JOURNAL OF COMPUTATIONAL PHYSICS 10, 252–271 (1972)

Flow Patterns Around Heart Valves: A Numerical Method

CHARLES S. PESKIN*

Albert Einstein College of Medicine, Yeshiva University, Bronx, New York 10461

Received August 13, 1971

The subject of this paper is the flow of a viscous incompressible fluid in a region containing immersed boundaries which move with the fluid and exert forces on the fluid. An example of such a boundary is the flexible leaflet of a human heart valve. It is the main achievement of the present paper that a method for solving the Navier–Stokes equations on a rectangular domain can now be applied to a problem involving this type of immersed boundary. This is accomplished by replacing the boundary by a field of force which is defined on the mesh points of the rectangular domain and which is calculated from the configuration of the boundary. In order to link the representations of the boundary and fluid, since boundary points and mesh points need not coincide, a semi-discrete analog of the δ function is introduced. Because the boundary forces are of order h^{-1}, and because they are sensitive to small changes in boundary configuration, they tend to produce numerical instability. This difficulty is overcome by an implicit method for calculating the boundary forces, a method which takes into account the displacements that will be produced by the boundary forces themselves. The numerical scheme is applied to the two-dimensional simulation of flow around the natural mitral valve.

I. INTRODUCTION

The subject of this paper is the flow of a viscous incompressible fluid in a region containing immersed boundaries which interact with the fluid. We use the word "interact" to emphasize that the boundaries in question are moved by the fluid and exert forces on the fluid. An example of such a boundary, the example that motivated the present work, is the flexible leaflet of a human heart valve. The valve leaflet moves with the fluid, but its effect on the flow pattern is by no means small. During part of each heartbeat the valve leaflets stop the flow.

Methods are available for solving the Navier–Stokes equations on rectangular

* Medical Scientist Trainee. This work was performed in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Sue Golding Graduate Division of the Albert Einstein College of Medicine.

domains [1, 2]. It is the main achievement of the present paper that one such method can now be applied to a problem involving a moving immersed boundary. In brief this is accomplished by replacing the boundary by a field of force which is defined on the mesh points of the rectangular domain and which has approximately the same effect on the fluid as the immersed boundary would have had.

The advantages of treating the boundaries of interest as if they were immersed in a rectangular domain are multiple. First, the heart-valve leaflet is actually immersed in blood so that no other treatment is reasonable. Even for other boundaries, however, there are advantages. Because of this approach the discrete equations of motion of the fluid can be taken to be identical at all the mesh points of the rectangular domain, without regard to whether those points lie inside, outside, or near the edge of the region of interest. This fact is especially important because the region of interest is constantly changing. Finally there is an advantage that we have not yet exploited: the availability of very fast stable direct methods for solving Poisson's equation on rectangular domains [5].

The present work is closely related to that of J. A. Viecelli [3, 4], who has treated arbitrary external boundaries which may be fixed or have some prescribed motion. Along such boundaries Viecelli applies pressures which are just sufficient to prevent the fluid from crossing the given instantaneous position of the boundary. Also related are free surface calculations, in which the pressure along the boundary is known, but the motion of the boundary is unknown. In the present work neither the boundary motion nor the boundary forces are known in advance. Instead, the elastic properties of the boundary determine the forces from the boundary configuration at each instant as the motion evolves.

II. Statement of the Problem

Consider a two-dimensional flow contained in a periodic rectangular domain R. The fluid is viscous and incompressible. Immersed in the domain is a moving boundary B which exerts forces on the fluid. In the most general case these forces may be any function of the boundary configuration. For example, the boundary may be assumed to be elastic, in which case the forces will be calculated from Hooke's law. In all cases we assume that the boundary is massless, and that there is fluid of the same type on both sides of the boundary.

The fluid is described by the velocity field $\mathbf{u}(\mathbf{x}, t)$ and by the pressure $p(\mathbf{x}, t)$. The boundary configuration is described by the curve $\mathbf{x}(s, t)$, where s is a parameter chosen in such a way that a given value of s represents a given physical point of the boundary for all times t. The forces which the boundary exerts on the fluid will be described by the function $\mathbf{f}(s, t)$, where $\mathbf{f}(s, t)\, ds$ is the force exerted on the fluid by the portion of boundary which lies in the interval $(s, s + ds)$.

The equations of motion of the fluid are the Navier–Stokes equations which are here given in nondimensional form

$$\mathbf{u}_t + \mathbf{u} \cdot \nabla \mathbf{u} = -\nabla p + \nabla^2 \mathbf{u} + \mathbf{F}, \tag{2.1}$$

$$\nabla \cdot \mathbf{u} = 0, \tag{2.2}$$

where the vector $\mathbf{F}$ gives the external force per unit area (per unit volume for a three-dimensional flow) applied to the fluid. In this nondimensionalization, the largest component of $\mathbf{u}$ that occurs in the fluid may be called the Reynolds number.

