The immersed boundary method

Charles S. Peskin
Courant Institute of Mathematical Sciences,
New York University, 251 Mercer Street,
New York, NY10012-1185, USA
E-mail: `peskin@cims.nyu.edu`

To Dora and Volodya

This paper is concerned with the mathematical structure of the immersed boundary (IB) method, which is intended for the computer simulation of fluid–structure interaction, especially in biological fluid dynamics. The IB formulation of such problems, derived here from the principle of least action, involves both Eulerian and Lagrangian variables, linked by the Dirac delta function. Spatial discretization of the IB equations is based on a fixed Cartesian mesh for the Eulerian variables, and a moving curvilinear mesh for the Lagrangian variables. The two types of variables are linked by interaction equations that involve a smoothed approximation to the Dirac delta function. Eulerian/Lagrangian identities govern the transfer of data from one mesh to the other. Temporal discretization is by a second-order Runge–Kutta method. Current and future research directions are pointed out, and applications of the IB method are briefly discussed.

CONTENTS

1. Introduction

The immersed boundary (IB) method was introduced to study flow patterns around heart valves and has evolved into a generally useful method for problems of fluid–structure interaction. The IB method is both a mathematical formulation and a numerical scheme. The mathematical formulation employs a mixture of Eulerian and Lagrangian variables. These are related by interaction equations in which the Dirac delta function plays a prominent role. In the numerical scheme motivated by the IB formulation, the Eulerian variables are defined on a fixed Cartesian mesh, and the Lagrangian variables are defined on a curvilinear mesh that moves freely through the fixed Cartesian mesh without being constrained to adapt to it in any way at all. The interaction equations of the numerical scheme involve a smoothed approximation to the Dirac delta function, constructed according to certain principles that we shall discuss.

This paper is concerned primarily with the mathematical structure of the IB method. This includes both the IB form of the the equations of motion and also the IB numerical scheme. Details of implementation are omitted, and applications are discussed only in summary fashion, although references to these topics are provided for the interested reader.

2. Equations of motion

Our purpose in this section is to derive the IB formulation of the equations of motion of an incompressible elastic material. The approach that we take is similar to that of Ebin and Saxton (1986, 1987). The starting point of this derivation is the principle of least action. Although we begin in Lagrangian variables, the key step is to introduce Eulerian variables along the way, and to do so in a manner that brings in the Dirac delta function. Roughly speaking, our goal here is to make the equations of elasticity look as much as possible like the equations of fluid dynamics. Once we have done that, fluid–structure interaction will be easier to handle.

Consider, then, an elastic incompressible material filling three-dimensional space. Let (q, r, s) be curvilinear coordinates attached to the material, so that fixed values of (q, r, s) label a material point. Let $\mathbf{X}(q, r, s, t)$ be the position at time t in Cartesian coordinates of the material point whose label is (q, r, s). Let $M(q, r, s)$ be the mass density of the material in the sense that $\int_Q M(q, r, s) \, dq \, dr \, ds$ is the mass of the part of the material defined by $(q, r, s) \in Q$. Note that M is independent of time, since mass is conserved.

Note that $\mathbf{X}(\,,\,,\,t)$ describes the configuration in space of the whole material at the particular time t. We assume that this determines the elastic (potential) energy of the material according to an energy functional $E[\mathbf{X}]$ such that $E[\mathbf{X}(\,,\,,\,t)]$ is the elastic energy stored in the material at time t.

A prominent role in the following will be played by the Fréchet derivative of E, which is implicitly defined as follows.

Consider a perturbation $\wp\mathbf{X}(\,,\,,\,,t)$ of the configuration $\mathbf{X}(\,,\,,\,,t)$. (We denote perturbations by the symbol $\wp$ instead of the traditional calculus-of-variations symbol δ. That is because we need δ for the Dirac delta function that will soon make its appearance.) Up to terms of first order, the resulting perturbation in elastic energy will be a *linear* functional of the perturbation in configuration of the material. Such a functional can always be put in the form

$$\wp E\left[\mathbf{X}(\,,\,,\,,t)\right] = \int \left(-\mathbf{F}(q,r,s,t)\right) \cdot \wp\mathbf{X}(q,r,s,t)\,\mathrm{d}q\,\mathrm{d}r\,\mathrm{d}s \qquad (2.1)$$

The function $-\mathbf{F}(\,,\,,\,,t)$ which appears in this equation is the Fréchet derivative of E evaluated at the configuration $\mathbf{X}(\,,\,,\,,t)$. The physical interpretation of the foregoing is that $\mathbf{F}$ is the force density (with respect to q,r,s) generated by the elasticity of the material. This is essentially the principle of virtual work. As shorthand for (2.1) we shall write

$$\mathbf{F} = -\frac{\wp E}{\wp\mathbf{X}}. \qquad (2.2)$$

We digress here to give an example of an elastic energy functional E and the elastic force $\mathbf{F}$ that it generates. Consider a system of elastic fibres, with the fibre direction $\boldsymbol{\tau}$ varying smoothly as a function of position. Let q,r,s be material (Lagrangian) curvilinear coordinates chosen in such a manner that $q,r = \text{constant}$ along each fibre. We assume that the elastic energy is of the form

$$E = \int \mathcal{E}\left(\left|\frac{\partial\mathbf{X}}{\partial s}\right|\right)\mathrm{d}q\,\mathrm{d}r\,\mathrm{d}s. \qquad (2.3)$$

The meaning of this is that the elastic energy depends only on the strain in the fibre direction, and not at all on the cross-fibre strain. In other words, we have an extreme case of an anisotropic material. Also, since there is no restriction on $|\partial\mathbf{X}/\partial s|$, which determines the local fibre strain, and since the local energy density $\mathcal{E}$ is an arbitrary function of this quantity, we are dealing here with a case of nonlinear elasticity. To evaluate $\mathbf{F}$, we apply the perturbation operator $\wp$ to E and then use integration by parts with respect to s to put the result in the following form:

$$\wp E = -\int \frac{\partial}{\partial s}\left(\mathcal{E}'\left(\left|\frac{\partial\mathbf{X}}{\partial s}\right|\right)\frac{\partial\mathbf{X}/\partial s}{|\partial\mathbf{X}/\partial s|}\right) \cdot \wp\mathbf{X}\,\mathrm{d}q\,\mathrm{d}r\,\mathrm{d}s, \qquad (2.4)$$

where $\mathcal{E}'$ is the derivative of $\mathcal{E}$. By definition, then,

$$\mathbf{F} = \frac{\partial}{\partial s}\left(\mathcal{E}'\left(\left|\frac{\partial\mathbf{X}}{\partial s}\right|\right)\frac{\partial\mathbf{X}/\partial s}{|\partial\mathbf{X}/\partial s|}\right). \qquad (2.5)$$

Let

$$T = \mathcal{E}'\left(\left|\frac{\partial \mathbf{X}}{\partial s}\right|\right) = \text{fibre tension}, \qquad (2.6)$$

and

$$\boldsymbol{\tau} = \frac{\partial \mathbf{X}/\partial s}{|\partial \mathbf{X}/\partial s|} = \text{unit tangent to the fibres}. \qquad (2.7)$$

Then

$$\mathbf{F} = \frac{\partial}{\partial s}(T\boldsymbol{\tau}). \qquad (2.8)$$

This formula can also be derived without reference to the elastic energy, by starting from the assumption that $\pm T\boldsymbol{\tau}\,\mathrm{d}q\,\mathrm{d}r$ is the force transmitted by the bundle of fibres $\mathrm{d}q\,\mathrm{d}r$, and by considering force balance on an arbitrary segment of such a bundle given by $s \in (s_1, s_2)$.

Expanding the derivative in (2.8), we reach the important conclusion that the elastic force density generated by a system of elastic fibres is locally parallel to the osculating plane of the fibres, that is, the plane spanned by $\boldsymbol{\tau}$ and $\partial\boldsymbol{\tau}/\partial s$. There is no elastic force density in the binormal direction.

Returning to our main task, we consider the constraint of incompressibility. Let

$$J(q, r, s, t) = \det\left(\frac{\partial \mathbf{X}}{\partial q}, \frac{\partial \mathbf{X}}{\partial r}, \frac{\partial \mathbf{X}}{\partial s}\right). \qquad (2.9)$$

The volume occupied at time t by the part of the material defined by $(q, r, s) \in Q$ is given by $\int_Q J(q, r, s, t)\,\mathrm{d}q\,\mathrm{d}r\,\mathrm{d}s$. Since the material is incompressible, this should be independent of time for every choice of Q, which is only possible if

$$\frac{\partial J}{\partial t} = 0. \qquad (2.10)$$

From now on, we shall write $J(q, r, s)$ instead of $J(q, r, s, t)$.

The principle of least action states that our system will evolve over the time interval $(0, T)$ in such a manner as to minimize the action S defined by

$$S = \int_0^T L(t)\,\mathrm{d}t, \qquad (2.11)$$

where L is the Lagrangian (defined below). The minimization is to be done subject to the constraint of incompressibility (equation (2.10), above), and also subject to given initial and final configurations:

$$\mathbf{X}(q, r, s, 0) = \mathbf{X}_0(q, r, s), \qquad (2.12)$$
$$\mathbf{X}(q, r, s, T) = \mathbf{X}_T(q, r, s). \qquad (2.13)$$

In general, the Lagrangian L is the difference between the kinetic and

potential energies. In our case, this reads

$$L(t) = \frac{1}{2} \int M(q,r,s) \left| \frac{\partial \mathbf{X}}{\partial t}(q,r,s,t) \right|^2 \, \mathrm{d}q \, \mathrm{d}r \, \mathrm{d}s - E[\mathbf{X}(\, , \, , \, ,t)]. \qquad (2.14)$$

Thus, for arbitrary $\wp\mathbf{X}$ consistent with the constraints, we require

$$0 = -\wp S = \int_0^T \int \left(M\frac{\partial^2 \mathbf{X}}{\partial t^2} - \mathbf{F} \right) \cdot \wp\mathbf{X} \, \mathrm{d}q \, \mathrm{d}r \, \mathrm{d}s \, \mathrm{d}t. \qquad (2.15)$$

To arrive at this result, we have used integration by parts with respect to t in the first term. In the second term we have made use of the definition of $-\mathbf{F}$ as the Fréchet derivative of the elastic energy E.

At this point, if $\wp\mathbf{X}$ were arbitrary we would be done, and the conclusion would be nothing more than Newton's law that force equals mass times acceleration. But $\wp\mathbf{X}$ is not arbitrary: it must be consistent with the constraints, in particular with the constraint of incompressibility (equations (2.9)–(2.10)). This constraint seems difficult to deal with in its present form.

Perhaps we can overcome this difficulty by introducing Eulerian variables. As is well known, the constraint of incompressibility takes on a very simple form in terms of the Eulerian velocity field $\mathbf{u}(\mathbf{x},t)$, namely $\nabla \cdot \mathbf{u} = 0$. The following definitions of Eulerian variables are mostly standard. The new feature, however, is to introduce a pseudo-velocity $\mathbf{v}$ that corresponds to the perturbation $\wp\mathbf{X}$ in the same way as the physical velocity field $\mathbf{u}$ corresponds to the rate of change of the material's configuration $\partial\mathbf{X}/\partial t$.

More precisely, let $\mathbf{u}$ and $\mathbf{v}$ be implicitly defined as follows:

$$\mathbf{u}\left(\mathbf{X}(q,r,s,t),t\right) = \frac{\partial\mathbf{X}}{\partial t}(q,r,s,t), \qquad (2.16)$$

$$\mathbf{v}\left(\mathbf{X}(q,r,s,t),t\right) = \wp\mathbf{X}(q,r,s,t). \qquad (2.17)$$

Thus $\mathbf{u}(\mathbf{x},t)$ is the velocity of whatever material point happens to be at position $\mathbf{x}$ at time t, and $\mathbf{v}(\mathbf{x},t)$ is the perturbation experienced by whatever material point happens to be at position $\mathbf{x}$ at time t (according to the unperturbed motion). We shall also make use of the familiar material derivative defined as follows:

$$\frac{D\mathbf{u}}{Dt} = \frac{\partial\mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla\mathbf{u}, \qquad (2.18)$$

for which we have the identity

$$\frac{D\mathbf{u}}{Dt}\left(\mathbf{X}(q,r,s,t),t\right) = \frac{\partial^2\mathbf{X}}{\partial t^2}(q,r,s,t). \qquad (2.19)$$

Thus $\frac{D\mathbf{u}}{Dt}(\mathbf{x},t)$ is the acceleration of whatever material point happens to be at position x at time t.

In the following, we shall need the form taken by the constraints in the new variables. There is no difficulty about the initial and final value constraints, (2.12)–(2.13). Since the perturbation must vanish at the initial and final times to be consistent with these constraints, we simply have $\mathbf{v}(\mathbf{x}, 0) = \mathbf{v}(\mathbf{x}, T) = 0$. The incompressibility constraint requires further discussion, however. The perturbations that we consider are required to be volume-preserving (for every piece of the material, not just for the material as a whole). This means that, if $\mathbf{X} + \wp\mathbf{X}$ is substituted for $\mathbf{X}$ in the formula for J, equation (2.9), then the value of $J(q, r, s)$ should be unchanged, at any particular q, r, s, up to terms of first order in $\wp\mathbf{X}$.

