Journal of Computational Physics **160,** 385–400 (2000)
doi:10.1006/jcph.2000.6474, available online at http://www.idealibrary.com on IDEAL®

A Vortex/Impulse Method for Immersed Boundary Motion in High Reynolds Number Flows

Ricardo Cortez[1]

Department of Mathematics, Tulane University, 6823 St. Charles Avenue, No. 424, New Orleans, Louisiana 70118
E-mail: cortez@math.tulane.edu

Received June 15, 1999

A Lagrangian method that combines vortices and impulse elements (vortex dipoles) is introduced. The applications addressed are flows induced by the motion of thin flexible boundaries immersed in a two-dimensional incompressible fluid. The impulse elements are attached to the boundaries and are used to account for the forces affecting the motion. The vortices occupy a region surrounding the boundaries and are used to account for the viscous effects via a deterministic diffusion method. The convergence of the method is demonstrated numerically. The method is then used to track the motion of an undulating filament, simulating the swimming of an organism in a slightly viscous fluid.

Key Words: particle methods; immersed boundaries.

1. INTRODUCTION

The goal of this paper is to introduce a numerical method for moderate to high Reynolds number flows interacting with immersed boundaries. This is a regime where relatively little work has been done. The numerical solution of fluid flow problems with thin flexible moving boundaries is motivated by the wide range of potential applications in biology and physiology. For example, the walls of the heart or lungs and swimming eels can be modeled as thin membranes embedded in fluids. These are examples of flows with a reasonably high Reynolds number, in contrast to microorganism and cell motion, where the length scales typically reduce the Reynolds number to $O(1)$ or less.

The method introduced here is based on the Lagrangian vortex method but generalizes it with the use of impulse density as a second computational variable. Impulse variables have

[1] This work was supported in part by NSF Grant DMS-9816951.

0021-9991/00 $35.00

shown their usefulness in problems with immersed boundaries, where forces are restricted to thin filaments in the fluid. The use of impulse in the context of immersed boundary motion has been studied for the case of Euler flow in [3, 6, and 21]. In those papers it was shown that the impulse method converged to asymptotic solutions obtained from a perturbation analysis. For viscous flows, a mechanism to model the diffusion term must be added. This can be done while maintaining the Lagrangian character of the method with the use of deterministic diffusion methods. These are schemes in which the vortex strengths, rather than the location of the particles, are changed at every time step to simulate diffusion. Various types of deterministic methods exist (see [4, 7, and 10] for example) but many make use of similar cutoff functions and only differ in the way they approximate the Laplacian operator. We chose to use Fishelov's method since it lends itself more naturally for the purposes of this paper; however, it is not the only available choice. The grid-free character of the method has the desirable property of introducing little numerical dissipation and remaining stable when decreasing values of viscosity are used.

McCracken and Peskin [16] developed a vortex method for the study of blood flow through heart valves. Their method combined point vortices near the immersed boundaries and finite differences on a grid. The results showed signs of noise in the boundary motion, possibly due to statistical error in the random diffusion model they used, compounded by the presence of very strong point vortices. The method presented here was motivated by their work.

The thin flexible membranes are described by points which move with the fluid velocity and are used to compute the forces in the same way as in other methods, such as Peskin's immersed boundary method [9]. The boundary is interpreted as a vortex dipole layer whose strength is space- and time-dependent due to the forces [3]. Typical forces are due to tension, the stretching of the membranes beyond a prescribed resting length; and due to bending, the deviations in the shape of the membrane away from a target curvature. Such forces depend on the specific application and the elasticity model used. For instance, an elastic chamber filling with fluid might only include elastic forces while an active swimming creature might include elastic and bending forces. It is interesting to study the physiological mechanisms that lead to various swimming motions displayed by undulating creatures. This, however, is not the objective of the current work. Instead, a target motion is prescribed in the numerical tests without regard to the biological justification or efficiency of such motion. The main goal is to present the numerical method and its usefulness for the computation of these motions.

Since the numerical method is based on the evolution equations for vorticity and impulse, these are presented first for the case of Euler flow and the numerical method derived from them is explained. The presentation of the method is then finished with a discussion of the diffusion model and its implementation in the present context. Numerical examples are presented at the end.

2. EQUATIONS OF MOTION

Several formulations of the equations of fluid motion are used in computations. The Navier–Stokes equations are commonly written in terms of the fluid velocity or in terms of the vorticity. In both cases, many numerical methods have been based on those equations. More recently the fluid equations have been expressed in terms of impulse density (sometimes called magnetization). Impulse, which can be thought of as linear momentum, can be

defined for constant density fluids as a vector field whose curl is the vorticity in the flow. At the same time the impulse field is not required to have zero divergence. In this way, the impulse and the fluid velocity differ at most by the gradient of some scalar function. The two formulations relevant to this paper are those based on vorticity and impulse.

The evolution equations for a viscous incompressible fluid in $\mathbb{R}^2$ are

$$\mathbf{u}_t + \mathbf{u}\cdot\nabla\mathbf{u} = -\nabla p + \nu\Delta\mathbf{u}, \qquad \nabla\cdot\mathbf{u} = 0, \tag{1}$$

where $\mathbf{u}$ is the velocity, p the pressure, and ν the kinematic viscosity. We assume from now on that the fluid density is constant and set to $\rho = 1$.

The diffusion term will be treated separately in the numerical method so we consider inviscid flow first. In two dimensions and in the absence of forces, the Euler equations in terms of vorticity, ω, are

$$\omega_t + \mathbf{u}\cdot\nabla\omega = 0, \qquad \nabla\cdot\mathbf{u} = 0 \tag{2}$$

so that the vorticity is simply transported by the incompressible flow. In unbounded domains, and assuming the vorticity has compact support, we require that $\mathbf{u}\to 0$ as $|\mathbf{x}|\to\infty$. The incompressible flow field can be given the form $\mathbf{u} = (u_1, u_2) = (\psi_y, -\psi_x)$ for some function ψ, and since vorticity is the curl of velocity, we have $\omega = \partial_x u_2 - \partial_y u_1 = -\Delta\psi$. Thus given the vorticity, the velocity is recovered by the Biot–Savart law,

$$\mathbf{u} = (G_y * \omega, -G_x * \omega),$$

where G is the Green's function satisfying $\Delta G = -\delta$. In full form, the velocity is written

$$\mathbf{u}(\mathbf{x}) = \int_{\mathbb{R}^2} K(\mathbf{x}-\mathbf{x}')\omega(\mathbf{x}')\,d\mathbf{x}' = \frac{1}{2\pi}\int_{\mathbb{R}^2}\frac{(-y+y', x-x')}{(x-x')^2+(y-y')^2}\omega(x', y')\,dx'\,dy'.$$

A more complete discussion of the vorticity formulation of fluid flow can be found in [20, 22].

