier–Stokes equations. The method is based on the discretization of the Laplacian on a Voronoi grid, whose nodes are at the vortex locations. The position of the vortices is updated according to the velocity, which is reconstructed from the vorticity via a vortex-blob method. The circulation associated to the vortices satisfies a diffusion equation discretized on the Voronoi mesh.

The conservation properties of the method are a direct consequence of the conservation properties of the vortex-blob schemes and of the discrete Laplacian on a Voronoi mesh. In particular this method preserves exactly the total circulation and gives the exact evolution law for the second moment of the vorticity distribution. Numerical tests show that the discretization error due to the diffusion algorithm is negligible compared to the error introduced by the discretization of the Biot–Savart law. The time spent in the diffusion step is much smaller than the time required for the computation of the velocity.

The stability restriction (4) is not very severe for slightly viscous flow. Implicit schemes for the solution of the diffusion equation on a Voronoi mesh are presently under investigation.

In a practical implementation it would be desirable to make the code adaptive, by implementing creation and annihilation of vortices. We are presently considering this possibility.

We plan to compare this scheme with other methods, such as a spectral method, for the computation of slightly viscous flow in a periodic domain.

ACKNOWLEDGMENTS

I am very grateful to Professor Charles Peskin who gave me the codes for the generation of the Voronoi diagrams. The work was supported by the U.S. Department of Energy Grant DE-FG02-86ER-53223 and by the National Science Foundation Grant DMS-8806727.

REFERENCES

1. A. Aggarwal, B. Chazelle, L. Guibas, C. Ó'Dúnlaing, and C. Yap, *Algorithmica* **3**, 293 (1988).
2. C. Anderson, *J. Comput. Phys.* **62**, 111 (1986).
3. C. Anderson and C. Greengard, *SIAM J. Math. Anal.* **22**, 413 (1985).
4. F. Aurenhammer, Report No. 263, November 1988, Gras Technical University, Austria (unpublished).
5. J. T. Beale and A. Majda, *Math. Comput.* **39**, 1 (1982).
6. C. Börgers, Ph.D. thesis, New York University, 1985 (unpublished).
7. C. Börgers and C. S. Peskin, *J. Comput. Phys.* **70**, 397 (1987).

8. R. Caflisch, "Lectures on Mathematical Aspects of Vortex Dynamics," Lecture Notes, Courant Institute of Mathematical Sciences, New York University, Spring 1987 (unpublished).
9. C. C. Chang, *J. Comput. Phys.* **76**, 281 (1988).
10. A. J. Chorin, *J. Fluid Mech.* **57**, 785 (1973).
11. A. J. Chorin and J. Marsden, *A Mathematical Introduction to Fluid Mechanics* (Springer Verlag, New York, 1979).
12. G. H. Cottet, Rap. Int. n. 108, 1984, Centre de Mathématiques Apliquées, Ecole Polytechnique, Palaiseau (unpublished).
13. G. H. Cottet, "Boundary Conditions and Deterministic Vortex Method for the Navier–Stokes Equations," *Proceeding of the Workshop on Mathematical Aspects of Vortex Dynamics* (Leesburg, 1988), edited by Russel E. Caflisch, (SIAM Publication, Philadelphia, 1989).
14. G. H. Cottet, *Math. Comp.* **56**, 45 (1991).
15. G. H. Cottet, S. Mas-Gallic, and P. A. Raviart, "Vortex Method for the Incompressible Euler and Navier–Stokes Equations," in *Proceedings Workshop on Computational Fluid Dynamics and Reacting Gas Flow* (I.M.A., Southend-on-Sea, September 1986).
16. G. H. Cottet, J. Goodman, and T. Y. Hou, *SIAM J. Num. Anal.* **28**, 291 (1991).
17. P. Degond and S. Mas-Gallic, *Numer. Math.* **57**, 805 (1990).
18. D. Fishelov, *J. Comput. Phys.* **86**, 211 (1990).
19. *The Free Lagrange Method*, Lecture Notes in Physics, Vol. 238, edited by M. J. Fritts, W. P. Crowley, and H. Trease (Springer-Verlag, Berlin, 1985).
20. J. Goodman, *Commun. Pure Appl. Math.* **40**, 189 (1987).
21. J. Goodman, T. H. Hou., and J. Lowengrub, *Commun. Pure Appl. Math.* **43**, 415 (1990).
22. C. Greengard, *J. Comput. Phys.* **61**, 345 (1985).
23. L. Greengard and V. Rohklin, *J. Comput. Phys.* **73**, 325 (1983).
24. A. Jameson and T. J. Baker, "Euler Calculations for a Complete Aircraft," in *Proceedings, International Conference on Numerical Methods in Fluid Dynamics*, Lecture Notes in Physics, Vol. 264 (Springer-Verlag, New York/Berlin, 1986), p. 334.
25. D. G. Long, Ph.D. thesis, University of California and Princeton University, November 1986 (unpublished).
26. C. Marchioro and M. Pulvirenti, *Commun. Math. Phys.* **84**, 483 (1982).
27. C. Marchioro and M. Pulvirenti, *Vortex Methods in Two Dimensional Fluid Dynamics*, Lecture Notes in Physics, Vol. 206 (Springer-Verlag, New York, 1984).
28. M. Perlman, *J. Comput. Phys.* **59**, 200 (1985).
29. G. Russo, *Commun. Pure Appl. Math.* **43**, 697 (1990).
30. G. Russo, "A Lagrangian Vortex Method for the Incompressible Navier–Stokes Equations," in *Proceedings, AMS-SIAM Summer Seminar Vortex Dynamics and Vortex Methods, Seattle, Washington*, 1990, Lectures in Applied Mathematics, Vol. 28, edited by C. R. Anderson and C. Greengard (Am. Math. Soc., Providence, RI, 1991), p. 585.
30a. S. Roberts, *J. Comput. Phys.* **58**, 29 (1985).
31. J. Strain, *SIAM J. Sci. Scat. Comp.*, to appear.
32. J. Strain, *SIAM J. Sci. Statist. Comput.*, submitted.