To see what this implies about $\mathbf{v}$, it is helpful to use matrix notation. Thus, let

$$\mathbf{a} = (q, r, s), \tag{2.20}$$

and let $\partial\mathbf{X}/\partial\mathbf{a}$ be the 3×3 matrix whose entries are the components of $\mathbf{X}$ differentiated with respect to q, r, or s. In this notation

$$J = \det\left(\frac{\partial\mathbf{X}}{\partial\mathbf{a}}\right). \tag{2.21}$$

Now, we need to apply the perturbation operator $\wp$ to both sides of this equation, but to do so we need the following identity involving the perturbation of the determinant of an arbitrary nonsingular square matrix A:

$$\wp\log\left(\det(A)\right) = \operatorname{trace}\left((\wp A)A^{-1}\right). \tag{2.22}$$

Although this identity is well known, we give a brief derivation of it for completeness. The starting point is the familiar formula for the inverse of a matrix in terms of determinants, written in the following possibly unfamiliar way:

$$\left(A^{-1}\right)_{ji} = \frac{1}{\det(\mathrm{A})}\frac{\partial\det(A)}{\partial A_{ij}}, \tag{2.23}$$

where we have written the signed cofactor of A_{ij} as $\partial\det(A)/\partial A_{ij}$. That we may do so follows from the expansion of the determinant by minors of row i (or column j). Making use of this formula (in the next-to-last step, below), we find

$$\begin{aligned}
\wp\log\left(\det(A)\right) &= \sum_{ij}(\wp A)_{ij}\frac{\partial\log\left(\det(A)\right)}{\partial A_{ij}} \\
&= \sum_{ij}(\wp A)_{ij}\frac{1}{\det(A)}\frac{\partial\det(A)}{\partial A_{ij}} \\
&= \sum_{ij}(\wp A)_{ij}(A^{-1})_{ji} \\
&= \operatorname{trace}\left((\wp A)A^{-1}\right).
\end{aligned} \tag{2.24}$$

This establishes the identity in (2.22).

Now consider the equation that implicitly defines $\mathbf{v}$, (2.17). Differentiating on both sides with respect to q, r, or s and using the chain rule, we get (in matrix notation)

$$\frac{\partial \wp \mathbf{X}}{\partial \mathbf{a}} = \frac{\partial \mathbf{v}}{\partial \mathbf{x}} \frac{\partial \mathbf{X}}{\partial \mathbf{a}}. \tag{2.25}$$

Now interchange the order of the operators $\wp$ and $\partial/\partial \mathbf{a}$, and then multiply both sides of this equation on the right by the inverse of $\partial \mathbf{X}/\partial \mathbf{a}$. The result is

$$\left(\wp \frac{\partial \mathbf{X}}{\partial \mathbf{a}} \right) \left(\frac{\partial \mathbf{X}}{\partial \mathbf{a}} \right)^{-1} = \frac{\partial \mathbf{v}}{\partial \mathbf{x}}. \tag{2.26}$$

Taking the trace of both sides, and making use of the identity derived above (equation (2.22)) as well as the definition of J (equation (2.21)), we see that

$$\wp \log \left(J(q, r, s) \right) = \nabla \cdot \mathbf{v}(\mathbf{X}(q, r, s, t), t). \tag{2.27}$$

Thus, the constraint of incompressibility ($\wp J = 0$) is equivalent to

$$\nabla \cdot \mathbf{v} = 0. \tag{2.28}$$

An argument that is essentially identical to the foregoing yields the familiar expression of the incompressibility constraint in terms of the Eulerian velocity field:

$$\nabla \cdot \mathbf{u} = 0. \tag{2.29}$$

This completes the discussion of the Eulerian form of the constraint of incompressibility.

We are now ready to introduce Eulerian variables into our formula for the variation of the action $\wp S$. We do so by using the defining property of the Dirac delta function, that one can evaluate a function at a point by multiplying it by an appropriately shifted delta function and integrating over all of space. We use this twice, once for the perturbation in configuration of the elastic body, and once for the dot product of that perturbation with the acceleration of the elastic body:

$$\wp \mathbf{X}(q, r, s, t) = \int \mathbf{v}(\mathbf{x}, t)\, \delta \left(\mathbf{x} - \mathbf{X}(q, r, s, t) \right) \mathrm{d}\mathbf{x}, \tag{2.30}$$

$$\frac{\partial^2 \mathbf{X}}{\partial t^2}(q, r, s, t) \cdot \wp \mathbf{X}(q, r, s, t) = \int \frac{D\mathbf{u}}{Dt} \cdot \mathbf{v}(\mathbf{x}, t)\, \delta \left(\mathbf{x} - \mathbf{X}(q, r, s, t) \right) \mathrm{d}\mathbf{x}. \tag{2.31}$$

In these equations (and throughout this paper), $\delta(\mathbf{x})$ denotes the three-dimensional delta function $\delta(x_1)\delta(x_2)\delta(x_3)$, where x_1, x_2, x_3 are the Cartesian components of the vector $\mathbf{x}$.

Substituting (2.30)–(2.31) into (2.15), we get

$$0 = \int_0^T \int \int \left(M(q,r,s)\frac{D\mathbf{u}}{Dt}(\mathbf{x},t) - \mathbf{F}(q,r,s,t) \right) \cdot \mathbf{v}(\mathbf{x},t)$$
$$\delta\left(\mathbf{x} - \mathbf{X}(q,r,s,t)\right) d\mathbf{x}\, dq\, dr\, ds\, dt. \quad (2.32)$$

Note that this expression contains a mixture of Lagrangian and Eulerian variables. The Lagrangian variables that remain in it are the mass density $M(q,r,s)$ and the elastic force density $\mathbf{F}(q,r,s,t)$. We can eliminate these (and get rid of the integral over q,r,s) by making the following definitions:

$$\rho(\mathbf{x},t) = \int M(q,r,s)\,\delta\left(\mathbf{x} - \mathbf{X}(q,r,s,t)\right) dq\, dr\, ds, \quad (2.33)$$

$$\mathbf{f}(\mathbf{x},t) = \int \mathbf{F}(q,r,s,t)\,\delta\left(\mathbf{x} - \mathbf{X}(q,r,s,t)\right) dq\, dr\, ds. \quad (2.34)$$

To see the meaning of the Eulerian variables $\rho(\mathbf{x},t)$ and $\mathbf{f}(\mathbf{x},t)$ defined by these two equations, integrate both sides of each equation over an arbitrary region Ω. Since $\int_\Omega \delta\left(\mathbf{x} - \mathbf{X}(q,r,s,t)\right) d\mathbf{x}$ is equal to 1 or 0 depending on whether or not $\mathbf{X}(q,r,s,t) \in \Omega$, the results are

$$\int_\Omega \rho(\mathbf{x},t)\, d\mathbf{x} = \int_{\mathbf{X}(q,r,s,t)\in\Omega} M(q,r,s)\, dq\, dr\, ds, \quad (2.35)$$

$$\int_\Omega \mathbf{f}(\mathbf{x},t)\, d\mathbf{x} = \int_{\mathbf{X}(q,r,s,t)\in\Omega} \mathbf{F}(q,r,s,t)\, dq\, dr\, ds. \quad (2.36)$$

These equations show that ρ is the Eulerian mass density and that $\mathbf{f}$ is the Eulerian density of the elastic force.

Making use of the definitions of ρ and $\mathbf{f}$, we see that (2.32) can be rewritten

$$0 = -\wp S = \int_0^T \int \left(\rho(\mathbf{x},t)\frac{D\mathbf{u}}{Dt}(\mathbf{x},t) - \mathbf{f}(\mathbf{x},t) \right) \cdot \mathbf{v}(\mathbf{x},t)\, d\mathbf{x}\, dt. \quad (2.37)$$

This completes the transition from Lagrangian to Eulerian variables.

Equation (2.37) holds for arbitrary $\mathbf{v}(\mathbf{x},t)$ subject to the constraints. As shown above, the constraints on $\mathbf{v}(\mathbf{x},t)$ are that $\mathbf{v}(\mathbf{x},0) = \mathbf{v}(\mathbf{x},T) = 0$ and that $\nabla \cdot \mathbf{v} = 0$. Otherwise, $\mathbf{v}$ is arbitrary.

We now appeal to the Hodge decomposition: an arbitrary vector field may be written as the sum of a gradient and a divergence-free vector field. In particular, it is always possible to write

$$\rho\frac{D\mathbf{u}}{Dt} - \mathbf{f} = -\nabla p + \mathbf{w}, \quad (2.38)$$

where

$$\nabla \cdot \mathbf{w} = 0. \quad (2.39)$$

We shall show that $\mathbf{w} = 0$. To do so, we make use of the freedom in the choice of $\mathbf{v}$ by setting

$$\mathbf{v}(\mathbf{x}, t) = \phi(t)\mathbf{w}(\mathbf{x}, t). \tag{2.40}$$

Clearly, this satisfies all of the constraints on $\mathbf{v}$ provided that $\phi(0) = \phi(T) = 0$. This leaves a lot of freedom in the choice of ϕ, and we use this freedom to make $\phi(t) > 0$ for all $t \in (0, T)$.

With these choices, (2.37) becomes

$$0 = \int_0^T \phi(t) \int \left(-\nabla p(\mathbf{x}, t) + \mathbf{w}(\mathbf{x}, t)\right) \cdot \mathbf{w}(\mathbf{x}, t) \, d\mathbf{x} \, dt. \tag{2.41}$$

Now the term involving ∇p drops out, as can easily be shown using integration by parts, since $\nabla \cdot \mathbf{w} = 0$. This leaves us with

$$0 = \int_0^T \phi(t) \int |\mathbf{w}(\mathbf{x}, t)|^2 \, d\mathbf{x} \, dt. \tag{2.42}$$

Recalling that ϕ is positive on $(0, T)$, we conclude that $\mathbf{w} = 0$, as claimed.

In conclusion, we have the IB form of the equations of motion of an incompressible elastic material. We shall collect these equations here, taking the liberty of adding a viscous term which was omitted in the derivation (since the principle of least action applies to conservative systems) but which is important in applications. We assume a uniform viscosity of the kind that appears in a Newtonian fluid, although this could of course be generalized. Thus, the viscous term that we add is of the form $\mu \Delta \mathbf{u}$, where μ is the (dynamic) viscosity, and Δ is the Laplace operator (which is applied separately to each Cartesian component of the velocity field $\mathbf{u}$).

The equations of motion with the viscous term thrown in but otherwise as derived above read as follows:

$$\rho\left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u}\right) + \nabla p = \mu \Delta \mathbf{u} + \mathbf{f}, \tag{2.43}$$

$$\nabla \cdot \mathbf{u} = 0, \tag{2.44}$$

$$\rho(\mathbf{x}, t) = \int M(q, r, s)\, \delta\left(\mathbf{x} - \mathbf{X}(q, r, s, t)\right) dq\, dr\, ds, \tag{2.45}$$

$$\mathbf{f}(\mathbf{x}, t) = \int \mathbf{F}(q, r, s, t)\, \delta\left(\mathbf{x} - \mathbf{X}(q, r, s, t)\right) dq\, dr\, ds, \tag{2.46}$$

$$\frac{\partial \mathbf{X}}{\partial t}(q, r, s, t) = \mathbf{u}\left(\mathbf{X}(q, r, s, t), t\right)$$
$$= \int \mathbf{u}(\mathbf{x}, t)\, \delta\left(\mathbf{x} - \mathbf{X}(q, r, s, t)\right) d\mathbf{x}, \tag{2.47}$$

$$\mathbf{F} = -\frac{\wp E}{\wp \mathbf{X}}. \tag{2.48}$$

In these equations, $M(q, r, s)$ is a given function, the Lagrangian mass density of the material, and $E[\mathbf{X}]$ is a given functional, the elastic potential energy of the material in configuration $\mathbf{X}$. The notation $\wp E/\wp\mathbf{X}$ is shorthand for the Fréchet derivative of E.

Note that (2.43)–(2.44) are completely in Eulerian form. In fact, they are the equations of a viscous incompressible fluid of non-uniform mass density $\rho(\mathbf{x}, t)$ subject to an applied body force (*i.e.*, a force per unit volume) $\mathbf{f}(\mathbf{x}, t)$. Equation (2.48) is completely in Lagrangian form. It expresses the elasticity of the material.

Equations (2.45)–(2.47) are interaction equations. They convert from Lagrangian to Eulerian variables (equations (2.45)–(2.46)) and *vice versa* (equation (2.47)). The first line of (2.47) is just the definition of the Eulerian velocity field, and the second line uses nothing more than the defining property of the Dirac delta function. All of the interaction equations involve integral operators with the same kernel $\delta(\mathbf{x} - \mathbf{X}(q, r, s, t))$ but there is this subtle difference between (2.45)–(2.46) and (2.47). The interaction equations that define ρ and $\mathbf{f}$ are relationships between corresponding densities: the numerical values of ρ and M are *not* the same at corresponding points, and similarly the numerical values of $\mathbf{f}$ and $\mathbf{F}$ are *not* the same at corresponding points. But the numerical value of $\partial\mathbf{X}/\partial t$ and $\mathbf{u}(\mathbf{x}, t)$ *are* the same at corresponding points, as stated explicitly on the first line of (2.47). This difference comes about because of the mapping $(q, r, s) \rightarrow \mathbf{X}(q, r, s, t)$, which appears within the argument of the Dirac delta function. This complicates the integrals over q, r, s in (2.45)–(2.46); there is no such complication in the integral over $\mathbf{x}$ in (2.47). The distinction that we have just described is especially important in a special case that we shall discuss later, in which the elastic material is confined to a surface immersed in a three-dimensional fluid. Then ρ and $\mathbf{f}$ become singular, while $\partial\mathbf{X}/\partial t$ remains finite.