As mentioned before, the impulse density, denoted by $\mathbf{m}$, is a vector field that coincides with the fluid velocity up to a gradient:

$$\mathbf{m} = \mathbf{u} + \nabla\phi. \tag{3}$$

For incompressible flows, $\mathbf{m}$ is the sum of a divergence-free field and a gradient, so the last equation is the orthogonal decomposition of $\mathbf{m}$. Substituting Eq. (3) into the Euler equations in primitive variables and identifying the pressure with $p = \phi_t + \mathbf{u}\cdot\nabla\phi + \frac{1}{2}|\mathbf{u}|^2$ we arrive at the Euler equations in terms of impulse,

$$\mathbf{m}_t + \mathbf{u}\cdot\nabla\mathbf{m} = -(\nabla\mathbf{u})^T\mathbf{m} + \mathbf{F}, \qquad \mathbf{u} = \mathbb{P}\mathbf{m}, \tag{4}$$

where $\nabla\mathbf{u}$ is a matrix with ij-entry given by $\partial u_i/\partial x_j$. We have included the force term since it is precisely the presence of forces that will give rise to an impulse field in our problems. The last equation represents a projection which expresses the fact that the fluid velocity can be recovered from impulse by extracting the divergence-free part of $\mathbf{m}$. This is done by taking the divergence of Eq. (3), solving the resulting Poisson equation for ϕ in terms of $\mathbf{m}$, and isolating $\mathbf{u}$. Equations (2) and (4) are used to derive the evolution equation of the vortex and impulse strengths in the numerical method.

2.1. *The Forces*

Immersed boundaries can be described in a general way as curves embedded in a two-dimensional flow along which forces act to influence the fluid motion. If the immersed boundary is parametrized by $X(\ell, t)$ for $0 \le \ell \le L$, then the forces concentrated on the membranes are given by

$$\mathbf{F}(\mathbf{x}) = \int_0^L \mathbf{f}(\ell, t)\delta(\mathbf{x} - X(\ell, t))\, d\ell,$$

where $\mathbf{f}(\ell, t)$ represents the force density on the curve. This is a singular expression since δ is a two-dimensional Dirac delta. This formulation is equivalent to specifying jump conditions across the immersed boundary in terms of the force density [19].

3. THE NUMERICAL METHOD

The approach used here is a vorticity-based method that combines impulse vector blobs and vortex blobs. The advantage of using impulse elements is the simplicity with which they introduce the effects of forces into the fluid motion. An organism, such as a fish or an eel, generates forces along its body in order to push the fluid near its tail, which results in forward motion. In terms of the mathematical formulation of the problem, these forces appear as a term on the right hand side of the impulse strength equation.

Suppose the immersed boundary is discretized by the points $\mathbf{x}^k$. The impulse in the flow can be represented as a sum of regularized delta functions centered at the $\mathbf{x}^k$'s,

$$\mathbf{m}(\mathbf{x}) = \Delta\ell \sum_{k=1}^{N_i} \mathbf{m}^k f_\delta(\mathbf{x} - \mathbf{x}^k), \tag{5}$$

where $\Delta\ell$ is a discretization parameter of the immersed boundaries and N_i is the number of impulse elements. The representation of impulse as a sum of blobs is reasonable given the expression of the forces. The cutoff function f_δ is a smooth approximation to the delta function given by $f_\delta(\mathbf{x}) = \delta^{-2} f(\mathbf{x}/\delta)$. Generally the order of the approximation is measured by the number of moment conditions f satisfies. A cutoff that satisfies $\iint f(\mathbf{x})\, d\mathbf{x} = 1$ and

$$\iint x_i^k f(x_1, x_2)\, dx_1\, dx_2 = 0 \qquad \text{for} \quad i = 1, 2 \quad \text{and} \quad k = 1, \ldots, m-1$$

is an mth order cutoff [1, 12].

One needs an evolution equation for the impulse strengths $\mathbf{m}^k(t)$. Based on Eq. (4), the equation of motion in the case of an inviscid fluid is

$$\frac{d\mathbf{m}^k}{dt} = -(\nabla\mathbf{u})^T \mathbf{m}^k + \mathbf{f}^k,$$

where $\mathbf{f}^k$ is the force density at $\mathbf{x}^k$. This equation clearly indicates that since the impulse strengths are initially set to zero, it is the forces that introduce impulse into the flow.

The term $(\nabla\mathbf{u})^T\mathbf{m}^k$ is a stretching term which results in the growth of the impulse vector $\mathbf{m}^k$ whenever a material curve passing through $\mathbf{x}^k$ and perpendicular to $\mathbf{m}^k$ undergoes stretching. This is possible in incompressible flows as long as the fluid compresses in the

direction of $\mathbf{m}^k$ as is the case, for example, at stagnation points. In terms of a numerical method, this growth in the magnitude of $\mathbf{m}$ can lead to a loss of accuracy if the stretching of material curves is too extensive (see [5]). For this reason it is not desirable to use impulse elements in the background space, i.e., detached from the immersed boundary, where the vectors may grow due to stretching of material curves. The impulse elements on a filament representing a swimming creature do not typically experience stretching, since the creature itself does not stretch. This is the motivation for using impulse along the immersed boundaries and vortices in free space.

For the purposes of modeling the diffusion (or possibly other reasons) a vorticity distribution covering a relevant domain is required. In a Lagrangian vortex method the vorticity is approximated by

$$\omega(\mathbf{x}) = h^2 \sum_{i=1}^{N_v} \omega^i \, f_\delta(\mathbf{x} - \mathbf{x}^i), \tag{6}$$

where h^2 is the element of area covered by each vortex and N_v is the total number of vortices. The size of the domain, and hence the number of vortices, depends on the total time of the simulation and the diffusive properties of the fluid. In practice, the domain size is determined by the distribution of significant vortices throughout the simulation. Since vorticity is constant along particle trajectories the vortex strengths do not change due to the advection. It is important to realize that the impulse can be viewed as additional vorticity in the form of a vortex dipole field. This interpretation will be explained in the next section since it is used in the diffusion process.