Since the fluid is viscous the equation of motion of the boundary is simply

$$\mathbf{x}_t(s) = \mathbf{u}(\mathbf{x}(s)). \tag{2.3}$$

We now make the observation that the interval $(s, s + ds)$ of B is exerting the force $\mathbf{f}(s)\, ds$ on every area in R however small which contains the interval $(s, s + ds)$ of B. The resulting element of external force per unit area applied to the fluid is, therefore,

$$d\mathbf{F}(\mathbf{x}) = \mathbf{f}(s)\, ds\, \delta(\mathbf{x} - \mathbf{x}(s)), \tag{2.4}$$

where the δ indicates a two-dimensional impulse function. Summing all such intervals of B we find that the external force density due to the boundary is given by

$$\mathbf{F}(\mathbf{x}) = \int_{\mathbf{x}(s) \in B} \mathbf{f}(s)\, ds\, \delta(\mathbf{x} - \mathbf{x}(s)). \tag{2.5}$$

To check that this equation makes sense, integrate it over all $\mathbf{x} \in R$ to find

$$\int_{\mathbf{x} \in R} \mathbf{F}(\mathbf{x})\, da = \int_{\mathbf{x}(s) \in B} \mathbf{f}(s)\, ds. \tag{2.6}$$

The field $\mathbf{F}$ as defined by Eq. (2.5) is singular, being zero everywhere except on B and yet having a finite integral over R.

Since the forces $\mathbf{f}(s)$ are determined by the boundary configuration $\mathbf{x}(s)$ we may write

$$\mathbf{f} = M(\mathbf{x}), \tag{2.7}$$

where M is a nonlinear operator describing the elastic properties of the boundary. Thus our five unknowns $\mathbf{u}$, p, $\mathbf{F}$, $\mathbf{x}$, and $\mathbf{f}$ are connected by the five equations (2.1), (2.2), (2.3), (2.5), (2.7). Together these constitute a statement of the problem.

We now put these equations into a form which will be more useful for numerical work. First we make the trivial observation that Eq. (2.3) can be rewritten as follows:

$$\mathbf{x}_t(s) = \mathbf{u}(\mathbf{x}(s)) = \int_{\mathbf{x} \in R} \mathbf{u}(\mathbf{x})\, \delta(\mathbf{x} - \mathbf{x}(s))\, da. \tag{2.8}$$

Comparison of Eqs. (2.5) and (2.8) shows that the δ function makes its appearance under an integral sign when fluid quantities are mapped into boundary quantities and vice versa. That is, of the five equations that describe the problem, three contain either boundary quantities alone [Eq. (2.7)] or fluid quantities alone [Eqs. (2.1) and (2.2)]. The remaining two equations form the link between boundary and fluid, and each of these involves the function $\delta(\mathbf{x} - \mathbf{x}(s))$. This fact will be used in construction of the numerical scheme.

Second, we eliminate the constraint equation $\nabla \cdot \mathbf{u} = 0$ and simultaneously eliminate the variable p. This is accomplished by application of an operator P which projects an arbitrary vector field onto the space of divergence-free vector fields. On the periodic rectangular domain R, the operator P is defined as follows:

$$\mathbf{w}^D = P\mathbf{w} \tag{2.9}$$

if and only if

$$\nabla \cdot \mathbf{w}^D = 0 \qquad \text{and} \qquad \mathbf{w} = \mathbf{w}^D + \nabla\varphi, \tag{2.10}$$

where φ is a scalar. The domain is assumed to be periodic so that $\mathbf{w}$, $\mathbf{w}^D$, and φ all obey boundary conditions of the form $\mathbf{w}(0, y) = \mathbf{w}(L_1, y)$, $\mathbf{w}(x, 0) = \mathbf{w}(x, L_2)$, etc.

From the definition it is clear at once that

(1) P is linear.

(2) For any scalar φ, $P\nabla\varphi = 0$.

(3) For any vector $\mathbf{u}$ such that $\nabla \cdot \mathbf{u} = 0$, $P\mathbf{u} = \mathbf{u}$.

Using these properties and applying P to both sides of (2.1) we find that

$$\mathbf{u}_t = P(-\mathbf{u} \cdot \nabla\mathbf{u} + \nabla^2\mathbf{u}) + P\mathbf{F}. \tag{2.11}$$

Equation (2.11) replaces both (2.1) and (2.2) because the form of (2.11) guarantees that if $\nabla \cdot \mathbf{u} = 0$ at $t = 0$, then $\nabla \cdot \mathbf{u} = 0$ at all times thereafter.