A feature of the foregoing equations of motion that the reader may find puzzling is that they do not include the equation

$$\frac{\partial\rho}{\partial t} + \mathbf{u} \cdot \nabla\rho = 0, \tag{2.49}$$

which asserts that the mass density at any given material point is independent of time, *i.e.*, that $D\rho/Dt = 0$. In fact (2.49) is a consequence of (2.44), (2.45) and (2.47), as we now show.

Applying $\partial/\partial t$ to both sides of (2.45), we get

$$\frac{\partial\rho}{\partial t} = -\int M(q, r, s)\frac{\partial\mathbf{X}}{\partial t}(q, r, s, t) \cdot \nabla\delta(\mathbf{x} - \mathbf{X}(q, r, s, t))\,\mathrm{d}q\,\mathrm{d}r\,\mathrm{d}s$$

$$= -\int M(q, r, s)\mathbf{u}(\mathbf{X}(q, r, s, t), t) \cdot \nabla\delta(\mathbf{x} - \mathbf{X}(q, r, s, t))\,\mathrm{d}q\,\mathrm{d}r\,\mathrm{d}s. \tag{2.50}$$

On the other hand,

$$\mathbf{u} \cdot \nabla \rho(\mathbf{x}, t) = \int M(q, r, s) \mathbf{u}(\mathbf{x}, t) \cdot \nabla \delta\left(\mathbf{x} - \mathbf{X}(q, r, s, t)\right) \mathrm{d}q \, \mathrm{d}r \, \mathrm{d}s. \quad (2.51)$$

To complete the proof, we need the equation

$$\mathbf{u}(\mathbf{x}) \cdot \nabla \delta(\mathbf{x} - \mathbf{X}) = \mathbf{u}(\mathbf{X}) \cdot \nabla \delta(\mathbf{x} - \mathbf{X}). \quad (2.52)$$

This is actually not true in general, but it is true when $\nabla \cdot \mathbf{u} = 0$. In that case we have, for any suitable test function ϕ,

$$\nabla \cdot (\mathbf{u}\phi) = \mathbf{u} \cdot \nabla \phi, \quad (2.53)$$

$$\nabla \cdot (\mathbf{u}\phi)(\mathbf{X}) = \mathbf{u}(\mathbf{X}) \cdot \nabla \phi(\mathbf{X}), \quad (2.54)$$

$$\int \nabla \cdot (\mathbf{u}\phi)(\mathbf{x}) \delta(\mathbf{x} - \mathbf{X}) \, \mathrm{d}\mathbf{x} = \mathbf{u}(\mathbf{X}) \cdot \int \nabla \phi(\mathbf{x}) \delta(\mathbf{x} - \mathbf{X}) \, \mathrm{d}\mathbf{x}, \quad (2.55)$$

$$-\int \phi(\mathbf{x}) \mathbf{u}(\mathbf{x}) \cdot \nabla \delta(\mathbf{x} - \mathbf{X}) \, \mathrm{d}\mathbf{x} = -\mathbf{u}(\mathbf{X}) \cdot \int \phi(\mathbf{x}) \nabla \delta(\mathbf{x} - \mathbf{X}) \, \mathrm{d}\mathbf{x}. \quad (2.56)$$

Since ϕ is arbitrary, this implies (2.52). Then, with the help of (2.52), it is easy to combine (2.50)–(2.51) to yield (2.49).

3. Fluid–structure interaction

In the previous section, we envisioned a viscoelastic incompressible material filling all of space. Let us now consider the situation in which only part of space is filled by the viscoelastic material and the rest by a viscous incompressible fluid. (As before, we assume that the viscosity μ is constant everywhere.) This actually requires no change at all in the equations of motion as we have formulated them. The only change is in the elastic energy functional $E[\mathbf{X}(\ ,\ ,\ ,t)]$, which now includes no contribution from those points (q, r, s) that are in the fluid regions. As a result of this, the Lagrangian elastic force density $\mathbf{F} = -\wp E / \wp \mathbf{X}$ is zero at such fluid points (q, r, s), and the Eulerian elastic force density $\mathbf{f}(\mathbf{x}, t)$ is zero at all space points $\mathbf{x}$ that happen to lie in fluid regions at time t.

Note that the Lagrangian coordinates (q, r, s) are still used in the fluid regions, since they carry the mass density $M(q, r, s)$, from which $\rho(\mathbf{x}, t)$ is computed. This formulation allows for the possibility of a *stratified* fluid, in which $\rho(\mathbf{x}, t)$ is non-uniform, even in the fluid regions.

If, on the other hand, the fluid has a uniform mass density ρ_0, then we can dispense with the Lagrangian variables in the fluid regions and let the Lagrangian coordinates (q, r, s) correspond to elastic material points only. In that case, in the spirit of Archimedes, we replace the Lagrangian mass

density $M(q, r, s)$ by the *excess* Lagrangian mass density $\tilde{M}(q, r, s)$ such that

$$\rho(\mathbf{x}, t) = \rho_0 + \int \tilde{M}(q, r, s) \, \delta \left(\mathbf{x} - \mathbf{X}(q, r, s, t) \right) \mathrm{d}q \, \mathrm{d}r \, \mathrm{d}s. \qquad (3.1)$$

Note that $\tilde{M}$ may be negative; that would represent elastic material that is less dense than the ambient fluid. The integral of $\tilde{M}$ over any part of the elastic material is the difference between the mass of that elastic material and the mass of the fluid displaced by it.

Now we come to the important special case that gives the immersed *boundary* method its name. Suppose that all or part of the elastic material is confined to certain surfaces that are immersed in fluid. Examples include heart valve leaflets, parachute canopies, thin airfoils (including bird, insect, or bat wings), sails, kites, flags, and weather vanes. For any such surface, only two Lagrangian parameters, say (r, s) are needed. Then the interaction equations become

$$\rho(\mathbf{x}, t) = \rho_0 + \int M(r, s) \, \delta \left(\mathbf{x} - \mathbf{X}(r, s, t) \right) \mathrm{d}r \, \mathrm{d}s, \qquad (3.2)$$

$$\mathbf{f}(\mathbf{x}, t) = \int \mathbf{F}(r, s, t) \, \delta \left(\mathbf{x} - \mathbf{X}(r, s, t) \right) \mathrm{d}r \, \mathrm{d}s, \qquad (3.3)$$

$$\frac{\partial \mathbf{X}}{\partial t} = \mathbf{u} \left(\mathbf{X}(r, s, t), t \right) = \int \mathbf{u}(\mathbf{x}, t) \, \delta \left(\mathbf{x} - \mathbf{X}(r, s, t) \right) \mathrm{d}\mathbf{x}. \qquad (3.4)$$

Note that there is no distinction here between $M(r, s)$ and $\tilde{M}(r, s)$, since the volume displaced by an elastic material *surface* is zero, by definition, so the excess mass of the immersed boundary is simply its mass. In (3.2) and (3.3), the delta function $\delta(\mathbf{x}) = \delta(x_1)\delta(x_2)\delta(x_3)$ is still three-dimensional, but there are only two integrals $\mathrm{d}r \, \mathrm{d}s$. As a result of this discrepancy, $\rho(\mathbf{x}, t)$ and $\mathbf{f}(\mathbf{x}, t)$ are each singular like a one-dimensional delta function, the singularity being supported on the immersed elastic boundary. Although $\rho(\mathbf{x}, t)$ and $\mathbf{f}(\mathbf{x}, t)$ are infinite on the immersed boundary, their integrals over any finite volume are finite. Specifically, the integral of $\rho(\mathbf{x}, t)$ over such a volume is the sum of the mass of the fluid contained within that volume and the mass of that part of the elastic boundary that is contained within that volume. Similarly, the integral of $\mathbf{f}(\mathbf{x}, t)$ over such a volume is the total force applied to the fluid by the part of the immersed boundary contained within the volume in question.

Unlike (3.2) and (3.3), the integral in (3.4) is three-dimensional and gives a finite result, the velocity of the immersed elastic boundary. Here again we see an important distinction between the interaction equations that convert from Lagrangian to Eulerian variables and the interaction equation that converts in the other direction.

An important remark is that the equations of motion that we have just derived for a viscous incompressible fluid containing an immersed elastic

boundary are mathematically equivalent to the conventional equations that one would write down involving the jump in the fluid stress across that boundary. This can be shown by integrating the Navier–Stokes equation across the boundary and noting the contribution from the delta function layers given by (3.2) and (3.3). For proofs of this kind, albeit in the case of a massless boundary, see Peskin and Printz (1993)and Lai and Li (2001).

In summary, the equations of motion derived in the previous section are easily adapted to problems of fluid–structure interaction, including problems involving stratified fluids (with or without structures) and also problems involving immersed elastic boundaries as well as immersed elastic solids.

4. Spatial discretization

This section begins the description of the *numerical* IB method. We shall present this method in the context of the equations of motion of an incompressible viscoelastic material, as derived above: see (2.43)–(2.48). The modifications of these equations needed for fluid–structure interaction have been described in the previous section and require corresponding minor changes in the numerical IB method. These will not be written out, since they are straightforward.

The plan that we shall follow in presenting the numerical IB method is to discuss spatial discretization first. Then (in the next section) we shall derive some identities satisfied by the spatially discretized scheme. Some of these motivate requirements that should be imposed on the approximate form of the Dirac delta function that is used in the interaction equations. The construction of that approximate delta function is discussed next, and finally we consider the temporal discretization of the scheme.

The spatial discretization that we shall describe employs two independent grids, one for the Eulerian and the other for the Lagrangian variables. The Eulerian grid, denoted g_h, is the set of points of the form $\mathbf{x} = \mathbf{j}h$, where $\mathbf{j} = (j_1, j_2, j_3)$ is a 3-vector with integer components. Similarly, the Lagrangian grid, denoted G_h is the set of (q, r, s) of the form $(k_q \Delta q, k_r \Delta r, k_s \Delta s)$, where $(k_q, k_r, k_s) = \mathbf{k}$ has integer components. To avoid leaks, we impose the restriction that

$$|\mathbf{X}(q + \Delta q, r, s, t) - \mathbf{X}(q, r, s, t)| < \frac{h}{2} \tag{4.1}$$

for all (q, r, s, t), and similarly for Δr and Δs.

In the continuous formulation, the elastic energy functional $E[\mathbf{X}]$ is typically an integral over a local energy density $\mathcal{E}$. Corresponding to this, we have in the discrete case

$$E_h = \sum_{\mathbf{k}'} \mathcal{E}_{\mathbf{k}'}(\cdots \mathbf{X}_{\mathbf{k}} \cdots) \, \Delta q \, \Delta r \, \Delta s. \tag{4.2}$$

Perturbing this, we find

$$\wp E_h = \sum_{\mathbf{k}} \sum_{\mathbf{k'}} \frac{\partial \mathcal{E}_{\mathbf{k'}}}{\partial \mathbf{X_k}} \cdot \wp \mathbf{X_k} \, \Delta q \, \Delta r \, \Delta s, \tag{4.3}$$

where $\partial / \partial \mathbf{X_k}$ denotes the gradient with respect to $\mathbf{X_k}$. This is of the form

$$\wp E_h = -\sum_{\mathbf{k}} \mathbf{F_k} \cdot \wp \mathbf{X_k} \, \Delta q \, \Delta r \, \Delta s, \tag{4.4}$$

provided that we set

$$\mathbf{F_k} = -\sum_{\mathbf{k'}} \frac{\partial \mathcal{E}_{\mathbf{k'}}}{\partial \mathbf{X_k}}. \tag{4.5}$$

Note that this is equivalent to

$$\mathbf{F_k} \, \Delta q \, \Delta r \, \Delta s = -\frac{\partial E_h}{\partial \mathbf{X_k}}. \tag{4.6}$$

Thus $\mathbf{F_k}$ is the discrete Lagrangian force density associated with the material point $\mathbf{k} = (k_q \Delta q, k_r \Delta r, k_s \Delta s)$, and $\mathbf{F_k} \, \Delta q \, \Delta r \, \Delta s$ is the actual *force* associated with that point. In a similar way, if $M_{\mathbf{k}}$ is the given Lagrangian mass density at the point $\mathbf{k}$, then $M_{\mathbf{k}} \, \Delta q \, \Delta r \, \Delta s$ is the actual *mass* associated with that point in the discrete formulation.

Let us now consider the spatial finite difference operators that are used on the Eulerian grid g_h. For the most part, these are built from the central difference operators $D_{h,\beta}^0$, $\beta = 1, 2, 3$, defined as follows:

$$\left(D_{h,\beta}^0 \right)(\mathbf{x}) = \frac{\phi(\mathbf{x} + h\mathbf{e}_\beta) - \phi(\mathbf{x} - h\mathbf{e}_\beta)}{2h}, \tag{4.7}$$

where $\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3$ are the standard basis vectors. We shall also use the notation that $\mathbf{D}_h^0$ is the vector difference operator whose components are $D_{h,\beta}^0$. Thus, $\mathbf{D}_h^0 \phi$ is the central difference approximation to the gradient of ϕ, and $\mathbf{D}_h^0 \cdot \mathbf{u}$ is the central difference approximation to the divergence of $\mathbf{u}$.