3.1. *The Velocity Field*

In the numerical method we must find an expression for the fluid velocity that can be evaluated at the particle locations in order to update their positions. The velocity gradients are also needed in the evolution equation of the impulse strengths. The velocity of the fluid is the sum of the contributions from impulse and vorticity. The vorticity in Eq. (6) induces a velocity given by

$$h^2 \sum_{i=1}^{N_v} \omega^i \, K_\delta(\mathbf{x} - \mathbf{x}^i), \tag{7}$$

where the regularized vortex kernel, $K_\delta = (\partial_y, -\partial_x)(G * f_\delta)$, is a smooth approximation to the singular kernel

$$K(\mathbf{y}) = \frac{(-y_2, y_1)}{2\pi |\mathbf{y}|^2}.$$

The velocity contribution from the impulse is obtained by finding the divergence-free part of the field in Eq. (5). This can be done exactly for radially symmetric cutoff functions f_δ. The result (see [2]) may be written as

$$\Delta\ell \sum_{k=1}^{N_i} [\mathbf{m}^k \, f_\delta(r) + \nabla(\mathbf{m}^k \cdot \nabla G_\delta(r))], \tag{8}$$

where $\Delta G_\delta(r) = -f_\delta(r)$ and $r = |\mathbf{x} - \mathbf{x}^k|$. The function $G_\delta = (G * f_\delta)$ is a regularized Green's function which is smooth everywhere. It is found analytically for the chosen cutoff function so that all smoothing functions in the velocity expression are known in advance of the computation. The fluid velocity then is the sum of the two contributions,

$$\mathbf{u}(\mathbf{x}) = \Delta\ell \sum_{k=1}^{N_i} [\mathbf{m}^k f_\delta(r) + \nabla(\mathbf{m}^k \cdot \nabla G_\delta)] + h^2 \sum_{i=1}^{N_v} \omega^i K_\delta(\mathbf{x} - \mathbf{x}^i). \tag{9}$$

The first stage of the numerical method can now be written as a system of ordinary differential equations for the vortex and impulse strengths and for the particle positions:

$$\frac{d\omega^i}{dt} = 0, \qquad i = 1, \ldots, N_v \tag{10}$$

$$\frac{d\mathbf{m}^k}{dt} = -(\nabla \mathbf{u})^T \mathbf{m}^k + \mathbf{f}^k, \qquad k = 1, \ldots, N_i \tag{11}$$

$$\frac{d\mathbf{x}^j}{dt} = \mathbf{u}(\mathbf{x}^j), \qquad \text{for all particles } j. \tag{12}$$

This system describes the evolution of an inviscid flow. Equation (11) requires the computation of the derivatives of the fluid velocity. These are found by direct differentiation of the expression in (9).

The method described so far is for the solution of the Euler equations using a combination of vortices and impulse variables. The solution of the Navier–Stokes equations requires in addition a model for the diffusion of vorticity. This is explained next.

4. THE DIFFUSION

The vortices, placed in a region surrounding the immersed boundaries, initially have zero strength. Their strengths change only as the vorticity on the membranes diffuses to the neighboring region. This is accomplished via a deterministic diffusion method due to Fishelov [10], which was shown to be convergent in [4]. The method takes advantage of the blob representation of a general function $B(\mathbf{x}, t)$ in terms of strengths $b^i(t)$ centered at $\mathbf{x}^i$

$$B(\mathbf{x}, t) = h^2 \sum_{i=1}^{N} b^i(t) f_\delta(\mathbf{x} - \mathbf{x}^i).$$

In order to approximate the Laplacian of $B(\mathbf{x}, t)$ one can differentiate the expression above to obtain

$$\Delta B(\mathbf{x}, t) = h^2 \sum_{i=1}^{N} b^i(t) \Delta f_\delta(\mathbf{x} - \mathbf{x}^i).$$

The accuracy of this approximation depends on the properties of the cutoff function f_δ but one can design such functions of arbitrarily high order. The error bound also depends on the bound of the flow map, which identifies the initial position $\mathbf{x}^i(0)$ to its position at time t. When particles that are initially placed on a regular grid become highly disorganized the error bounds are large. If one has a priori knowledge that the flow map and its derivatives

are bounded, a uniform error bound can be obtained. If such information is not available it may be necessary to regrid the particles occasionally to keep errors from growing to intolerable levels. There are various algorithms that perform this regridding procedure while maintaining accuracy. We explain in the next section the algorithm used here.

In our numerical method the vorticity has two contributions: one from the vortices and another from the impulse. By the definition of impulse, Eq. (3), one can see that the vorticity induced by it is $\nabla \times \mathbf{m}$. Each impulse element contributes to the vorticity an amount $\nabla f_\delta(\mathbf{x} - \mathbf{x}^k) \times \mathbf{m}^k$, which is equivalent to a vortex dipole with a prescribed dipole moment (see [5]). The total vorticity in the flow is given by

$$\omega_{tot}(\mathbf{x}) = \omega(\mathbf{x}) + \nabla \times \mathbf{m}(\mathbf{x}) = h^2 \sum_{i=1}^{N_v} \omega^i f_\delta(\mathbf{x} - \mathbf{x}^i) + \Delta\ell \sum_{k=1}^{N_i} \nabla f_\delta(\mathbf{x} - \mathbf{x}^k) \times \mathbf{m}^k.$$

Extending the diffusion method to this function we write

$$\Delta\omega_{tot}(\mathbf{x}) = h^2 \sum_{i=1}^{N_v} \omega^i \Delta f_\delta(\mathbf{x} - \mathbf{x}^i) + \Delta\ell \sum_{k=1}^{N_i} \nabla\Delta f_\delta(\mathbf{x} - \mathbf{x}^k) \times \mathbf{m}^k$$

and adjust the vortex strengths within the diffusion step by

$$\frac{d\omega^i}{dt} = \nu\Delta\omega_{tot}(\mathbf{x}^i), \qquad i = 1, \ldots, N_v. \tag{13}$$

The impulse strengths are left unchanged.

The numerical method is summarized by performing the following at each time step:

1. Compute the forces $\mathbf{f}^k$ on the immersed boundaries.
2. Update the particle positions and impulse strengths by solving Eqs. (10)–(12).
3. Account for diffusion by solving Eq. (13).