Note also that Eqs. (2.10) imply

$$\nabla \cdot \mathbf{w} = \nabla^2\varphi, \tag{2.12}$$

so that if $\mathbf{w}$ is known the calculation of φ, and hence of $P\mathbf{w}$, involves the solution of an equation of the Laplace type. For example to calculate $P\mathbf{F}$ it is necessary first to solve for φ according to

$$\begin{aligned} \nabla^2\varphi &= \nabla \cdot \mathbf{F} \\ &= \nabla \cdot \int_B \mathbf{f}(s)\, \delta(\mathbf{x} - \mathbf{x}(s))\, ds. \end{aligned} \tag{2.13}$$

With φ thus defined

$$P\mathbf{F} = \mathbf{F} - \nabla\varphi. \tag{2.14}$$

Note that $\mathbf{F}$, and hence the sources of φ, are zero everywhere except along B. But φ itself, and hence $P\mathbf{F}$, is in general nonzero throughout R. Thus the boundary forces influence the entire flow instantaneously. This is a consequence of the incompressibiliy of the fluid.

To conclude this section we summarize the equations of motion in the form which we shall use as a point of departure for the construction of a numerical scheme. They are

$$\mathbf{u}_t = P(-\mathbf{u}\cdot\nabla\mathbf{u} + \nabla^2\mathbf{u} + \mathbf{F}), \tag{2.15a}$$

$$\mathbf{x}_t(s) = \int_R \mathbf{u}(\mathbf{x})\,\delta(\mathbf{x} - \mathbf{x}(s))\,da, \tag{2.15b}$$

$$\mathbf{F}(\mathbf{x}) = \int_B \mathbf{f}(s)\,\delta(\mathbf{x} - \mathbf{x}(s))\,ds, \tag{2.15c}$$

$$\mathbf{f} = M(\mathbf{x}). \tag{2.15d}$$

III. Difference Equations

In this section we construct difference equations that correspond to Eqs. (2.15a)–(2.15d). As a preliminary we state that the rectangular domain R is covered by a square mesh of mesh width h, the points of which will be designated by (ij), and that we shall use the obvious notation $\mathbf{u}_{ij} = \mathbf{u}(ih, jh)$ and similar notation for other functions defined on R. The immersed boundary is represented by discrete points $\mathbf{x}_k = \mathbf{x}(h_B k)$, k integer, where h_B is the distance between neighboring points of the unstressed boundary. Other functions on the boundary are defined similarly.

The points $\mathbf{x}_k$ are not required to coincide with mesh points but may lie anywhere in R. In fact, it would be difficult to imagine a workable scheme for this problem in which such a requirement were imposed. In the first place, the numerical scheme is accurate only if $|\mathbf{u}|\,\delta t < h$ [see below Eq. (3.25)] everywhere. This implies that the displacement of the immersed boundary points is less than one mesh width on any given time step. Moreover, the boundary forces will be very sensitive to small displacements, so that the seemingly small error that would be introduced by localizing boundary points to mesh points would in fact produce large errors in the boundary forces. The foregoing can be summarized by saying that the fluid is described in Eulerian form while the boundary is described in Lagrangian form.

A. *Difference Scheme for the Navier–Stokes Equations*

In this section we shall consider the external force density $\mathbf{F}$ as though it were known in advance, and we shall discuss a numerical scheme due to A. J. Chorin [1, 2] which would be applicable in that case. We have incorporated this scheme into the present work essentially verbatim, except for the addition of a convenient method of handling a periodic boundary condition. Hence we discuss this scheme in outline form only, referring the reader to the references for details and proofs. We shall describe Chorin's method here in two steps. First the time steps will be made discrete, and then the space steps.

Let $\mathbf{u}^n = \mathbf{u}(n\,\delta t) = (u_1{}^n, u_2{}^n)$, and consider the system of equations

$$\left[I + \delta t \left(u_1{}^n \frac{\partial}{\partial x} - \frac{\partial^2}{\partial x^2}\right)\right] \mathbf{u}^* = \mathbf{u}^n + \delta t \mathbf{F}, \tag{3.1}$$

$$\left[I + \delta t \left(u_2{}^n \frac{\partial}{\partial y} - \frac{\partial^2}{\partial y^2}\right)\right] \mathbf{u}^{**} = \mathbf{u}^*, \tag{3.2}$$

$$\mathbf{u}^{n+1} = P\mathbf{u}^{**}. \tag{3.3}$$

Substituting (3.2) in (3.1) and ignoring terms of order $(\delta t)^2$ we find

$$\mathbf{u}^{**} + \delta t(\mathbf{u}^n \cdot \nabla \mathbf{u}^{**} - \nabla^2 \mathbf{u}^{**}) = \mathbf{u}^n + \delta t\, \mathbf{F}. \tag{3.4}$$