For the Laplacian that appears in the viscous terms, however, we do not use $\mathbf{D}_h^0 \cdot \mathbf{D}_h^0$, since this would entail a staggered stencil of total width equal to 4 meshwidths. Instead, we use the 'tight' Laplacian L_h defined as follows:

$$(L_h \phi)(\mathbf{x}) = \sum_{\beta=1}^{3} \frac{\phi(\mathbf{x} + h\mathbf{e}_\beta) + \phi(\mathbf{x} - h\mathbf{e}_\beta) - 2\phi(\mathbf{x})}{h^2}. \tag{4.8}$$

Finally, we consider the operator $\mathbf{u} \cdot \nabla$ which appears in the nonlinear terms of the Navier–Stokes equations. Since $\nabla \cdot \mathbf{u} = 0$, we have the identity

$$\mathbf{u} \cdot \nabla \phi = \nabla \cdot (\mathbf{u}\phi). \tag{4.9}$$

We take advantage of this by introducing the skew-symmetric difference operator $S_h(\mathbf{u})$ defined as follows:

$$S_h(\mathbf{u})\phi = \frac{1}{2}\mathbf{u} \cdot \mathbf{D}_h^0\phi + \frac{1}{2}\mathbf{D}_h^0 \cdot (\mathbf{u}\phi). \qquad (4.10)$$

Note that the two quantities that are averaged on the right-hand side are discretizations of the equal expressions $\mathbf{u} \cdot \nabla\phi$ and $\nabla \cdot (\mathbf{u}\phi)$, respectively. Nevertheless, the two discretizations are *not* in general equal to each other, not even if $\mathbf{D}_h^0 \cdot \mathbf{u} = 0$. That is because the product rule that holds for derivatives does not apply to the difference operator $\mathbf{D}_h^0$.

The skew symmetry of $S_h(\mathbf{u})$ is shown as follows. Let $(\,,\,)_h$ be the discrete inner product defined by

$$(\phi, \psi)_h = \sum_{\mathbf{x} \in g_h} \phi(\mathbf{x})\psi(\mathbf{x})h^3 \qquad (4.11)$$

for scalars, and

$$(\mathbf{u}, \mathbf{v})_h = \sum_{\mathbf{x} \in g_h} \mathbf{u}(\mathbf{x}) \cdot \mathbf{v}(\mathbf{x})h^3 \qquad (4.12)$$

for vectors. It is easy to show by 'summation by parts' (which is merely re-indexing) that

$$\left(\phi, \mathbf{D}_h^0 \cdot \mathbf{u}\right)_h = -\left(\mathbf{D}_h^0\phi, \mathbf{u}\right)_h. \qquad (4.13)$$

In other words, $\mathbf{D}_h^0$ is skew-symmetric with respect to this inner product. With the help of this identity, we see that

$$\begin{aligned}
(\psi, S_h(\mathbf{u})\phi)_h &= \frac{1}{2}\left(\psi, \mathbf{u} \cdot \mathbf{D}_h^0\phi\right)_h + \frac{1}{2}\left(\psi, \mathbf{D}_h^0 \cdot (\mathbf{u}\phi)\right)_h \\
&= \frac{1}{2}\left(\mathbf{u}\psi, \mathbf{D}_h^0\phi\right)_h - \frac{1}{2}\left(\mathbf{D}_h^0\psi, \mathbf{u}\phi\right)_h \\
&= -\frac{1}{2}\left(\mathbf{D}_h^0 \cdot (\mathbf{u}\psi), \phi\right)_h - \frac{1}{2}\left(\mathbf{u} \cdot \mathbf{D}_h^0\psi, \phi\right)_h \\
&= -(S_h(\mathbf{u})\psi, \phi)_h, \qquad (4.14)
\end{aligned}$$

so $S_h(\mathbf{u})$ is indeed skew-symmetric, as claimed. Note that the skew symmetry of $S_h(\mathbf{u})$ holds for all $\mathbf{u}$; it does not require $\mathbf{D}_h^0 \cdot \mathbf{u} = 0$. This completes the discussion of the difference operators that are needed on the Eulerian grid g_h.

Finally, we consider the interaction equations, which involve the Dirac delta function via the kernel $\delta(\mathbf{x} - \mathbf{X}(q, r, s, t))$. The integrals in these equations are, of course, replaced by sums over the appropriate grid (g_h or G_h), but what about the delta function itself? What is clearly needed is a

function $\delta_h(\mathbf{x})$ that is nonsingular for each h but approaches $\delta(\mathbf{x})$ as $h \to 0$. There are, of course, many ways to construct such δ_h, and we shall describe the method of construction that we recommend in a subsequent section. For now, however, we proceed on the assumption that the function $\delta_h(\mathbf{x})$ is known.

With the apparatus introduced above, we may spatially discretize (2.43)–(2.48) as follows:

$$\rho\left(\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} + S_h(\mathbf{u})\mathbf{u}\right) + \mathbf{D}_h^0 p = \mu L_h \mathbf{u} + \mathbf{f}, \tag{4.15}$$

$$\mathbf{D}_h^0 \cdot \mathbf{u} = 0, \tag{4.16}$$

$$\rho(\mathbf{x}, t) = \sum_{(q,r,s) \in G_h} M(q,r,s)\, \delta_h\left(\mathbf{x} - \mathbf{X}(q,r,s,t)\right) \Delta q\, \Delta r\, \Delta s, \tag{4.17}$$

$$\mathbf{f}(\mathbf{x}, t) = \sum_{(q,r,s) \in G_h} \mathbf{F}(q,r,s,t)\, \delta_h\left(\mathbf{x} - \mathbf{X}(q,r,s,t)\right) \Delta q\, \Delta r\, \Delta s, \tag{4.18}$$

$$\frac{\mathrm{d}\mathbf{X}}{\mathrm{d}t}(q,r,s,t) = \sum_{\mathbf{x} \in g_h} \mathbf{u}(\mathbf{x}, t)\, \delta_h\left(\mathbf{x} - \mathbf{X}(q,r,s,t)\right) h^3, \tag{4.19}$$

$$\mathbf{F}(q,r,s,t)\, \Delta q\, \Delta r\, \Delta s = -\frac{\partial}{\partial \mathbf{X}(q,r,s)} E_h\left(\cdots \mathbf{X}(q',r',s',t) \cdots\right). \tag{4.20}$$

These comprise a system of ordinary differential equations, with $\mathbf{x}$ restricted to g_h, and with (q,r,s) or (q',r',s') restricted to G_h.

5. Eulerian/Lagrangian identities

First, we show that total momentum and total power each come out the same whether they are evaluated in Eulerian or in Lagrangian form. These two identities require no assumptions at all on the form of δ_h. They are important because they reveal the dual nature of the operations defined by (4.17) and (4.19), and similarly between the operations defined by (4.18) and (4.19). The operations that take Lagrangian input and generate Eulerian output (equations (4.17) and (4.18)) define relationships between corresponding densities, and we shall refer to them as 'spreading' operations. Thus (4.17) spreads the Lagrangian mass out onto the Eulerian grid, and the operation defined by (4.18) does the same for the Lagrangian force. The operation defined by (4.19), on the other hand, is best described as 'interpolation', since it averages the numerical values on nearby Eulerian grid points to produce a corresponding numerical value at a Lagrangian material point. The mathematical content of the two identities that we shall now derive is that spreading is the *adjoint* of interpolation.

Starting from the Eulerian formula for the momentum, we find

$$\sum_{\mathbf{x}\in g_h} \rho(\mathbf{x},t)\mathbf{u}(\mathbf{x},t)h^3$$

$$= \sum_{\mathbf{x}\in g_h}\sum_{(q,r,s)\in G_h} M(q,r,s)\,\delta_h\left(\mathbf{x}-\mathbf{X}(q,r,s,t)\right)\Delta q\,\Delta r\,\Delta s\,\mathbf{u}(\mathbf{x},t)\,h^3$$

$$= \sum_{(q,r,s)\in G_h} M(q,r,s)\frac{d\mathbf{X}}{dt}(q,r,s,t)\,\Delta q\,\Delta r\,\Delta s, \tag{5.1}$$

which is the corresponding Lagrangian formula for the momentum. In the first step we used (4.17) to eliminate ρ and in the second step we used (4.19) to eliminate $\mathbf{u}$. Note that the argument only works because the same kernel $\delta_h\left(\mathbf{x}-\mathbf{X}(q,r,s,t)\right)$ is used the spreading step and in the interpolation step.

In exactly the same way, starting from the Eulerian formula for the power that is generated by the elastic force, we find

$$\sum_{\mathbf{x}\in g_h}\mathbf{f}(\mathbf{x},t)\cdot\mathbf{u}(\mathbf{x},t)\,h^3$$

$$= \sum_{\mathbf{x}\in g_h}\sum_{(q,r,s)\in G_h}\mathbf{F}(q,r,s,t)\,\delta_h\left(\mathbf{x}-\mathbf{X}(q,r,s,t)\right)\Delta q\,\Delta r\,\Delta s\cdot\mathbf{u}(\mathbf{x},t)h^3$$

$$= \sum_{(q,r,s)\in G_h}\mathbf{F}(q,r,s,t)\cdot\frac{d\mathbf{X}}{dt}(q,r,s,t)\,\Delta q\,\Delta r\,\Delta s, \tag{5.2}$$

which is the corresponding Lagrangian expression. Here we used (4.18) and (4.19) in the same manner as above. Again, the argument only works because the same δ_h appears in both interaction equations.

Next, we show that mass, force, and torque all come out the same whether they are evaluated in terms of Eulerian or Lagrangian variables. These identities differ from those derived above in that they rely on certain properties of the function δ_h. We state these properties now, so that we may use them as needed in the derivations that follow. They will play a central role in the construction of δ_h as described in the following section. The two key properties that we need to impose on δ_h to obtain the mass, force, and torque identities are:

$$\sum_{\mathbf{x}\in g_h}\delta_h(\mathbf{x}-\mathbf{X})h^3 = 1, \quad \text{all } \mathbf{X}, \tag{5.3}$$

$$\sum_{\mathbf{x}\in g_h}(\mathbf{x}-\mathbf{X})\,\delta_h(\mathbf{x}-\mathbf{X})h^3 = 0, \quad \text{all } \mathbf{X}. \tag{5.4}$$

It is important that these properties hold for all real shifts $\mathbf{X}$, not merely for $\mathbf{X}\in g_h$. Note that these properties are the discrete analogues of $\int\delta(\mathbf{x}-\mathbf{X})\,d\mathbf{x}=1$ and $\int(\mathbf{x}-\mathbf{X})\,\delta(\mathbf{x}-\mathbf{X})\,d\mathbf{x}=0$. Finally, we remark that the two

properties stated above can be combined to yield

$$\sum_{\mathbf{x}\in g_h} \mathbf{x}\, \delta_h(\mathbf{x}-\mathbf{X})h^3 = \mathbf{X}, \quad \text{all } \mathbf{X}. \tag{5.5}$$

The mass and force identities need only (5.3) and are derived as follows:

$$\sum_{\mathbf{x}\in g_h} \rho(\mathbf{x},t)h^3$$
$$= \sum_{\mathbf{x}\in g_h} \sum_{(q,r,s)\in G_h} M(q,r,s)\, \delta_h\left(\mathbf{x}-\mathbf{X}(q,r,s,t)\right) \Delta q\, \Delta r\, \Delta s\, h^3$$
$$= \sum_{(q,r,s)\in G_h} M(q,r,s)\, \Delta q\, \Delta r\, \Delta s. \tag{5.6}$$

By the way, this establishes conservation of mass, since the last line is independent of time. Exactly the same manipulations yield the force identity:

$$\sum_{\mathbf{x}\in g_h} \mathbf{f}(\mathbf{x},t)h^3$$
$$= \sum_{\mathbf{x}\in g_h} \sum_{(q,r,s)\in G_h} \mathbf{F}(q,r,s,t)\, \delta_h\left(\mathbf{x}-\mathbf{X}(q,r,s,t)\right) \Delta q\, \Delta r\, \Delta s\, h^3$$
$$= \sum_{(q,r,s)\in G_h} \mathbf{F}(q,r,s,t)\, \Delta q\, \Delta r\, \Delta s. \tag{5.7}$$

The torque identity is similar but relies on (5.5):

$$\sum_{\mathbf{x}\in g_h} \mathbf{x}\times \mathbf{f}(\mathbf{x},t)h^3$$
$$= \sum_{\mathbf{x}\in g_h} \sum_{(q,r,s)\in G_h} \mathbf{x}\times \mathbf{F}(q,r,s,t)\, \delta_h\left(\mathbf{x}-\mathbf{X}(q,r,s,t)\right) \Delta q\, \Delta r\, \Delta s\, h^3$$
$$= \sum_{(q,r,s)\in G_h} \mathbf{X}(q,r,s,t)\times \mathbf{F}(q,r,s,t)\, \Delta q\, \Delta r\, \Delta s. \tag{5.8}$$

In summary, we have established the equivalence of the Lagrangian and Eulerian expressions for mass, momentum, force, torque and power. (There do not seem to be similar identities for angular momentum or kinetic energy, though the reader is welcome to look for them.) The mass identity also establishes *conservation* of mass, since the Lagrangian expression for the mass is time-independent. The significance of the force, torque, and power identities is that momentum, angular momentum, and energy are not spuriously created or destroyed by the interaction equations. This is not the same thing as saying that these quantities are conserved, since that will depend on other aspects of the numerical scheme.