One can see that although the presentation of the numerical method was made with a common cutoff function f_δ and common cutoff parameter δ, this is not necessary. Different cutoffs can be selected for the vortices and the impulse and yet another one can be used in the diffusion step. This may actually be preferable so that each of the procedures is used with the parameters that lead to optimal overall properties.

5. REGRIDDING PROCEDURE

If the particles (vortices) that cover a region in space become very disorganized with respect to their initial positions, it is an indication that gradients of the flow map are large in parts of the domain. This in turn signals the possibility that errors in the particle method, especially in the diffusion model, may be larger than one is willing to accept. In such cases there is a need to lay down a new uniform mesh where the vorticity can be interpolated and the new grid points become the new vortices. In some of the numerical simulations presented here regridding was performed using a procedure of Monaghan [17], which we briefly describe.

Define for $x \geq 0$ the function

$$W_4(x) = \begin{cases} 1 - \frac{1}{2}x^2(5-3x) & \text{for } 0 \leq x < 1 \\ \frac{1}{2}(1-x)(2-x)^2 & \text{for } 1 \leq x < 2 \\ 0 & \text{for } 2 \leq x \end{cases}$$

and notice that $W_4(x)$ and $W_4'(x)$ are continuous. If ω_k are the values of a smooth function at the points $x_k \in \mathbb{R}$ and if z_k are points on a uniform grid of size h, then the interpolation

$$\sum_{k=1}^{N_v} \omega_k W_4(|x_k - z|/h)$$

approximates $\omega(z)$ with errors of $O(h^3)$. The function $W_4(x)$ has the properties that it exactly reproduces polynomials of order up to 2 and if the points x_k already lie on a regular grid of size h, the interpolated values of ω_k are left unchanged. See [17, 18] for more details and other choices.

6. NUMERICAL EXAMPLES

Two numerical examples are presented in this section. The first one is a test case designed to provide some insight into the properties of the method. It consists of tracking the motion of a closed elastic membrane from a given initial shape to its circular equilibrium position. The second example shows the application of the numerical method to the motion of a swimming organism.

6.1. *A Closed Membrane*

Consider an incompressible fluid of constant density occupying all of $\mathbb{R}^2$ and a closed elastic membrane separating the fluid into two disjoint regions. The membrane is assumed to be stretched beyond its rest position so that its equilibrium configuration is the unit circle under tension. The initial position of the membrane is given in polar coordinates by

$$r(\theta) = \sqrt{1 - \epsilon^2/2} + \epsilon \cos(2\theta)$$

with $\epsilon = 0.1$. This represents a perturbation of the circle whose enclosed area is π. The force density at a point $X(\ell, t)$ on the boundary is taken to be proportional to the second derivative of the position vector with respect to the initial parametrization:

$$\mathbf{f}(\ell, t) = \sigma \frac{\partial^2 X(\ell, t)}{\partial \ell^2}.$$

Initially, this force density is approximately proportional to the membrane curvature if ℓ is the arclength parameter. For $t > 0$, however, this may no longer be true. The membrane motion is oscillatory with damping due to the viscosity. The fluid density was set to $\rho = 1$ and the viscosity to $\nu = 0.02$. The stiffness constant in the force density was fixed at $\sigma = 2\pi$.

The membrane was discretized with 256 particles placed initially at equal Euclidean distances. The force density at the points $\mathbf{x}^k$ was approximated by

$$\mathbf{f}^k = \sigma \frac{\mathbf{x}^{k+1} - 2\mathbf{x}^k + \mathbf{x}^{k-1}}{\Delta \ell^2}.$$

The annulus $(1/2)^2 < x^2 + y^2 < (3/2)^2$ was discretized with vortices placed initially on a regular Cartesian grid of size $h = 0.05$. The initial vortex and impulse strengths were set to zero so that the forces developed along the membrane initiate the motion by introducing impulse. The diffusion of the vorticity carried by the impulse elements add to the vortex strengths. In the results presented below, the vortex and impulse elements were regularized with the cutoff $f_\delta(r) = \delta^{-2} f(r/\delta)$, where $f(r) = \frac{1}{2\pi}(r^4 - 6r^2 + 6)e^{-r^2}$ and $\delta = 0.15$. The regularization parameter for the diffusion step was set to 0.19. The system of ordinary differential equations was solved using a fifth-order Runge–Kutta method with time step $\Delta t = 0.0127$.

The position of the particles and contours of the vorticity in the flow are shown in Fig. 1. Although no symmetry was imposed, the motion remains symmetric and hence only one quadrant of the solution is shown. As the membrane oscillates from a left/right elongation to a top/bottom elongation and back, the vorticity turns from positive to negative. The figure shows four instances where the membrane is nearly circular on its way to an elliptical shape. The distortion of the original vortex locations is also apparent though not severe, so no regridding procedure was used in this example.