Substituting this into (3.3) yields

$$\mathbf{u}^{n+1} = \mathbf{u}^n + \delta t\, P(-\mathbf{u}^n \cdot \nabla \mathbf{u}^{**} + \nabla^2 \mathbf{u}^{**} + \mathbf{F}), \tag{3.5}$$

which is clearly consistent with (2.15a) since, as $\delta t \to 0$, $\mathbf{u}^{**} \to \mathbf{u}^n$. Each of the Eqs. (3.1) and (3.2) has derivatives in only one space direction. The corresponding difference equations are written out in Section III.D-(2) below. Each of the difference equations has the following form

$$-A_k X_{k-1} + B_k X_k - C_k X_{k+1} = D_k \qquad k = 1, 2, \ldots, N, \tag{3.6}$$

in which the subscript arithmetic is understood to be modulo N because of the periodic nature of the domain. We note two features of the matrix of this system of equations. First, it is well conditioned if $A > 0$, $C > 0$, $B > A + C$. This leads to the condition

$$hR < 2, \tag{3.7}$$

where R is the Reynolds number as defined in the discussion following Eq. (2.1). Second, the matrix is almost tridiagonal, the discrepancy being due to the periodic nature of the domain. This introduces nonzero terms in the off-diagonal corners

of the matrix. Let L_kX denote the left-hand side of (3.6). Then, if we revert to ordinary subscript arithmetic, introduce an extra point $k = N + 1$, and temporarily drop the equation at $k = 1$, we can define the following systems of equations which are strictly tridiagonal, and which can therefore be solved by the method given in Richtmyer and Morton [6]:

$$L_kX^{(0)} = D_k \qquad k = 2, 3,..., N \qquad X_1^{(0)} = X_{N+1}^{(0)} = 0, \tag{3.8}$$

$$L_kX^{(1)} = 0 \qquad k = 2, 3,..., N \qquad X_1^{(1)} = X_{N+1}^{(1)} = 1. \tag{3.9}$$

The linear combination $X = X^{(0)} + \lambda X^{(1)}$ is then a solution of (3.6) at $k = 2, 3,..., N$. To complete the solution it is only necessary to substitute this linear combination into (3.6) with $k = 1$ and solve for λ. The foregoing is essentially rank -1 modification [7].

We now discuss the spatial difference equations corresponding to Eq. (3.3). Referring to Eqs. (2.9) and (2.10) we define the discrete projection operator P as follows:

$$\mathbf{w}^D = P\mathbf{w} \tag{3.10}$$

if and only if

$$D\mathbf{w}^D = 0, \qquad \mathbf{w} = \mathbf{w}^D + G\varphi, \tag{3.11}$$

where D and G are the discrete operators defined by

$$(D\mathbf{w})_{ij} = (1/2h)[(w_1)_{i+1,j} - (w_1)_{i-1,j} + (w_2)_{i,j+1} - (w_2)_{i,j-1}], \tag{3.12}$$

$$(G\varphi)_{ij} = (1/2h)(\{\varphi_{i+1,j} - \varphi_{i-1,j}\}, \{\varphi_{i,j+1} - \varphi_{i,j-1}\}), \tag{3.13}$$

and where all functions are assumed to be periodic in (ij).

To actually compute $P\mathbf{w}$ from $\mathbf{w}$ it is necessary to solve

$$DG\varphi = D\mathbf{w} \tag{3.14}$$

or

$$(1/4h^2)[\varphi_{i+2,j} + \varphi_{i-2,j} + \varphi_{i,j+2} + \varphi_{i,j-2} - 4\varphi_{ij}] = (D\mathbf{w})_{ij}\,. \tag{3.15}$$

This can be done by successive overrelaxation [8]. We emphasize here that once $(D\mathbf{w})_{ij}$ has been found, Eqs. (3.15) actually represent four separate systems of equations over the four distinct subdomains ("chains") listed here: (i even, j even), (i even, j odd), (i odd, j even), (i odd, j odd). In order that the solution remain smooth, it is necessary that these chains be treated on an equal footing insofar as the immersed boundary is concerned. This point will be discussed below. Once φ has been found, we have, of course,

$$P\mathbf{w} = \mathbf{w} - G\varphi. \tag{3.16}$$

This completes the description of the difference equations that apply to the fluid.

B. *Semidiscrete Representation of the* δ *Function*

We now come to a crucial aspect of the present work, the means by which the discrete boundary and fluid representations are linked. We recall that Eqs. (2.15b) and (2.15c) both involve the function $\delta(\mathbf{x} - \mathbf{x}(s))$, where $\mathbf{x} \in R$ and $\mathbf{x}(s) \in B$. Now in the discrete representation the only representatives of points $\mathbf{x} \in R$ are the mesh points (ij), and the representatives $\mathbf{x}_k$ of the points of B need not coincide with these mesh points. We therefore have need of a function $D_{ij}(\mathbf{x}_k)$, $\mathbf{x}_k \in B$, which will replace the δ function in the discrete equations (which we have yet to write down) that will replace Eqs. (2.15b) and (2.15c). Thus the function $D_{ij}(\mathbf{x}_k)$ will link the boundary and the fluid in two ways:

(1) by evaluating the velocity field at a given point of the immersed boundary;

(2) by spreading the boundary forces onto the nearby mesh points of the fluid domain.