We conclude this section by showing in an important special case that momentum and energy are in fact conserved by the spatially discretized IB method. (In the case of energy, the rate of production of heat by viscosity has to be taken into account.) The special case that we consider is the one in which the whole system has a uniform mass density ρ. In other words, the immersed elastic material is neutrally buoyant in the ambient fluid, or it occupies so little volume that its mass can be neglected. In that case, we can dispense with (4.17) and instead treat ρ as a given constant.

Conservation of momentum requires the additional hypothesis that the elastic energy functional E be translation invariant, that is,

$$E[\mathbf{X}(\,,\,,\,,t) + \mathbf{Z}] = E[\mathbf{X}(\,,\,,\,,t)], \tag{5.9}$$

where $\mathbf{Z}$ is an arbitrary fixed shift, independent of the Lagrangian parameters (q, r, s). It is easy to show that this is equivalent to the condition that the total force generated by the elastic energy be identically zero. Thus, we assume here that

$$\sum_{(q,r,s)\in G_h} \mathbf{F}(q, r, s, t)\,\Delta q\,\Delta r\,\Delta s = 0. \tag{5.10}$$

According to our force identity, (5.7), this property is inherited by the Eulerian force density so that

$$\sum_{\mathbf{x}\in g_h} \mathbf{f}(\mathbf{x}, t)h^3 = 0. \tag{5.11}$$

Now, since ρ is constant, we can find the rate of change of momentum simply by summing (4.15) over the entire Eulerian grid. It is easy to see that this yields zero for all three of the terms $\rho S_h(\mathbf{u})\mathbf{u}$, $\mathbf{D}_h^0 p$, and $\mu L_h\mathbf{u}$ (recall that μ is constant throughout this paper). The reason is that summing over the grid is equivalent to taking the inner product with a constant. The operators involved in these three terms are all either antisymmetric or symmetric. Thus, they can be made to apply to the constant, where they all yield zero. The conclusion is that

$$\frac{\mathrm{d}}{\mathrm{d}t} \sum_{\mathbf{x}\in g_h} \rho\mathbf{u}(\mathbf{x}, t)h^3 = 0, \tag{5.12}$$

which is conservation of momentum.

Conservation of energy does not require a translation-invariant elastic energy function, so we no longer make use of (5.9)–(5.11). Taking the inner product of both sides of (4.15) with $\mathbf{u}$, we find

$$\frac{\mathrm{d}}{\mathrm{d}t}\frac{1}{2}\rho(\mathbf{u}, \mathbf{u})_h = -\rho(\mathbf{u}, S_h(\mathbf{u})\mathbf{u})_h - (\mathbf{u}, \mathbf{D}_h^0 p)_h + \mu(\mathbf{u}, L_h\mathbf{u})_h + (\mathbf{u}, \mathbf{f})_h. \tag{5.13}$$

Now S_h and $\mathbf{D}_h^0$ are both skew-symmetric, so the term involving S_h is immediately zero, and the term involving $\mathbf{D}_h^0$ can be rewritten $(\mathbf{D}_h^0 \cdot \mathbf{u}, p)$, which

Having said this, we can state the rest of our postulates in terms of the function $\phi(r)$, where r can denote x_1/h, x_2/h, or x_3/h. These postulates are:

$$\phi(r) \text{ is continuous for all real } r, \tag{6.2}$$

$$\phi(r) = 0 \text{ for } |r| \geq 2, \tag{6.3}$$

$$\sum_{j \text{ even}} \phi(r - j) = \sum_{j \text{ odd}} \phi(r - j) = \frac{1}{2} \text{ for all real } r, \tag{6.4}$$

$$\sum_j (r - j)\phi(r - j) = 0 \text{ for all real } r, \tag{6.5}$$

$$\sum_j (\phi(r - j))^2 = C \text{ for all real } r, \tag{6.6}$$

where the constant C is independent of r. Its numerical value will be determined later. Even though the sums are over integer values of j, it is important that the above equations hold for all real shifts r, not just for integer values of r.

For future reference, we note that (6.4) and (6.5) can be combined to yield

$$\sum_j j\phi(r - j) = r \text{ for all real } r. \tag{6.7}$$

Before solving the above equations for $\phi(r)$, let us consider the motivation for each of these postulates in turn. The product form given by (6.1) is not essential, but it makes a tremendous simplification, since it reduces all subsequent considerations to the one-dimensional case. Also, the scaling introduced in (6.1) has the simplifying effect of removing the parameter h from all subsequent formulae. Moreover, this scaling is the natural one, so that $\delta_h \to \delta$ as $h \to 0$.

Generally speaking, the purpose of the other postulates is to hide the presence of the Eulerian computational lattice as much as possible. The continuity of ϕ, for example, is imposed to avoid jumps in velocity or in applied force as the Lagrangian marker points cross over the Eulerian grid planes.

The postulate of bounded support (equation (6.3)), however, is made for a different reason: computational efficiency. There are functions with unbounded support that one might well like to use for $\phi(r)$, such as $\exp(-r^2/2)$ or $\sin(r)/r$, but these would entail enormous computational cost, since *each* Lagrangian marker point would then interact directly with *all* grid points of the Eulerian computational lattice. The support of $\phi(r)$ as specified in (6.3) is the smallest possible while maintaining consistency with the other postulates.

An immediate consequence of (6.4) is that

$$\sum_j \phi(r - j) = 1 \text{ for all real } r, \tag{6.8}$$

and this, in turn, implies the identity (5.3), which was used in the derivation of the mass, force, and torque identities of the previous section. What requires further explanation, however, is why we impose the stronger conditions that the sum over the even grid points should separately give $1/2$, and similarly the sum over odd grid points. This is a technical issue relating to the use of the central difference operator $\mathbf{D}^0$ in our numerical scheme. The null space of the gradient operator based on $\mathbf{D}^0$ is eight-dimensional. It contains not only the constants but any function that is constant on each of the 8 'chains' of points $\{j_1 \text{ even}, j_2 \text{ even}, j_3 \text{ even}\}$, $\{j_1 \text{ even}, j_2 \text{ even}, j_3 \text{ odd}\}$, *etc.* The separate conditions given in (6.4) ensure that all eight chains get the same amount of force from each Lagrangian marker point, and also that each Lagrangian marker point assigns equal weight to all eight chains when computing its interpolated velocity. This avoids oscillations from one grid point to the next that would otherwise occur, especially when localized forces are applied.

Equation (6.5) is postulated because it implies the three-dimensional identity (5.4), which was used to derive the torque identity in the previous section. Notice, too, that (6.4) and (6.5) together imply that, when δ_h is used for interpolation, linear functions are interpolated exactly, and smooth functions are therefore interpolated with second-order accuracy. Having said this, however, we must also point out that, in those cases in which the IB method is used with true boundaries (*i.e.*, immersed elastic structures of zero thickness), the velocity field that must be interpolated at the boundary is *not*, in fact, smooth. It is continuous but suffers a jump in its normal derivative. In this situation, the interpolation formula involving δ_h is only first-order accurate. The 'immersed interface method' (LeVeque and Li 1994, LeVeque and Li 1997, Li and Lai 2001) overcomes this difficulty.

The final postulate, (6.6), is different in character from the others and requires more explanation. It comes from a condition that we would like to impose but cannot (for a reason that will be explained below), namely

$$\sum_j \phi(r_1 - j)\phi(r_2 - j) = \Phi(r_1 - r_2) \text{ for all real } r_1, r_2, \tag{6.9}$$

where Φ is some other function related to ϕ. If a condition of this form could be imposed (regardless of the function Φ), it would imply (proof left as an exercise for the reader) the exact translation invariance of the IB method as applied to a translation-invariant linear system like the Stokes equations, that is, the IB results would remain exactly the same despite shifts in position of all immersed elastic structures by fixed (possibly fractional)

amounts relative to the Eulerian grid. This would be the ultimate in 'hiding the grid'. Even though we are solving the nonlinear Navier–Stokes equations, exact translation invariance in the case of the Stokes equations would suggest a good approximation to translation invariance in the case of the Navier–Stokes equations.

It is easy to prove, however, that (6.9) is incompatible with any condition of bounded support, like (6.3). The idea of the proof is to choose $r_1 - r_2$ such that the width of the overlap of the support of the functions of j given by $\phi(r_1 - j)$ and $\phi(r_2 - j)$ is some fraction θ that is strictly between 0 and 1. Then, by varying r_1 and r_2 in such a manner that their difference remains constant, one will find that sometimes the sum on the left-hand side of (6.9) contains exactly one nonzero term, so it cannot be zero, and sometimes that sum contains no nonzero terms, so it has to be zero. This contradicts the assumption that this sum depends only on $r_1 - r_2$.

Here is a more detailed version of the above argument. Suppose, for example, that $\phi(r) \neq 0$ for $a < r < b$, and that $\phi(r) = 0$ otherwise. Then $\phi(r_1 - j)\phi(r_2 - j)$ is nonzero if and only if

$$a < (r_1 - j) < b \ \text{and} \ a < (r_2 - j) < b, \tag{6.10}$$

which is equivalent to

$$r_1 - b < j < r_1 - a \ \text{and} \ r_2 - b < j < r_2 - a. \tag{6.11}$$

If $r_2 > r_1$, this is equivalent to

$$r_2 - b < j < r_1 - a. \tag{6.12}$$

Now fix the difference between r_2 and r_1 as follows:

$$r_2 = r_1 + (b - a) - \theta, \tag{6.13}$$

where $0 < \theta < 1$ and $\theta < (b - a)$. Under these conditions, $\phi(r_1 - j)\phi(r_2 - j)$ is nonzero if and only if

$$r_1 - a - \theta < j < r_1 - a. \tag{6.14}$$

Note that the interval which must contain the integer j is of length $\theta < 1$, so there is at most one integer j that satisfies this condition. By varying the real number r_1 while keeping $r_2 - r_1$ constant, we can clearly find cases where $\sum_j \phi(r_1 - j)\phi(r_2 - j)$ has exactly one nonzero term, and is therefore nonzero, and other cases in which that sum has no such term, and is therefore zero. This contradicts (6.9), which asserts that this sum depends only on the difference $r_2 - r_1$, which has indeed been held constant as we varied r_1.

Since (6.9) cannot be imposed, we content ourselves with the weaker condition that is derived from (6.9) by setting $r_1 = r_2$, namely

$$\sum_j (\phi(r - j))^2 = \Phi(0) = C, \tag{6.15}$$

which is (6.6). Although weaker than (6.9), our postulate (6.6) does give some information about the sum that appears on the left-hand side of (6.9), since it implies the inequality

$$\left| \sum_j \phi(r_1 - j)\phi(r_2 - j) \right| \leq C \text{ for all real } r_1, r_2 \tag{6.16}$$

with equality (see (6.9)) when $r_1 = r_2$. This is just the Schwarz inequality, since the left-hand side is the magnitude of an inner product, and the right-hand side $C = C^{1/2}C^{1/2}$ is the product of the norms of the vectors in question: see (6.9).

The significance of this inequality is as follows. Expressions like the sum that appears on the left-hand side arise when one considers the interaction between two Lagrangian marker points via the Eulerian grid. The inequality (6.16) guarantees that such coupling is strongest when the two Lagrangian points coincide, and moreover (according to (6.9)) that the self-coupling in that case is independent of the position of the Lagrangian marker with respect to the grid.

This completes the discussion of the motivation for the postulates given by (6.1)–(6.6). We now show how these postulates determine the form of ϕ and hence δ_h. The key step is to restrict r to the interval $[0, 1]$ and then to write out the equations for $\phi(r)$ explicitly as follows:

$$\phi(r - 2) + \phi(r) = \frac{1}{2}, \tag{6.17}$$

$$\phi(r - 1) + \phi(r + 1) = \frac{1}{2}, \tag{6.18}$$

$$2\phi(r - 2) + \phi(r - 1) - \phi(r + 1) = r, \tag{6.19}$$

$$(\phi(r - 2))^2 + (\phi(r - 1))^2 + (\phi(r))^2 + (\phi(r + 1))^2 = C. \tag{6.20}$$

Note that (6.19) is the transcription of (6.7).

These are 4 equations in the 4 unknowns $\phi(r - 2)$, $\phi(r - 1)$, $\phi(r)$, and $\phi(r + 1)$. (A complication is that we still do not know C, but that will be remedied shortly.) One way to solve this system is to use the first three (linear) equations to express all of the other unknowns in terms of $\phi(r)$. Then (6.20) becomes a quadratic equation for $\phi(r)$. The choice of root is uniquely determined by the continuity of ϕ and the condition that $\phi(\pm 2) = 0$. Here are the details.