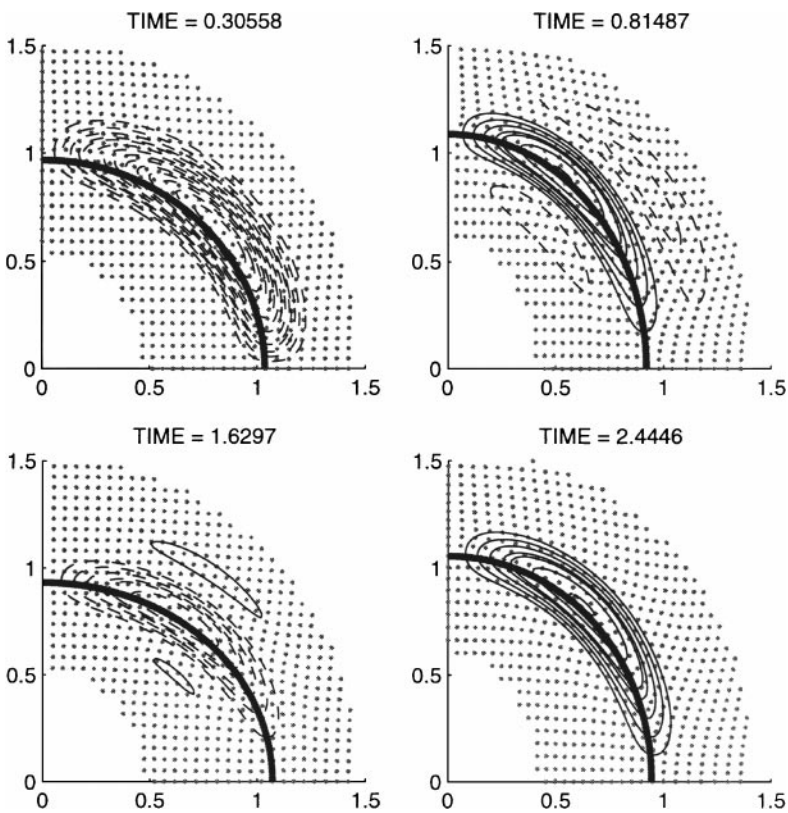

FIG. 1. Vorticity contours on one quadrant of the solution with $\nu = 0.02$. Shown are the membrane, the vortices, contours of positive vorticity (solid), and contours of negative vorticity (dashed) at various times.

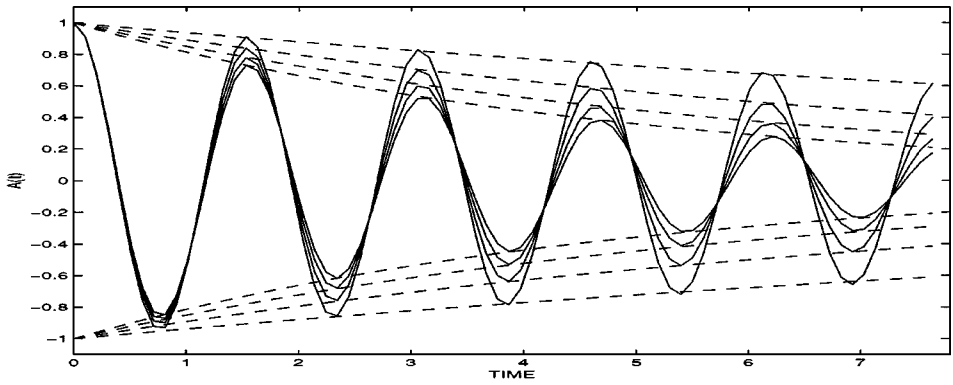

FIG. 2. Decay of the amplitude of the perturbed mode for $\nu = 0.005$, 0.01, 0.015, and 0.02. The envelope in all cases is given by $\exp(-c\nu^p t)$ with $c = 5.39$ and $p = 0.835$.

One can think of the solution in polar coordinates as

$$r(\theta, t) = 1 + \epsilon A(t)\cos(2\theta) + O(\epsilon^2) \tag{14}$$

and plot the amplitude $A(t)$ of the perturbed mode. In [6] it was shown that in the absence of viscosity, it is the magnitude of the force that sets the time scale of the problem. A constant factor multiplying the force density can be eliminated by rescaling time. However, the presence of viscosity introduces a second time scale which describes the decay of the amplitude of the perturbation. Figure 2 shows the evolution of the amplitude $A(t)$ during the course of several oscillations of the membrane. The figure shows the solution for several values of the viscosity ν ranging from 0.005 to 0.02. One can see from the plots that variations in ν produce only slight changes in the frequency of the oscillations but affect visibly the envelope of their amplitude. As a way of measuring empirically the decay of the amplitude, a curve of the form $\exp(-c\nu^p t)$ was fitted to the data. The values $c = 5.39$ and $p = 0.835$ seem to fit all cases well.

A convergence experiment was conducted by computing the solution of the problem using three discretizations of the membrane, each one smaller than the previous one by a factor of $4/5$. The vortices were placed initially on a grid whose size was also reduced by a factor of $4/5$. The parameters mentioned before were used for the coarsest run. The discretizations used were 256, 320, and 400 particles on the membrane and 2494 ($h = 0.05$), 3932 ($h = 0.04$), and 6140 ($h = 0.032$) vortices, respectively. The regularization parameters of the vorticity and the diffusion were reduced at the slower rate $\sqrt{4/5}$. This is consistent with convergence theory for vortex methods [1, 11, 12] and diffusion methods [5, 7]. The impulse variables, however, are used to discretize a line integral rather than a two-dimensional integral. We found that best results were obtained by reducing the impulse regularization parameter proportionally to $\Delta\ell$. An estimate of the convergence rate can be found from the results by assuming the L^2 norm of the error is of the form Ch^r. Then one can compute the error ratio

$$\varepsilon_r = \frac{Ch^r - C\left(\frac{4}{5}h\right)^r}{C\left(\frac{4}{5}h\right)^r - C\left(\frac{16}{25}h\right)^r} = \left(\frac{5}{4}\right)^r$$

and the rate $r = \log(\varepsilon_r)/\log(5/4)$. Table I shows the error ratio and estimated rate of

TABLE I
Error Ratios between Successive Solutions and Corresponding Convergence Rates

Time	ε_r	$\frac{\log(\varepsilon_r)}{\log(1.25)}$
$t = 0.51$	1.4331	1.61
$t = 1.12$	1.3916	1.48
$t = 1.94$	1.6119	2.14
$t = 2.65$	1.3836	1.46

convergence for the membrane location at selected times. The times were chosen when the membranes were nearly circular during the first two periods of oscillation. When the membranes reach their maximum eccentricity, all three solutions tend to coincide, making it difficult to compute ε_r accurately. The table shows convergence rates between 1 and 2. This may be an indication that the various terms that contribute to the error are not sufficiently balanced to give a uniform rate or that not all error terms produce the same rate. The results clearly show that the method converges with at least first order rate. Figure 3 shows the numerator and denominator of the expression for ε_r at time $t = 1.94$ as an illustration of the relative error magnitudes.

6.2. *A Swimming Creature*

The second example is that of a fast-swimming creature via undulatory motion. Typically, a wave is observed to move through the body of the creature from head to tail with larger amplitude at the tail (see [14, p. 14]). This results in an overall forward motion. As this paper is not concerned with the physiological mechanism that leads from muscle contraction to the development of such a wave, we select a sinusoidal wave to act as the target shape of the creature. However, the shape itself is not imposed, only its curvature. This defines the shape of the creature relative to itself but the exact position in space or orientation with respect to a fixed line are not prescribed and the creature is allowed to experience solid-body rotation and translation. In this example we follow the description of forces used in [8, 9].