We impose certain requirements on the function $D_{ij}(\mathbf{x}_k)$:

(1) Recall that the defining property of the (two-dimensional) δ function is

$$\int_R \varphi(\mathbf{x})\, \delta(\mathbf{x} - \mathbf{x}_k)\, da = \varphi(\mathbf{x}_k) \tag{3.17}$$

for all functions φ in some suitable test space. Suppose φ is continuous and let $\varphi_{ij} = \varphi(ih, jh)$. Then we require of our function $D_{ij}(\mathbf{x}_k)$ that

$$\lim_{h \to 0} \sum_{ij \in R} h^2 D_{ij}(\mathbf{x}_k)\, \varphi_{ij} = \varphi(\mathbf{x}_k) \tag{3.18}$$

for all continuous functions φ.

(2) As the boundary moves, it is desirable that the connection between it and the fluid domain change in a continuous manner. Therefore we require that for each (ij), $D_{ij}(\mathbf{x}_k)$ be a continuous function of $\mathbf{x}_k$.

(3) Recall the discussion above concerning the independence of the subdomains (i even, j even), etc., with regard to evaluating the discrete projection operator. Recall also that it is the projection operator that expresses the incompressibility of the fluid and that makes the boundary forces have an instantaneous effect throughout the fluid. It follows from these notions that in order to keep the solution smooth it is necessary to apply the force due to any portion of the boundary equally onto all of the four chains. Since the forces are applied to the fluid via the function $D_{ij}(\mathbf{x}_k)$, we impose the requirement that for all $\mathbf{x}_k$

$$\frac{1}{4} = \sum_{\substack{i \text{ even} \\ j \text{ even}}} h^2 D_{ij}(\mathbf{x}_k) = \sum_{\substack{i \text{ even} \\ j \text{ odd}}} h^2 D_{ij}(\mathbf{x}_k) = \sum_{\substack{i \text{ odd} \\ j \text{ even}}} h^2 D_{ij}(\mathbf{x}_k) = \sum_{\substack{i \text{ odd} \\ j \text{ odd}}} h^2 D_{ij}(\mathbf{x}_k). \tag{3.19}$$

A function which meets all of these requirements is

$$D_{ij}(\mathbf{x}) = D_{ij}(\alpha h, \beta h) = \begin{cases} \dfrac{1}{16h^2}(2 - |\alpha - i|)(2 - |\beta - j|) \\ \qquad |\alpha - i| \leqslant 2 \qquad |\beta - j| \leqslant 2 \\ 0 \quad \text{otherwise.} \end{cases} \tag{3.20}$$

Some examples are given in Fig. 1.

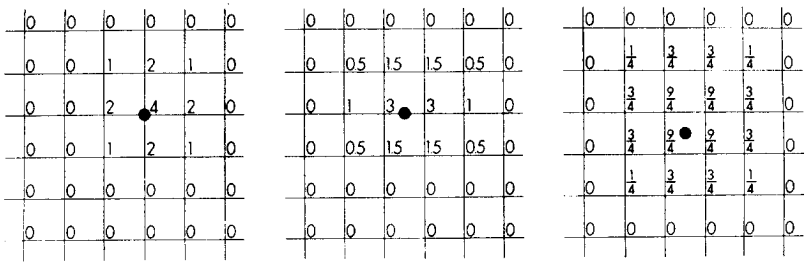

FIG. 1. Semidiscrete analog of the δ function: three typical cases (the factor $1/16h^2$ has been omitted). The coefficients shown provide the linkage between the boundary points and the fluid mesh.

With this function in hand we can write down at once the discrete equations corresponding to (2.15b) and (2.15c):

$$\mathbf{x}_k^{n+1} = \mathbf{x}_k{}^n + \delta t \sum_{ij} h^2 \mathbf{u}_{ij}^{n+1} D_{ij}(\mathbf{x}_k{}^n), \tag{3.21}$$

$$\mathbf{F}_{ij} = \sum_k h_B \mathbf{f}_k D_{ij}(\mathbf{x}_k). \tag{3.22}$$

Note that the field $\mathbf{F}_{ij}$ is of order h_B/h^2 in the neighborhood of B, but zero at points far (i.e., greater than two mesh widths) away from B. We may therefore say that $\mathbf{F}_{ij}$ is an "approximately singular" field that represents the boundary forces on the mesh.