The three linear equations yield:

$$\phi(r-2) = \frac{1}{2} - \phi(r), \tag{6.21}$$

$$\phi(r-1) = \frac{r}{2} - \frac{1}{4} + \phi(r), \tag{6.22}$$

$$\phi(r+1) = -\frac{r}{2} + \frac{3}{4} - \phi(r). \tag{6.23}$$

At this point we can determine C. Consider the special case $r = 0$. Then $\phi(r-2) = \phi(-2) = 0$ (see (6.2) and (6.3)). It follows immediately from the above results that $\phi(0) = \frac{1}{2}$ and $\phi(\pm 1) = \frac{1}{4}$. Substituting these results into (6.20), we get

$$C = 0 + \left(\frac{1}{4}\right)^2 + \left(\frac{1}{2}\right)^2 + \left(\frac{1}{4}\right)^2 = \frac{3}{8}. \tag{6.24}$$

Substituting all of these results into (6.20), we get the following quadratic equation for $\phi(r)$:

$$4\left(\phi(r)\right)^2 - (3-2r)\phi(r) + \frac{1}{2}(1-r)^2 = 0. \tag{6.25}$$

We may choose a root based on the requirement found above that $\phi(0) = \frac{1}{2}$. With this choice, the unique solution is

$$\phi(r) = \frac{3 - 2r + \sqrt{1 + 4r - 4r^2}}{8}, \quad 0 \le r \le 1. \tag{6.26}$$

It is important to note the restriction that this formula is only valid for $r \in [0,1]$. To determine ϕ on the intervals $[-2,-1]$, $[-1,0]$, and $[1,2]$, use (6.26) together with (6.21)–(6.23), always bearing in mind the restriction that all of these equations are only valid for $r \in [0,1]$. Simple changes of variable are needed to get explicit formulae for $\phi(r)$ on the other intervals. Finally, of course, we have by hypothesis that $\phi(r) = 0$ for $|r| \ge 2$. Combining all these results, we get

$$
\begin{aligned}
\phi(r) &= 0, \ r \le -2 \\
&= \frac{1}{8}\left(5 + 2r - \sqrt{-7 - 12r - 4r^2}\right), \quad -2 \le r \le -1 \\
&= \frac{1}{8}\left(3 + 2r + \sqrt{1 - 4r - 4r^2}\right), \quad -1 \le r \le 0 \\
&= \frac{1}{8}\left(3 - 2r + \sqrt{1 + 4r - 4r^2}\right), \quad 0 \le r \le 1 \\
&= \frac{1}{8}\left(5 - 2r - \sqrt{-7 + 12r - 4r^2}\right), \quad 1 \le r \le 2 \\
&= 0, \quad 2 \le r
\end{aligned}
\tag{6.27}
$$

Comparison of the corresponding formulae for negative and positive r reveal that $\phi(-r) = \phi(r)$, a property we did not postulate, although it is certainly expected. One can check that the function $\phi(r)$ defined by the above is not only continuous (as postulated) but actually has a continuous first derivative (a property we did not postulate but got for free) at the points $r = -2, -1, 0, 1, 2$, and indeed everywhere.

It may be worth mentioning that the function $\phi(r)$ that we have just constructed is *extremely* well approximated by the simple formula

$$\tilde{\phi}(r) = \begin{cases} \frac{1}{4}\left(1 + \cos\left(\frac{\pi r}{2}\right)\right), & |r| \leq 2, \\ 0, & \text{otherwise,} \end{cases} \tag{6.28}$$

which satisfies all of the above postulates except for the first-moment condition, (6.5). This function was discovered earlier and is still sometimes used in IB computations.

We conclude this section with some remarks on possible modifications of the above construction. First, it may be desirable to narrow the support of δ_h, both for reasons of computational cost, and to improve the resolution of the boundary. One way to do this is to dispense with the separate 'even/odd' conditions in (6.4) and just use (6.8) instead. This makes it possible to reduce the width of the support of $\phi(r)$ from 4 to 3. In a three-dimensional computation, this is a considerable saving, since it reduces the volume of the support of $\delta_h(\mathbf{x})$ from 64 to 27 cubic meshwidths. The construction of ϕ is similar to that given above, and the reader is invited to have the fun of working out the details. This 3-point delta function is only recommended, however, for use in conjunction with a numerical scheme that does not suffer from the problems described above concerning the eight-dimensional null space of the central difference gradient operator. An example of the use of the 3-point delta function can be found in the work of A. Roma (Roma, Peskin and Berger 1999), who used it in conjunction with the MAC scheme (Harlow and Welch 1965).

Another way to try to get better boundary resolution is to impose more moment conditions. This requires *broadening* the support of δ_h which sounds like going in the wrong direction. To see why it might work, though, consider the second-moment condition:

$$\sum_j (r - j)^2 \phi(r - j) = 0 \text{ for all real } r. \tag{6.29}$$

This essentially states that the mean square thickness of the boundary is zero! Note that the imposition of this condition requires $\phi(r)$ to change sign. Imposing higher moments has been investigated by J. Stockie (1997), who

has written a Maple program that automates the process, and independently by D. M. McQueen and the present author (unpublished). It turns out that the moments have to be introduced two at a time, and the introduction of each new pair of moments increases the width of the support of $\phi(r)$ by 2. Thus, starting from the 4-point delta function introduced above, one can impose second and third moment conditions and thereby arrive at a 6-point delta function, *etc.* The 6-point delta function has negative tails, and resembles the 'Mexican hat' functions that are used to describe receptive fields in natural and machine vision. Preliminary numerical experiments indicate some improvement in boundary resolution despite the increased width of the support of the delta function. Note that it is also possible to start from the 3-point delta function mentioned above and impose higher moment conditions, two at a time, to arrive at a 5-point delta function, a 7-point delta function, *etc.*

Another direction would be to aim at improved translation invariance. Here is a possible strategy, which has not yet been tried. Recall the condition given by (6.9) which would guarantee translation invariance (in the Stokes case), but which could not be imposed because it is incompatible with bounded support. Our strategy in the foregoing was simply to impose the special case of (6.9) that arises by setting $r_1 = r_2$. This resulted in our postulate given by (6.6). Now suppose that, instead of merely doing this, we also differentiate (6.9) one or more times before setting $r_1 = r_2$. It turns out that differentiating once gives nothing new, just a condition that can be derived from (6.6). But if we differentiate (6.9) twice, once with respect to r_1 and once with respect to r_2, and then set $r_1 = r_2 = r$, we get the new condition,

$$\sum_j \left(\phi'(r) \right)^2 = -\Phi''(0) = C_1 \text{ for all real } r, \qquad (6.30)$$

where $'$ denotes differentiation, and where C_1, like C, is a constant that has to be determined. It remains to be seen how to mix this condition in with the moment conditions, but that seems like a worthwhile subject of future investigation.

7. Temporal discretization

The temporal discretization that we currently use (Lai and Peskin 2000, McQueen and Peskin 2001) is a second-order accurate Runge–Kutta method, based primarily on the midpoint rule. For a system of ordinary differential equations of the form

$$\frac{\mathrm{d}y}{\mathrm{d}t} = f(y), \qquad (7.1)$$

such a scheme looks like this:

$$\frac{y^{n+\frac{1}{2}} - y^n}{\Delta t/2} = f(y^n), \tag{7.2}$$

$$\frac{y^{n+1} - y^n}{\Delta t} = f(y^{n+\frac{1}{2}}), \tag{7.3}$$

where the superscript is the time-step index. The salient feature of this scheme is that each time-step involves a 'preliminary substep' to the half-time level followed by a 'final substep' from time level n to $n + 1$, in which the results of the preliminary substep are used. The preliminary substep involves a first-order accurate scheme (forward Euler in the above example), and the final substep is done by a second-order accurate scheme (here, the midpoint rule). One would think that the accuracy would be limited by the least accurate substep, but the magic of Runge–Kutta is that the overall scheme is second-order accurate.

Our time-stepping scheme in the IB method follows the general idea of the above, but with some modifications. Instead of using pure forward Euler for the preliminary substep, we use a scheme that is a mixture of forward and backward Euler. In the final substep of each time-step, we use not only the midpoint rule but also (in one place) the trapezoidal rule. These changes are made to improve numerical stability by using implicit methods wherever that can be done at reasonable cost. But we retain the important feature that the preliminary substep is done by a first-order accurate method, and the final substep is done by a second-order accurate method that makes use of the results of the preliminary substep.

The preliminary substep, from time level n to $n + \frac{1}{2}$, proceeds as follows. First, find the positions of the Lagrangian markers at time level $n + \frac{1}{2}$:

$$\mathbf{X}^{n+\frac{1}{2}}(q,r,s) = \mathbf{X}^n(q,r,s) + \frac{\Delta t}{2} \sum_{\mathbf{x} \in g_h} \mathbf{u}^n(\mathbf{x}) \, \delta_h \left(\mathbf{x} - \mathbf{X}^n(q,r,s) \right) h^3. \tag{7.4}$$

Next, use the configuration of the Lagrangian markers at the half-time level to calculate the elastic force by taking the gradient of the elastic energy function E_h evaluated at that configuration:

$$\mathbf{F}^{n+\frac{1}{2}}(q,r,s,) \, \Delta q \, \Delta r \, \Delta s = -\frac{\partial}{\partial \mathbf{X}(q,r,s)} E_h \left(\cdots \mathbf{X}^{n+\frac{1}{2}}(q',r',s') \cdots \right). \tag{7.5}$$

Now spread the Lagrangian force and mass densities onto the Eulerian grid:

$$\mathbf{f}^{n+\frac{1}{2}}(\mathbf{x}) = \sum_{(q,r,s) \in G_h} \mathbf{F}^{n+\frac{1}{2}}(q,r,s) \, \delta_h \left(\mathbf{x} - \mathbf{X}^{n+\frac{1}{2}}(q,r,s) \right) \Delta q \, \Delta r \, \Delta s, \tag{7.6}$$

$$\rho^{n+\frac{1}{2}}(\mathbf{x}) = \sum_{(q,r,s) \in G_h} M(q,r,s) \, \delta_h \left(\mathbf{x} - \mathbf{X}^{n+\frac{1}{2}}(q,r,s) \right) \Delta q \, \Delta r \, \Delta s. \tag{7.7}$$

With these Eulerian quantities defined, we are ready to integrate the Navier–Stokes equations on the Eulerian grid g_h from time level n to time level $n+\frac{1}{2}$. This is done by the following implicit scheme, which is forward Euler in the nonlinear terms, but backward Euler otherwise:

$$\rho^{n+\frac{1}{2}}\left(\frac{\mathbf{u}^{n+\frac{1}{2}} - \mathbf{u}^n}{\Delta t/2} + S_h(\mathbf{u}^n)\mathbf{u}^n\right) + \mathbf{D}_h^0 \tilde{p}^{n+\frac{1}{2}} = \mu L_h \mathbf{u}^{n+\frac{1}{2}} + \mathbf{f}^{n+\frac{1}{2}}, \qquad (7.8)$$

$$\mathbf{D}_h^0 \cdot \mathbf{u}^{n+\frac{1}{2}} = 0. \qquad (7.9)$$

Note that (7.8)–(7.9) form a system of equations in the unknowns $\mathbf{u}^{n+\frac{1}{2}}(\mathbf{x})$ and $\tilde{p}^{n+\frac{1}{2}}(\mathbf{x})$, for $\mathbf{x} \in g_h$. (The only reason for the 'tilde' over the p is to distinguish it from another $p^{n+\frac{1}{2}}$ that will appear later.) The method(s) used to solve this system will be discussed below, after we have finished the statement of the time-stepping scheme.

The preliminary substep to the half-time level is now complete. The final substep uses the results of the preliminary substep and proceeds as follows. Note how everything in it is centred in time, thus achieving second-order accuracy.

First, we move the Lagrangian configuration from $\mathbf{X}^n$ to $\mathbf{X}^{n+1}$ by interpolating the velocity $\mathbf{u}^{n+\frac{1}{2}}$ to the configuration $\mathbf{X}^{n+\frac{1}{2}}$:

$$\mathbf{X}^{n+1}(q, r, s) = \mathbf{X}^n(q, r, s) + \Delta t \sum_{\mathbf{x} \in g_h} \mathbf{u}^{n+\frac{1}{2}}(\mathbf{x})\, \delta_h\big(\mathbf{x} - \mathbf{X}^{n+\frac{1}{2}}(q, r, s)\big)\, h^3.$$
$$(7.10)$$

Now we can proceed directly to the Navier–Stokes equations, since we are going to use the same $\rho^{n+\frac{1}{2}}$ and $\mathbf{f}^{n+\frac{1}{2}}$ as we did in the preliminary substep. Our task here is to integrate the Navier–Stokes equations from time level n to time level $n + 1$, and we do so by a scheme that is very similar to (7.8)–(7.9):

$$\rho^{n+\frac{1}{2}}\left(\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} + S_h(\mathbf{u}^{n+\frac{1}{2}})\mathbf{u}^{n+\frac{1}{2}}\right) + \mathbf{D}_h^0 p^{n+\frac{1}{2}} = \mu L_h \frac{\mathbf{u}^n + \mathbf{u}^{n+1}}{2} + \mathbf{f}^{n+\frac{1}{2}},$$
$$(7.11)$$

$$\mathbf{D}_h^0 \cdot \mathbf{u}^{n+1} = 0. \qquad (7.12)$$

Again, this is a system of equations in the unknowns $\mathbf{u}^{n+1}(\mathbf{x})$ and $p^{n+\frac{1}{2}}(\mathbf{x})$, for $\mathbf{x} \in g_h$. The method(s) of solution will be discussed below. Note that the nonlinear terms are here evaluated at the half-time level, thus using the result of the preliminary substep. In the viscous terms, however, we do not do this but instead make the scheme more implicit (and presumably more stable) by using the trapezoidal rule. The pressures $\tilde{p}^{n+\frac{1}{2}}$ and $p^{n+\frac{1}{2}}$ are just two different approximations to the pressure at the half-time level.