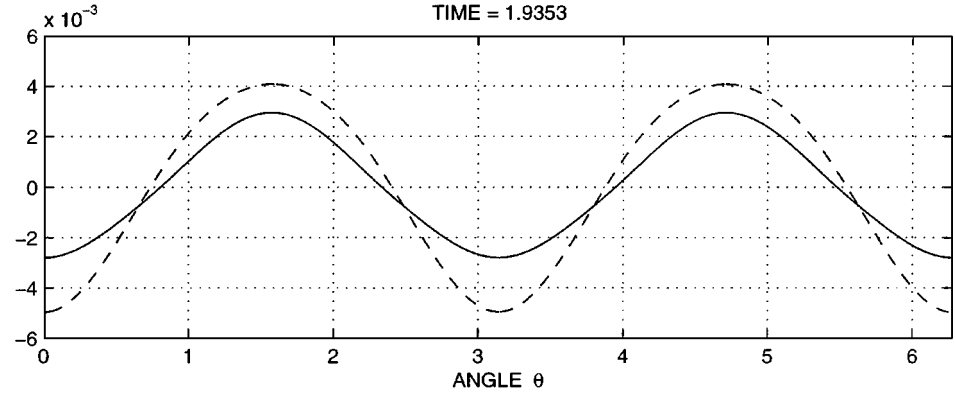

FIG. 3. Radial difference between successive solutions at time $t = 1.94$. The dashed line is the numerator of ε_r and the solid line is the denominator.

Let s be the arclength parameter, $0 \le s \le L$, with $s = 0$ corresponding to the tail of the creature and $s = L$ to the head. Let $y(s, t)$ be given by

$$y(s, t) = \epsilon \left(\frac{L - s}{L} \right) \sin(\kappa s - \Omega t)$$

and $x(s, t)$ by $(dx/ds)^2 + (dy/ds)^2 = 1$. Now let $\hat{n}$ be a unit vector normal to this curve and let $c(s, t)$ be its curvature. This represents the target curvature of the swimming creature. The numerical parameters common to all runs are $L = 0.2$, $\epsilon = 0.02$, $\kappa = 2\pi/L$, and $\Omega = 8\pi$. The Reynolds number based on the wavelength is calculated as

$$R = \frac{\Omega}{\nu \kappa^2}.$$

During the computation of the motion, forces are introduced to keep the creature's shape as close as possible to the target shape. It is convenient to define a type of energy as a function of the immersed boundary discretization. Given such energy function, say E_d, the forces can be defined by

$$\mathbf{f}^j = -\nabla_{\mathbf{x}^j} E_d.$$

The discrete energy function is formed so that as the discretization is reduced, E_d has a well-defined limit. This is done by defining the continuous energy function

$$E(\mathbf{x}, t) = C_1 \int_0^L [\|d\mathbf{x}/ds\| - 1]^2 \, ds + C_2 \int_0^L [(d^2\mathbf{x}/ds^2 \cdot \hat{n}) - c(s, t)]^2 \, ds$$

which increases from zero as the actual shape of the creature departs from its target. A discrete version of this energy is

$$\begin{aligned} E_d = {} & C_1 \Delta s \sum_{k=1}^{N_i - 1} [\|(\mathbf{x}^{k+1} - \mathbf{x}^k)/\Delta s\| - 1]^2 \\ & + C_2 \Delta s \sum_{k=2}^{N_i - 1} [((\mathbf{x}^{k+1} - \mathbf{x}^k) \times (\mathbf{x}^k - \mathbf{x}^{k-1}))/\Delta s^3 - c(s, t)]^2. \end{aligned}$$

The term containing the cross product is chosen as an approximation to the curvature of the computed shape. The constants C_1 and C_2 are chosen to be large to ensure a shape reasonably close to the target. The values used here are $C_1 = 10$ and $C_2 = 10^5$.

The organism was discretized with 128 particles so that $\Delta s = 1.575 \times 10^{-3}$. The vortices were placed on a uniform rectangular grid of size $h = 4.6875 \times 10^{-3}$ covering an area of 0.0545 squared units, requiring 2480 vortices. The cutoff parameter was chosen to be $\delta = 0.01342$ for both the vorticity and the impulse. The cutoff used was the radial function $f_\delta(r) = \delta^{-2} f(r/\delta)$, where $f(r) = \frac{1}{2\pi}(4e^{-r^2} - e^{-r^2/2})$.

The forces in this computation may be extremely large due to the high stiffness required in the energy function. This imposes a time-step limitation that can be severe (with a grid-based method this limitation is much more severe than a CFL condition), which makes it practical to treat the force calculation at least semi-implicitly [8, 13, 15]. The results below, however, were obtained using a 5th-order Runge–Kutta method with a time step of

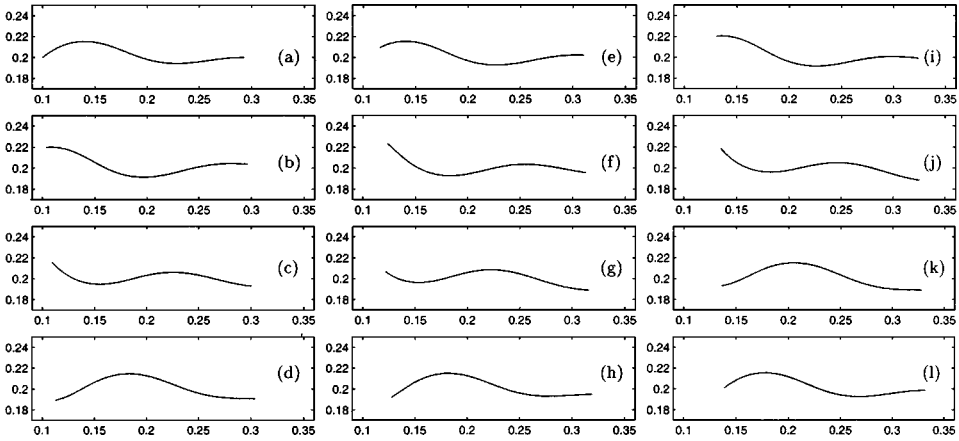

FIG. 4. Swimming motion over three periods. Frames shown at $t = 0$, 0.05, 0.125, 0.2, 0.275, 0.35, 0.4, 0.475, 0.55, 0.625, 0.7, and 0.75. The viscosity was $\nu = 2 \times 10^{-4}$.