Remark. In Eq. (3.21) the choice of $\mathbf{x}^n$ as the argument of D_{ij} is somewhat arbitrary. For example, the implicit formula

$$\mathbf{x}_k^{n+1} = \mathbf{x}_k{}^n + \delta t \sum_{ij} h^2 \mathbf{u}_{ij}^{n+1} D_{ij}(\mathbf{x}_k^{n+1}) \tag{3.23}$$

would be equally consistent with (2.15b). To study the relationship between (3.21) and (3.23) note that

$$A\mathbf{x}^* = \mathbf{x}^n + \delta t \sum_{ij} h^2 \mathbf{u}_{ij}^{n+1} D_{ij}(\mathbf{x}^*) \tag{3.24}$$

is a contraction mapping on R in the maximum norm if

$$u_{\max} \, \delta t < h/4, \tag{3.25}$$

where

$$u_{\max} = \max_{ij,p=1,2} | \mathbf{u}_{p,ij}^{n+1} |.$$

Therefore, if (3.25) holds, (3.23) has a unique solution which can be found by the straightforward iteration

$$\mathbf{x}^{n+1,m+1} = A\mathbf{x}^{n+1,m}. \tag{3.26}$$

The first step of this iteration is Eq. (3.21). These considerations suggest that a necessary condition for the accuracy of Eq. (3.21) is the inequality (3.25).

C. *Discrete Computation of the Boundary Forces*

In this section we study the computation of the boundary forces $\mathbf{f}_k$ from the boundary configuration $\mathbf{x}_k$. In the most general case this relationship is simply expressed by the arbitrary functions

$$\mathbf{f}_k = \mathbf{f}_k(\cdots \mathbf{x}_{k'} \cdots). \tag{3.27}$$

In practice, however, we have restricted consideration to functions which fall into a class which is somewhat less general, but still general enough to represent a great variety of physical boundaries. This class of functions will now be described in terms of the manner in which the functional form is stored in computer memory.

First we construct in computer memory a table of links, numbered according to any arbitrary scheme. Every link has associated with it two end points. At least one of these is a boundary point, and the other is again a boundary point or a fixed point in space. Also associated with each link is a resting length R_0 and a stiffness S. Links are of two types: cord like or rod like, and the nature of each link is stored in the link table. If R is the actual length of a link for some given boundary configuration, then the tension in that link is given by

$$T = S(R - R_0) \tag{3.28}$$

for $R \geqslant R_0$. When $R < R_0$ the rod-like links continue to obey (3.28) while the cord-like links have $T = 0$. Let $\hat{\mathbf{a}}_{kj}$ be a unit vector pointing away from the k-th boundary point parallel to the j-th link that emanates from the k-th boundary point. Then the resultant forces $h_B\mathbf{f}_k$ are computed from

$$h_B\mathbf{f}_k = \sum_j T_j \hat{\mathbf{a}}_{kj}. \tag{3.29}$$

The algorithm that evaluates these forces is rendered simple and efficient if it looks at each link only once, computes the tension in that link and applies this tension to the boundary points at each end of the link. The quantities in the link table are fixed for all time. For example, the subscript of each boundary point is recorded there, but not the coordinates of the boundary point.

The diversity of physical structures which can be specified by means of a link

(1) An infinitely flexible elastic membrane is specified by simply linking successive points of the membrane to each other by cord-like links.

(2) An almost rigid structure fixed in space is specified by connecting each of its boundary points to a corresponding fixed space point by a link of resting length zero and large stiffness.

(3) A moving structure can be made to retain its shape by specifying an appropriate network of rod-like links. For example, a circle will be kept circular by the introduction of rod-like spokes.

Moreover, the link formalism offered here has a natural generalization to the case of active boundaries like the muscular heart wall. To make a link act like a segment of muscle fibre it is only necessary to make the resting length variable, and to specify an "equation of motion" for the resting length. In that case the rate of change of resting length becomes what physiologists call velocity of the contractile element, which depends on fibre length, tension, and state of activation of the muscle.

D. *Summary of the Calculation of* $\mathbf{u}^{n+1}$ *and* $\mathbf{x}^{n+1}$

In the previous section it was shown how the boundary forces $\mathbf{f}_k$ are calculated from the boundary configuration. It is not obvious, however, whether the appropriate configuration to use is $\mathbf{x}_k{}^n$, $\mathbf{x}_k^{n+1}$, or some other configuration. This important question is the subject of Section IV. For the time being we simply assume that forces $\mathbf{f}_k$ have been determined, and we write down the algorithm by means of which $\mathbf{u}_{ij}^{n+1}$ and $\mathbf{x}_k^{n+1}$ are calculated. The steps are as follows:

(1) Evaluate

$$\mathbf{F}_{ij} = \sum_k h_B \mathbf{f}_k \, D_{ij}(\mathbf{x}_k{}^n).$$

(2) Successively solve for $\mathbf{u}^*$ and $\mathbf{u}^{**}$

$$-\left(\frac{\delta t}{2h} u_{1,ij}^n + \frac{\delta t}{h^2}\right)\mathbf{u}_{i-1,j}^* + \left(1 + 2\frac{\delta t}{h^2}\right)\mathbf{u}_{ij}^* - \left(-\frac{\delta t}{2h} u_{1,ij}^n + \frac{\delta t}{h^2}\right)\mathbf{u}_{i+1,j}^* = \mathbf{u}_{ij}^n + \mathbf{F}_{ij}\,\delta t,$$

$$-\left(\frac{\delta t}{2h} u_{2,ij}^n + \frac{\delta t}{h^2}\right)\mathbf{u}_{i,j-1}^{**} + \left(1 + 2\frac{\delta t}{h^2}\right)\mathbf{u}_{ij}^{**} - \left(-\frac{\delta t}{2h} u_{2,ij}^n + \frac{\delta t}{h^2}\right)\mathbf{u}_{i,j+1}^{**} = \mathbf{u}_{ij}^* .$$