Although it might appear in the foregoing that the (7.12) is not centred in time between time levels n and $n+1$, this is illusory. The equation holds for all n and so we have $\mathbf{D}_h^0 \cdot \mathbf{u}^n = 0$ as well as $\mathbf{D}_h^0 \cdot \mathbf{u}^{n+1} = 0$. Similarly, one might wonder looking at the above scheme whether $\mathbf{u}^{n+1}$ is actually used for anything or just computed to be thrown away, since it is not used in the computation of the final configuration $\mathbf{X}^{n+1}$. The answer is that it is very much used, in both the preliminary and final substeps of the *next* time-step, where it plays the role of $\mathbf{u}^n$.

To complete the discussion of the time-stepping scheme, we have to say how to solve the two very similar systems of equations (7.8)–(7.9) and (7.11)–(7.12). These are linear systems of equations, since the nonlinear terms are known and can be evaluated before solving the system in each case. Similarly, the coefficient $\rho^{n+\frac{1}{2}}(\mathbf{x})$ is known (see (7.7)), so the terms in which ρ appears do not involve products of unknowns. The method of choice for solving these Stokes systems depends, however, on whether the mass density ρ is constant or not.

In many applications of the IB method, the immersed elastic material is neutrally buoyant in the ambient fluid, so ρ is a constant, independent of space and time, and (7.7) is not used. In this case, the linear systems of equations that we have to solve on the Eulerian grid g_h are difference equations with constant coefficients. As such, they are easily solved by the discrete Fourier transform, implemented by the Fast Fourier Transform algorithm. (To facilitate this, we typically formulate our problem on a periodic domain, *i.e.*, a cube with opposite faces identified.) For details, see Peskin and McQueen (1996).

When ρ is not constant, these linear systems of equations contain a non-constant (albeit known) coefficient, namely $\rho^{n+\frac{1}{2}}(\mathbf{x})$, and the Fourier transform is no longer useful, since different modes get coupled together. In this case, some iterative method has to be used. For IB computations of this type, see Fogelson and J. Zhu (1996), L. Zhu (2001) and L. Zhu and Peskin (2002), in which the multigrid method is used, but actually to solve the equations of a somewhat different time-stepping scheme. The specific time-stepping scheme stated here has not yet been implemented in the case of nonconstant mass density.

8. Research directions

The purpose of this section is to outline some of the ways in which the IB method (as stated above) might be improved. For most of the research directions suggested here, some work has already been done, as will be pointed out with references.

The time-stepping scheme presented in the previous section is mainly explicit, although it does contain a partly implicit Navier–Stokes solver. In

particular, the time-stepping scheme is explicit in the computation of the elastic force. In most applications, this results in a severe restriction on the time-step duration, and it would be a huge improvement if this could be overcome. For research on implicit and semi-implicit versions of the IB or related methods, see Peskin (1992), Tu and Peskin (1992), Mayo and Peskin (1993), Fauci and Fogelson (1993) and LeVeque and Li (1997).

The IB method described in this paper employs a uniform Eulerian grid. It would obviously be an advantage to be able to concentrate the computational effort where it is most needed. This is especially true for problems involving high Reynolds number flow, where high resolution is needed primarily in thin boundary layers or in other regions of limited extent where vorticity is concentrated. Several approaches to this problem have been and are being tried. The IB method has been combined with vortex methods (McCracken and Peskin 1980, Peskin 1981), with impulse methods (Cortez 1996, Cortez and Varela 1997, Cortez 2000), and with adaptive mesh refinement (Roma *et al.* 1999).

The IB method described in this paper is second-order accurate in space and time for problems with smooth solutions, such as the vibrations of an incompressible elastic material filling all of space, but problems with sharp interfaces do not have smooth solutions, and for these the IB method seems to be limited to first-order spatial accuracy. Another way to say this is that the IB method smears out sharp interfaces to a thickness which is of the order of the meshwidth. As mentioned above, the 'immersed interface method' (LeVeque and Li 1994, LeVeque and Li 1997, Li and Lai 2001) avoids this smearing and maintains second-order accuracy by modifying the difference equations near the interface. Another route to high accuracy is the 'blob-projection method' (Cortez and Minion 2000). It remains to be seen whether there is some simple way to modify the IB method to achieve this same goal.

One of the ways that the numerical error of the IB method shows up is a lack of volume conservation. Even though the velocity field on the Eulerian computational lattice is discretely divergence-free, this does not guarantee that the interpolated velocity field, in which the Lagrangian markers move, is continuously divergence-free. This error, of course, goes to zero as $h \to 0$, but it would be nice to reduce it at any particular h. Peskin and Printz (1993) (see also Rosar and Peskin (2001)) have shown that this can be done by using divergence and gradient operators that are 'tuned' to the interpolation scheme. Perhaps these principles can be applied to the Navier–Stokes solver as a whole, not just to the discrete divergence and gradient operators.

Although implementation issues have not been discussed in this paper, it is clearly important to parallelize the IB method. Shared-memory parallelization has been done (McQueen and Peskin 1997), and a distributed implementation in Titanium (Yelick *et al.* 1998) is under development (Yau 2002).

With regard to the mathematical formulation, it is not clear how to handle variable viscosity. The issue is not merely how to write the Navier–Stokes equations with a variable viscosity $\mu(\mathbf{x}, t)$, which is well known, but rather how to determine $\mu(\mathbf{x}, t)$ from the Lagrangian configuration of the material $\mathbf{X}(q, r, s, t)$. Also, in the case of an anisotropic elastic material, the viscoelasticity may be anisotropic, too. How should the formulation given here be modified to include such effects?

Can a turbulence model be effectively combined with the IB method (Schmid 200x)? This would make it possible to study the interaction of turbulent flows with immersed elastic bodies or boundaries.

Finally, there is not yet any convergence proof for the IB method (but see the stability analysis of Stockie and Wetton (1995, 1999)). Perhaps the reader will be inspired to discover such a proof.

9. Applications

The IB method was created in order to study the fluid dynamics of heart valves, and it has been applied to both natural (Peskin 1972, McQueen, Peskin and Yellin 1982, Peskin 1982, Meisner *et al.* 1985) and prosthetic (McQueen and Peskin 1983, 1985, 1991) cardiac valves. An outgrowth of this work has been a three-dimensional computer model of the whole heart (Peskin and McQueen 1996, McQueen and Peskin 1997, 2000, 2001, Kovacs, McQueen and Peskin 2001, McQueen, Peskin and Zhu 2001). Some other applications related to cardiovascular physiology include platelet aggregation during blood clotting (Fogelson 1984, Fauci and Fogelson 1993, Fogelson 1992, Wang and Fogelson 1999, Haoyu 2000), the deformation of red blood cells in a shear flow (Eggleton and Popel 1998), flow in elastic blood vessels (Vesier and Yoganathan 1992), flow in collapsible thin-walled vessels (Rosar and Peskin 2001), and flow in arterioles subject to the influence of a vasodilator transported by the flow itself (Arthurs, Moore, Peskin, Pitman and Layton 1998).

The swimming of eels, sperm, and bacteria have been studied by the IB method (Fauci and Peskin 1988, Fauci and McDonald 1994, Dillon, Fauci and Gaver 1995), as have the crawling locomotion of amoeba (Bottino 1998, Bottino and Fauci 1998), the waving motions of cilia beating in fluid (Dillon and Fauci 2000a), and the rotary motions and interactions of bacterial flagella immersed in fluid (Lim and Peskin 200x). Computer simulations of biofilms have been done in which the swimming of multiple microorganisms is handled by the IB method (Dillon, Fauci, Fogelson and Gaver 1996, Dillon and Fauci 2000b). Subtle fluid-dynamic interactions between nearby swimmers have been elucidated (Fauci 1990, Fauci and McDonald 1994). Embryological motions, too, have been studied by the IB method (Dillon and Othmer 1999).

Flows with suspended particles have been simulated by the IB method (Fogelson and Peskin 1988). Some noteworthy features of the IB method as applied to this problem are that the computational effort grows only linearly with the number of suspended particles, that the particles can be flexible and of arbitrary shape (*e.g.*, wood pulp fibres (Stockie and Green 1998)), and that the methodology is not restricted to Stokes flow but can handle nonzero Reynolds number. A variant (Sulsky and Brackbill 1991) of the IB method has been developed specifically for this problem. It has many interesting features including the direct transfer of stress (rather than force) from the immersed elastic body to the fluid, and it facilitates the use of conventional elasticity theory in describing the material properties of the suspended particles.

There are creatures that eat suspended (food) particles but they have to catch them first. The fluid dynamics of this has been studied by the IB method (Grunbaum, Eyre and Fogelson 1998).

The basilar membrane of the inner ear is an elastic shell immersed in fluid that vibrates in response to the incident sound. Its mechanical properties have the effect of sorting the sound energy into its different frequency components. This process has been simulated by the IB method, first in a simplified model (Beyer 1992), but more recently in the context of a three-dimensional model that includes the spiral geometry of the cochlea (Givelberg 1997).

Flow past a cylinder has been studied by the IB method (Lai and Peskin 2000). This traditional (fixed-boundary) fluid dynamics problem is mentioned for two reasons. The first reason is to emphasize that the IB method can do such problems: the technique is simply to put Lagrangian markers on the boundary and to tie each of these markers by a stiff spring to its target position in space. Small deviations of the markers from their target positions then generate the force needed to keep the boundary (approximately) fixed in place. (This idea was actually used before the introduction of the IB method – see Sirovich (1967) – and has recently been further developed (Goldstein, Handler and Sirovich 1993).) Note that this technique is as easily applied to a boundary of arbitrarily complicated geometry as it is to a cylinder. Also, there is no added difficulty if the boundary moves in a prescribed manner – just move the target points.

The second reason for mentioning flow past a cylinder is that it is a benchmark problem that provides validation for the IB method. The formation of the von Karmen vortex street in the wake of the cylinder is characterized by a Strouhal number (dimensionless frequency of vortex shedding) which varies with the Reynolds number (dimensionless reciprocal viscosity) in a manner that is well characterized experimentally. The IB method results reported in Lai and Peskin (2000) are in close agreement with experiment

over a range of Reynolds numbers (100–200) in which the Strouhal number varies significantly as the Reynolds number is changed.

Valveless pumping has been simulated by the IB method (Jung and Peskin 2001). This is a peculiar phenomenon in which a loop of fluid-filled tubing comprised of a flexible part and a stiff part is driven by periodic compression near one end of the flexible part. Despite the absence of valves, a robust mean flow (superimposed on an oscillating flow at the driving frequency) can be driven around the tube in this manner. Although this qualitative phenomenon was known before the IB simulations were done, those simulations led to an unexpected prediction: that the direction of the mean flow would be dependent on the frequency (and for some frequencies also on the amplitude) of the imposed periodic compression. This prediction has since been experimentally confirmed (Zhang 200x).

A laboratory experiment (Zhang, Childress, Libchaber and Shelley 2000) involving a flexible filament that flaps in a flowing soap film (like a flag in the wind) has been simulated by the IB method (Zhu 2001, Zhu and Peskin 2002, McQueen, Peskin and Zhu 2001). From a methodological standpoint, this computation is particularly significant because it takes into account the mass of the immersed elastic boundary. Indeed, an important discovery, made first in the simulation and only later confirmed experimentally, is that filament mass is essential for sustained flapping. The computation reproduces the observed frequency of flapping (about 50 Hz), and also the observed bistability of the filament: For the same parameters, different initial conditions can lead either to a flapping state or to a straight, stationary state of the filament.

Several aerodynamic applications of the IB method are well underway. They include sails, parachutes, flags, and insect flight. Airborne seeds (like the maple seed that spins as it falls) are natural candidates for IB simulation, but this has not yet been tried.

10. Conclusions

The immersed boundary (IB) method is for problems in which immersed incompressible viscoelastic bodies or immersed elastic boundaries interact with viscous incompressible fluid. The immersed viscoelastic bodies do not need to be neutrally buoyant in the ambient fluid, and the immersed elastic boundaries do not have to be massless. Indeed, the fluid itself does not need to have a uniform mass density.

The philosophy of the IB method is to blur the distinction between fluid dynamics and elasticity. Both the Eulerian variables that are conventionally used in fluid dynamics and also the Lagrangian variables that are conventionally used in elasticity theory are employed. Upon discretization, these two kinds of variables are defined on a fixed Cartesian grid and a moving

curvilinear grid, respectively. These grids are linked by a smoothed approximation to the Dirac delta function.

The simplicity of the Cartesian grid for the Eulerian variables facilitates the numerical integration of the Navier–Stokes equations, and the generality of the curvilinear grid for the Lagrangian variables makes it easy to model anisotropic (*e.g.*, fibre-reinforced) elastic material. It is an important feature of the IB method that these two grids are not constrained to relate to each other in any way at all, except that the moving curvilinear grid must be sufficiently dense with respect to the fixed Cartesian grid. A broad range of applications, mostly but not exclusively in biofluid dynamics, attest to the versatility of the immersed boundary method.

Acknowledgement

REFERENCES

K. M. Arthurs, L. C. Moore, C. S. Peskin, E. B. Pitman and H. E. Layton (1998), Modeling arteriolar flow and mass transport using the immersed boundary method. *J. Comput. Phys.* **147** 402–440.