$\Delta t = 6.25 \times 10^{-5}$. The viscosity was set to $\nu = 2 \times 10^{-4}$, which gives a Reynolds number of about $R = 127$. For comparison we mention that the time step used in [9] for a similar problem, but with an implicit method, was smaller than the one used here, even when the Reynolds number in [9] was about 50 times smaller.

Frames (a)–(l) of Fig. 4 show the creature from $t = 0$ to $t = 0.75$, which corresponds to three periods of the target wave. The creature swims to the right at a rate of about 6.5% of its body length per wave period.

The forces are designed to keep contiguous particles describing the organism at a separation of approximately Δs throughout the motion. Figure 5 shows the computed length of the creature during the simulation. The original length of 0.2 is maintained within a relative error of less than 0.1%.

Due to the nature of the forces, the creature itself is a very strong vortex sheet. Figure 6 shows contours of vorticity at four instants during the motion. Solid contours are of positive vorticity and dashed contours are of negative vorticity. A distinctive feature is that the regions of vorticity along the organism alternate in sign according to its curvature and these regions slide down the body of the creature until they reach the tail and are shed behind it. This shedding of vorticity is not observed at low Reynolds numbers. Flow circulation regions matching these vortex patterns have been observed experimentally [14, p. 57]. Figure 7 shows the velocity field at two times when the wave through the body of the creature is in opposite phases.

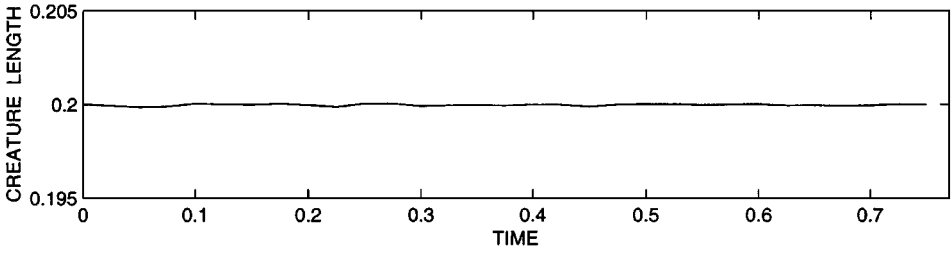

FIG. 5. Length of the swimming creature during three periods of the computation.

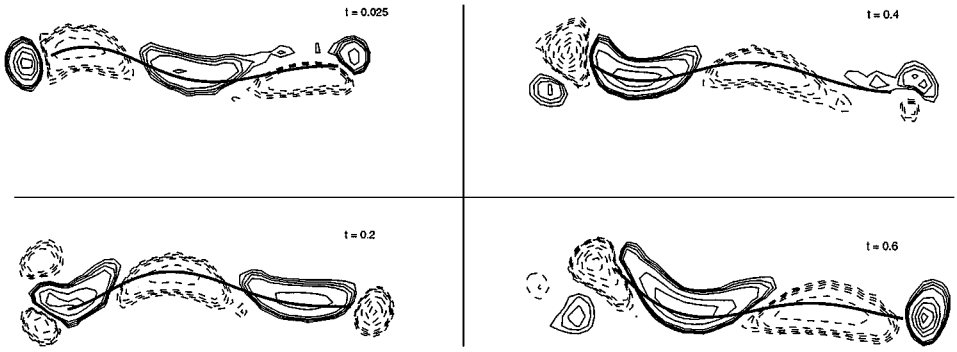

FIG. 6. Vorticity contours during the motion at four different times. Solid curves are of positive vorticity.

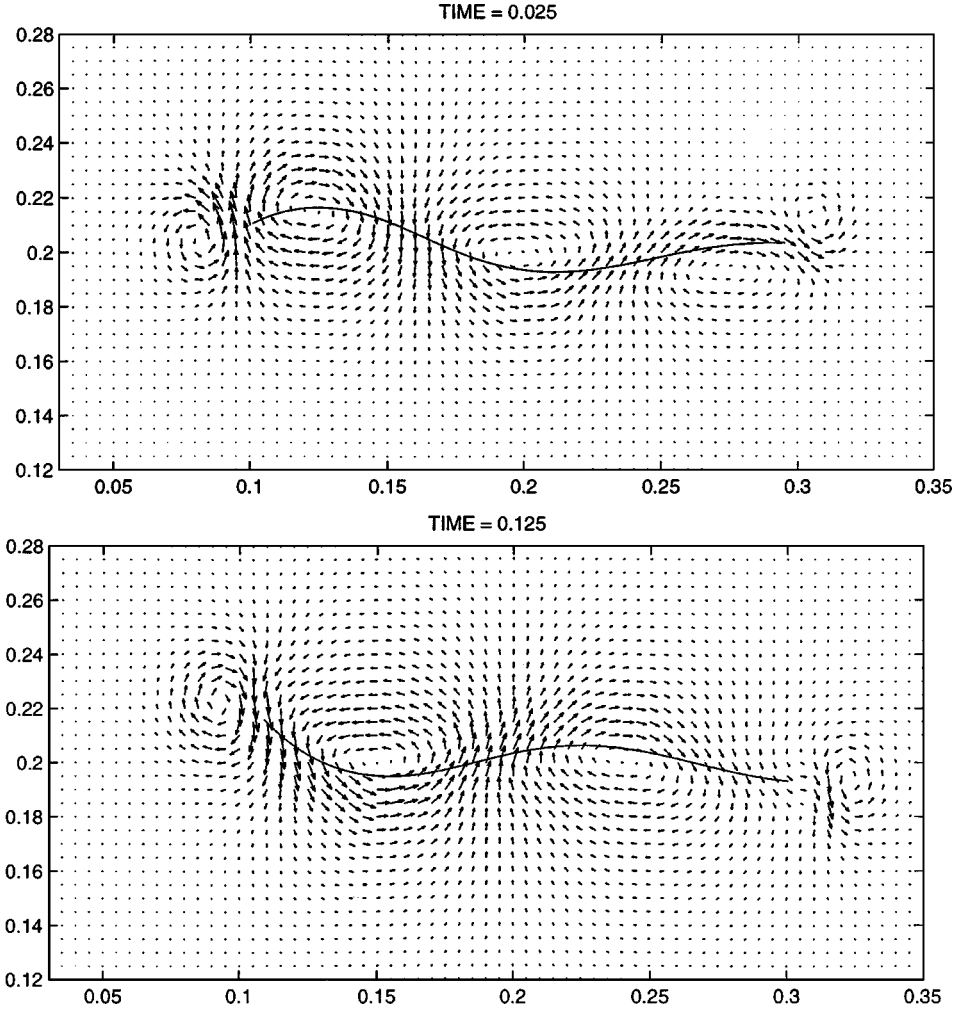

FIG. 7. Velocity field at $t = 0.025$ and $t = 0.125$.