(3) Evaluate $\mathbf{u}^{n+1} = P\mathbf{u}^{**}$ as follows:

(a) Evaluate

$$(D\mathbf{u}^{**})_{ij} = \frac{1}{2h}[u^{**}_{1,i+1,j} - u^{**}_{1,i-1,j} + u^{**}_{2,i,j+1} - u^{**}_{2,i,j-1}].$$

(b) Solve by successive overrelaxation the equation

$$\frac{1}{4h^2}[\varphi_{i+2,j} + \varphi_{i-2,j} + \varphi_{i,j+2} + \varphi_{i,j-2} - 4\varphi_{ij}] = (D\mathbf{u}^{**})_{ij}\,.$$

(c) Evaluate

$$\mathbf{u}^{n+1}_{ij} = (P\mathbf{u}^{**})_{ij} = \mathbf{u}^{**}_{ij} - (G\varphi)_{ij}\,,$$

that is,

$$u^{n+1}_{1,ij} = u^{**}_{1,ij} - \frac{1}{2h}(\varphi_{i+1,j} - \varphi_{i-1,j}),$$

$$u^{n+1}_{2,ij} = u^{**}_{2,ij} - \frac{1}{2h}(\varphi_{i,j+1} - \varphi_{i,j-1}).$$

(4) Finally evaluate

$$\mathbf{x}_k^{n+1} = \mathbf{x}_k{}^n + \delta t \sum_{ij} h^2 \mathbf{u}^{n+1}_{ij}\, D_{ij}(\mathbf{x}_k{}^n).$$

This completes the computation of $\mathbf{x}_k^{n+1}$ and $\mathbf{u}^{n+1}_{ij}$.

IV. Numerical Stability

In the summary which concluded the previous section it was stated that the boundary forces were calculated from the boundary configuration. In fact it is possible to choose this boundary configuration in different ways without violating consistency. The different choices result in numerical schemes with different qualitative behavior. Let $\mathbf{x}_k{}^*$ be the boundary configuration from which the boundary forces are calculated. Then the consistency requirement is

$$\lim_{\delta t \to 0} \mathbf{x}_k{}^* = \mathbf{x}_k{}^n. \tag{4.1}$$

Four distinct choices of $\mathbf{x}^*$ that satisfy (4.1) are

$$\mathbf{x}_k{}^* = \mathbf{x}_k{}^n, \tag{4.2}$$

$$\mathbf{x}_k{}^* = \mathbf{x}_k{}^n + \mathbf{u}^n(\mathbf{x}_k{}^n)\,\delta t, \tag{4.3}$$

$$\mathbf{x}_k{}^* = \mathbf{x}_k{}^n + \mathbf{u}^n(\mathbf{x}_k{}^n)\,\delta t + (h_B/h^2)\,\mathbf{f}_k{}^*(\delta t)^2, \tag{4.4}$$

$$\mathbf{x}_k{}^* = \mathbf{x}_k^{n+1} \tag{4.5}$$

In (4.4) the symbol $\mathbf{f}_k^*$ denotes the boundary forces that correspond to the boundary configuration $\mathbf{x}_k^*$. We remark that (4.2) and (4.3) are explicit in that $\mathbf{x}_k^*$ is calculated from data which are known at the beginning of the time step. By contrast, in (4.4) and (4.5) $\mathbf{x}_k^*$ is defined implicitly, i.e., in terms of something that depends on $\mathbf{x}_k^*$ itself.

The use of Eqs. (4.2) or (4.3) will lead to numerical instability unless the time step taken is exceedingly small. The breakdown that occurs is illustrated in Fig. 2, and may be intuitively understood as an oscillation occurring because the finite time step allows the boundary to move past equilibrium in a single jump under the influence of the initial boundary forces. Moreover the force field $\mathbf{F}$ generated by the boundary is of order h^{-1} even when the boundary forces $\mathbf{f}$ are of order 1, and a change of order h in the position of a boundary point is sufficient to produce a change of order 1 in the corresponding value of $\mathbf{f}$. In short, small changes in configuration produce large changes in forces and stability is difficult to achieve.