R. P. Beyer (1992), A computational model of the cochlea using the immersed boundary method. *J. Comput. Phys.* **98** 145–162.

D. C. Bottino (1998), Modeling viscoelastic networks and cell deformation in the context of the immersed boundary method. *J. Comput. Phys.* **147** 86–113.

D. Bottino and L. J. Fauci (1998), A computational model of ameboid deformation and locomotion. *Eur. Biophys. J.* **27** 532–539.

A. J. Chorin (1968) Numerical solution of the Navier–Stokes equations. *Math. Comput.* **22** 745–762.

A. J. Chorin (1969) On the convergence of discrete approximations to the Navier–Stokes equations. *Math. Comput.* **23** 341–353.

R. Cortez (1996), An impulse-based approximation of fluid motion due to boundary forces. *J. Comput. Phys.* **123** 341–353.

R. Cortez (2000), A vortex/impulse method for immersed boundary motion in high Reynolds number flows. *J. Comput. Phys.* **160** 385–400.

R. Cortez and M. Minion (2000), The blob projection method for immersed boundary problems. *J. Comput. Phys.* **161** 428–453.

R. Cortez and D. A. Varela (1997), The dynamics of an elastic membrane using the impulse method. *J. Comput. Phys.* **138** 224–247.

R. Dillon and L. J. Fauci (2000*a*), An integrative model of internal axoneme mechanics and external fluid dynamics in ciliary beating. *J. Theor. Biol.* **207** 415–430.

R. Dillon and L. J. Fauci (2000*b*), A microscale model of bacterial and biofilm dynamics in porous media. *Biotechnol. Bioeng.* **68** 536–547.

R. Dillon and H. Othmer (1999), A mathematical model for the outgrowth and spatial patterning of the vertebrate limb bud. *J. Theor. Biol.* **197** 295–330.

R. Dillon, L. J. Fauci and D. Gaver (1995), A microscale model of bacterial swimming, chemotaxis and substrate transport. *J. Theor. Biol.* **177** 325–340.

R. Dillon, L. J. Fauci, A. L. Fogelson and D. Gaver (1996), Modeling biofilm processes using the immersed boundary method. *J. Comput. Phys.* **129** 57–73.

D. G. Ebin and R. A. Saxton (1986), The initial-value problem for elastodynamics of incompressible bodies. *Arch. Rat. Mech. Anal.* **94** 15–38.

D. G. Ebin and R. A. Saxton (1987), The equations of incompressible elasticity. In *Nonstrictly Hyperbolic Conservation Laws: Proc. AMS Special Session*, Vol. 60 of *Contemporary Mathematics*, AMS, pp. 25–34.

C. D. Eggleton and A. S. Popel (1998), Large deformation of red blood cell ghosts in a simple shear flow. *Phys. Fluids* **10** 1834–1845.

L. J. Fauci (1990), Interaction of oscillating filaments: A computational study. *J. Comput. Phys.* **86** 294–313.

L. J. Fauci and A. L. Fogelson (1993), Truncated Newton methods and the modeling of complex immersed elastic structures. *Comm. Pur. Appl. Math.* **46** 787–818.

L. J. Fauci and A. McDonald (1994), Sperm motility in the presence of boundaries. *B. Math. Biol.* **57** 679–699.

L. Fauci and C. S. Peskin (1988), A computational model of aquatic animal locomotion. *J. Comput. Phys.* **77** 85–108.

A. L. Fogelson (1984), A mathematical model and numerical method for studying platelet adhesion and aggregation during blood clotting. *J. Comput. Phys.* **56** 111–134.

A. L. Fogelson (1992) Continuum models of platelet aggregation: Formulation and mechanical properties. *SIAM J. Applied Math.* **52** 1089–1110.

A. L. Fogelson and C. S. Peskin (1988), A fast numerical method for solving the

three-dimensional Stokes' equations in the presence of suspended particles. *J. Comput. Phys.* **79** 50–69.

A. L. Fogelson and J. Zhu (1996), Implementation of a variable-density immersed boundary method, unpublished. Available from:
`http:/www.math.utah.edu/~fogelson`

E. Givelberg (1997), Modeling elastic shells immersed in fluid. PhD Thesis, Mathematics, New York University. Available from:
`http://www.umi.com/hp/Products/DisExpress.html`, order no. 9808292.

D. Goldstein, R. Handler and L. Sirovich (1993) Modeling a no-slip flow boundary with an external force-field. *J. Comput. Phys.* **105** 354–366.

D. Grunbaum, D. Eyre, and A. Fogelson (1998), Functional geometry of ciliated tentacular arrays in active suspension feeders. *J. Exp. Biol.* **201** 2575–2589.

Y. Haoyu (2000) Three dimensional computational modeling and simulation of platelet aggregation on parallel computers. PhD thesis, University of Utah.

F. H. Harlow and E. Welch (1965), Numerical calculation of time-dependent viscous incompressible flow of fluids with free surface. *Phys. Fluids* **8** 2182–2189.

E. Jung and C. S. Peskin (2001), Two-dimensional simulations of valveless pumping using the immersed boundary method. *SIAM J. Sci. Comput.* **23** 19–45.

S. J. Kovacs, D. M. McQueen and C. S. Peskin (2001), Modeling cardiac fluid dynamics and diastolic function. *Phil. Trans. R. Soc. Lond. A* **359** 1299–1314.

M.-C. Lai and Z. L. Li (2001), A remark on jump conditions for the three-dimensional Navier–Stokes equations involving an immersed moving membrane. *Applied Math. Letters* **14** 149–154.

M.-C. Lai and C. S. Peskin (2000), An immersed boundary method with formal second order accuracy and reduced numerical viscosity. *J. Comput. Phys.* **160** 705–719.

R. J. LeVeque and Z. L. Li (1994), The immersed interface method for elliptic equations with discontinuous coefficients and singular sources. *SIAM J. Numer. Anal.* **31** 1019–1044.

R. J. LeVeque and Z. Li (1997), Immersed interface methods for Stokes flow with elastic boundaries or surface tension. *SIAM J. Sci. Comput.* **18** 709–735.

Z. L. Li and M.-C. Lai (2001), Immersed interface method for the Navier–Stokes equations with singular forces. *J. Comput. Phys.* **171** 822–842.

S. K. Lim and C. S. Peskin (200x), Unpublished.

M. F. McCracken and C. S. Peskin (1980), A vortex method for blood flow through heart valves. *J. Comput. Phys.* **35** 183–205.

D. M. McQueen and C. S. Peskin (1983), Computer-assisted design of pivoting-disc prosthetic mitral valves. *J. Thorac. Cardiovasc. Surg.* **86** 126–135.

D. M. McQueen and C. S. Peskin (1985), Computer-assisted design of butterfly bileaflet valves for the mitral position. *Scand. J. Thorac. Cardiovasc. Surg.* **19** 139–148.

D. M. McQueen and C. S. Peskin (1991), Curved Butterfly Bileaflet Prosthetic Cardiac Valve. US Patent Number 5,026,391.

D. M. McQueen and C. S. Peskin (1997), Shared-memory parallel vector implementation of the immersed boundary method for the computation of blood flow in the beating mammalian heart. *J. Supercomputing* **11** 213–236.

D. M. McQueen and C. S. Peskin (2000), A three-dimensional computer model of the human heart for studying cardiac fluid dynamics. *Computer Graphics* **34** 56–60.

D. M. McQueen and C. S. Peskin (2001), Heart simulation by an immersed boundary method with formal second-order accuracy and reduced numerical viscosity. In *Mechanics for a New Millennium, Proceedings of the International Conference on Theoretical and Applied Mechanics* (ICTAM) *2000* (H. Aref and J. W. Phillips, eds.), Kluwer Academic Publishers.

D. M. McQueen, C. S Peskin, and E. L. Yellin (1982), Fluid dynamics of the mitral valve: Physiological aspects of a mathematical model. *Amer. J. Physiol.* **242** H1095–H1110.

D. M. McQueen, C. S. Peskin and L. Zhu (2001), The immersed boundary method for incompressible fluid–structure interaction. In *Proceedings of the First M.I.T. Conference on Computational Fluid and Solid Mechanics, June 12–15, 2001* (K. J. Bathe, ed.), Elsevier Science Ltd, Oxford, UK, Vol. 1, pp. 26–30.

A. A. Mayo and C. S. Peskin (1993), An implicit numerical method for fluid dynamics problems with immersed elastic boundaries. In *Fluid Dynamics in Biology: Proc. AMS–IMS–SIAM Joint Summer Research Conf. Biofluiddynamics*, Vol. 141 of *Contemporary Mathematics*, AMS, pp. 261–277.

J. S. Meisner, D. M. McQueen, Y. Ishida, H. O. Vetter, U. Bortolotti, J. A. Strom, R. W. M. Frater, C. S. Peskin, and E. L. Yellin (1985), Effects of timing of atrial systole on LV filling and mitral valve closure: Computer and dog studies. *Amer. J. Physiol.* **249** H604–H619.

C. S. Peskin (1972), Flow patterns around heart valves: A digital computer method for solving the equations of motion. PhD Thesis, Albert Einstein College of Medicine, July, 211pp. Available from: `http://www.umi.com/hp/Products/DisExpress.html`, order no. 7230378.

C. S. Peskin (1981), Vortex dynamics of the aortic sinus. In *Mathematical Aspects of Physiology* (Hoppensteadt FC, ed.), Vol. 19 of *Lectures in Applied Mathematics*, AMS, Providence, RI, pp. 93–104.

C. S. Peskin (1982), The fluid dynamics of heart valves: Experimental, theoretical, and computational methods. *Ann. Rev. Fluid Mech.* **14** 235–259.

C. S. Peskin (1992), Two examples of mathematics and computing in the biological sciences: Blood flow in the heart and molecular dynamics. In *Mathematics into the Twenty-first Century* (F. E. Browder, ed.), AMS, pp. 395–415.

C. S. Peskin and D. M. McQueen (1996), Fluid dynamics of the heart and its valves. In *Case Studies in Mathematical Modeling: Ecology, Physiology, and Cell Biology* (H. G. Othmer, F. R. Adler, M. A. Lewis and J. C. Dallon, eds.), Prentice-Hall, Englewood Cliffs, NJ, pp. 309–337.

C. S. Peskin and B. F. Printz (1993), Improved volume conservation in the computation of flows with immersed elastic boundaries. *J. Comput. Phys.* **105** 33–46.

A. M. Roma, C. S. Peskin and M. J. Berger (1999), An adaptive version of the immersed boundary method. *J. Comput. Phys.* **153** 509–534.

M. E. Rosar and C. S. Peskin (2001), Fluid flow in collapsible elastic tubes: A three-dimensional numerical model. *New York J. Math.* **7** 281–302.

P. Schmid (200x), Unpublished.

L. Sirovich (1967) Initial and boundary value problems in dissipative gas dynamics. *Phys. Fluids* **10** 24–34.

J. M. Stockie (1997), Analysis and computation of immersed boundaries, with application to pulp fibres. PhD thesis, Institute of Applied Mathematics, University of British Columbia, Vancouver, BC, Canada. Available from: `http://www.iam.ubc.ca/theses/stockie/stockie.html`

J. M. Stockie and S. I. Green (1998), Simulating the motion of flexible pulp fibres using the immersed boundary method. *J. Comput. Phys.* **147** 147–165.

J. M. Stockie and B. T. R. Wetton (1995) Stability analysis for the immersed fiber problem. *SIAM J. Applied Math.* **55** 1577–1591.

J. M. Stockie and B. T. R. Wetton (1999) Analysis of stiffness in the immersed boundary method and implications for time-stepping schemes. *J. Comput. Phys.* **154** 41–64.

D. Sulsky and J. U. Brackbill (1991), A numerical method for suspension flow. *J. Comput. Phys.* **96** 339–368.

C. Tu and C. S. Peskin (1992), Stability and instability in the computation of flows with moving immersed boundaries: A comparison of three methods. *SIAM J. Sci. Statist. Comput.* **13** 1361–1376.

C. C. Vesier and A. P. Yoganathan (1992), A computer method for simulation of cardiovascular flow fields: Validation of approach. *J. Comput. Phys.* **99** 271–287.

N. T. Wang and A. L. Fogelson (1999) Computational methods for continuum models of platelet aggregation. *J. Comput. Phys.* **151** 649–675.

S. M. Yau (2002), A generic immersed boundary method package in titanium. Master's Report, Computer Science Division, University of California, Berkeley.

K. A. Yelick, L. Semenzato, G. Pike, C. Miyamoto, B. Liblit, A. Krishnamurthy, P. N. Hilfinger, S. L. Graham, D. Gay, P. Colella and A. Aiken (1998), Titanium: A high-performance Java dialect. *Concurrency: Practice and Experience* **10**.

J. Zhang (200x), Unpublished.

J. Zhang, S. Childress, A. Libchaber and M. Shelley (2000), Flexible filaments in a flowing soap film as a model for one-dimensional flags in a two-dimensional wind. *Nature* **408** 835.

L. Zhu (2001), Simulation of a flapping filament in a flowing soap film by the immersed boundary method. PhD Thesis, New York University.

L. Zhu and C. S. Peskin (2002), Simulation of a flapping filament in a flowing soap film by the immersed boundary method. Submitted to *J. Comput. Phys.*