7. DISCUSSION AND CONCLUSIONS

A method designed to model the interaction of high Reynolds number flows with flexible membranes has been presented. The method uses a combination of vortices and impulse elements. Numerical experiments indicate that the method converges at least at a linear rate. The Lagrangian nature of the method allows the modeling of slightly viscous fluids and avoids instabilities associated with grid methods. These are especially severe when very stiff forces are combined with low viscosity [23]. There has been relatively little work done on the development of numerical methods specifically designed for a high Reynolds number regime, although many applications fall into this category. In the vortex/impulse method the time step is restricted by the stiffness of the system of ordinary differential equations, but is unaffected by increasingly large Reynolds numbers. The stiffness of the problem has been addressed in various contexts. Proposed approaches designed to increase the time step include implicit or semi-implicit treatment of the forces [8, 13, 15] and other ways of removing the stiffness [24]. These have not been used here but are being explored for future implementations of the method.

In high Reynolds number flows, the vorticity that affects the fluid motion significantly is restricted to a smaller region near the immersed boundaries. In order to resolve these boundary layers numerically, the vortices must be packed more closely within this region. The fast evaluation of the velocity is necessary once the number of particles becomes large enough. This was not done in the present paper for the modest number of particles used. Since in the method presented here there are interactions between different types of particles, there are several potentials that will have to be evaluated using fast algorithms. They are the velocity induced by the vortices, Eq. (7), the velocity induced by the impulse, Eq. (8), the derivatives of these velocity contributions since they are used to update the impulse strengths, and the potential associated with the diffusion. Of these, only the vortex potential and the diffusion kernel have been previously addressed. Work in this direction is needed.

REFERENCES

1. J. T. Beal and A. Majda, High order accurate vortex methods with explicit velocity kernels, *J. Comput. Phys* **58**, 188 (1985).
2. T. F. Buttke, Velicity methods: Lagrangian numerical methods which preserve the Hamiltonian structure of incompressible fluid flow, *in* "Vortex Flows and Related Numerical Methods" (J. T. Beale, G.-H. Cottet, and S. Huberson, Eds.), NATO ASI Series C, Vol. 395 (Kluwer Academic, Dordrecht/Norwell, MA, 1993).
3. R. Cortez, An impulse-based approximation of fluid motion due to boundary forces, *J. Comput. Phys.* **123**, 341 (1996).
4. R. Cortez, Convergence of high-order deterministic particle methods for the convection-diffusion equation, *Comm. Pure Appl. Math.* **50**, 1235, (1997).
5. R. Cortez, On the accuracy of impulse methods for fluid flow, *SIAM J. Sci. Comput.* **19**, 1290 (1998).
6. R. Cortez and D. A. Varela, The dynamics of an elastic membrane using the impulse method, *J. Comput. Phys.* **138**, 224 (1997).
7. P. Degond and S. Mas-Gallic, The weighted particle method for convection–diffusion equations. 1. The case of an isotropic viscosity, *Math. Comp.* **53**, 485 (1989).
8. L. J. Fauci and A. L. Fogelson, Truncated Newton methods and the modeling of complex immersed elastic structures, *Comm. Pure Appl. Math.* **46**, 787 (1993).
9. L. J. Fauci and C. Peskin, A computational model of aquatic animal locomotion, *J. Comput. Phys.* **77**, 85 (1988).
10. D. Fishelov, A new vortex scheme for viscous flows, *J. Comput. Phys.* **86**, 211 (1990).

11. O. H. Hald, Convergence of vortex methods, II, *SIAM J. Sci. Stat. Comput.* **16**, 726 (1979).
12. O. H. Hald, Convergence of vortex methods, *in* "Vortex Methods and Vortex Motion" (K. E. Gustafson and J. A. Sethian, Eds.), p. 33, (SIAM, Philadelphia, 1991).
13. R. J. LeVeque and Z. Li, Immersed interface methods for Stokes flow with elastic boundaries or surface tension, *SIAM J. Sci. Comput.* **18**(3), 709 (1997).
14. Sir M. J. Lighthill, "Mathematical Biofluiddynamics" (SIAM, Philadelphia, 1975).
15. A. Mayo and C. S. Peskin, An implicit numerical method for fluid dynamics problems with immersed elastic boundaries, *Contemp. Math.* **141**, 261 (1993).
16. M. F. McCracken and C. S. Peskin, A vortex method for blood flow through heart valves, *J. Comput. Phys* **35**, 183 (1980).
17. J. J. Monaghan, Extrapolating B-splines for interpolation, *J. Comput. Phys.* **60**, 253 (1985).
18. H. Nordmark, Deterministic high order vortex methods for the 2D Navier–Stokes equation with rezoning, *J. Comput. Phys.* **129**, 41 (1996).
19. C. S. Peskin and B. F. Printz, Improved volume conservation in the computation of flows with immersed elastic boundaries, *J. Comput. Phys* **105**, 33 (1993).
20. E. G. Puckett, A review of vortex methods, *in* "Incompressible Computational Fluid Mechanics" (R. Nicolaides and M. Ginzburger, Eds.) (Cambridge Univ. Press, Cambridge, UK, 1992).
21. M. C. Recchioni and G. Russo, Hamilton-based numerical methods for a fluid–membrane interaction in two and three dimensions, *SIAM J. Sci. Comput.* **19**, 861 (1998).
22. P. G. Saffman, "Vortex Dynamics" (Cambridge Univ. Press, Cambridge, UK, 1992).
23. J. M. Stockie and B. R. Wetton, Analysis of stiffness in the immersed boundary method and implications for time-stepping schemes, *J. Comput. Phys* (1999), to appear.
24. J. S. Lowengrub, T. Y. Hou, and M. J. Shelley, Removing the stiffness from interfacial flows with surface tension, *J. Comput. Phys.* **114**, 312 (1994).