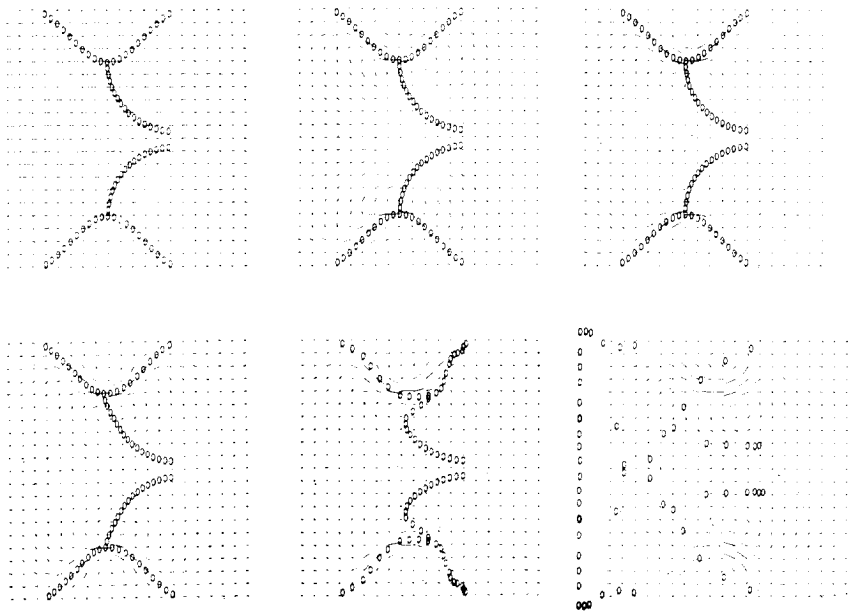

FIG. 2. Numerical instability with an explicit scheme for calculating the boundary forces. At each time step, the initial configuration of the boundary has been used to calculate the boundary forces. Contrast these results with Fig. 5 in which an implicit scheme has been used.

The use of Eq. (4.4) is sufficient to overcome these difficulties; it produces the stable results shown in Section V. This stability is achieved without changing any of the physical or numerical parameters of the problem: In Figs. 2 and 5

the viscosity and the time step are the same, only the formula for calculating the boundary forces is changed, and in both cases the formulas obey the consistency condition Eq. (4.1).

The remainder of this section will be devoted to the algorithm that implements Eq. (4.4). First, we note that the right-hand side of (4.4) is a function of $\mathbf{f}_{ik}{}^*$. We represent this function by the symbol $\tilde{\mathbf{x}}$ so that (4.4) becomes

$$\mathbf{x}^* = \tilde{\mathbf{x}}(\mathbf{f}^*). \tag{4.6}$$

Similarly, we have defined $\mathbf{f}^*$ as the system of forces calculated (by the algorithm of Section III.C) from the boundary configuration $\mathbf{x}^*$. We denote this dependence by

$$\mathbf{f}^* = \mathbf{f}(\mathbf{x}^*). \tag{4.7}$$

Substituting (4.6) in (4.7) we find that Eq. (4.4) yields the following fixed-point problem for the boundary forces $\mathbf{f}^*$:

$$\mathbf{f}^* = \mathbf{f}(\tilde{\mathbf{x}}(\mathbf{f}^*)). \tag{4.8}$$

Remark. In a similar way Eq. (4.5) leads to a fixed-point problem which differs from the foregoing by involving the entire algorithm of Section III.D.

In Eq. (4.8), as $\delta t \to 0$, $\tilde{\mathbf{x}}(\mathbf{f}) \to \mathbf{x}^n$ independent of $\mathbf{f}$, so that $\mathbf{f}^* \to \mathbf{f}(\mathbf{x}^n)$ as is required by consistency.

The system (4.8) can be solved by underrelaxation. That is, we can find γ sufficiently small that the iteration

$$\mathbf{f}^{m+1} = (1 - \gamma)\mathbf{f}^m + \gamma\mathbf{f}(\tilde{\mathbf{x}}(\mathbf{f}^m)) \tag{4.9}$$

converges to $\mathbf{f}^*$. The forces $\mathbf{f}^*$ are used at the beginning of the algorithm of Section III.D, and that algorithm is used only once per time step.

V. Example: Two-Dimensional Simulation of Flow Through the Natural Mitral Heart Valve

We have applied the method of this paper to the problem illustrated in Fig. 3. A valve with a pair of flexible leaflets is set in a narrowing of a periodic pipe. The leaflets are restrained by cords which prevent eversion. The setting, of course, is not realistic but provides a convenient means of testing the method. The square domain of Fig. 3 is assumed to be periodic in the x direction, so that $x = 0$ and $x = 1$ are equivalent. At $y = 0$ and $y = 1$ the flow is assumed to be zero. The "cushions" ABC and $A'B'C'$ have restoring forces which attract them to the equilibrium positions shown, so that their shape changes only slightly throughout

the computation. Each cushion has fluid on both sides. The valve leaflets BD and $B'D'$, on the other hand, are completely flexible. Hence they have forces which only resist stretching but are indifferent to bending. Points D and D' are thought of as connected to P by cords which are slack if shorter than some resting length but which pull D and D' toward P if either point becomes further away from P than this resting length. The fluid is excited by an externally applied body

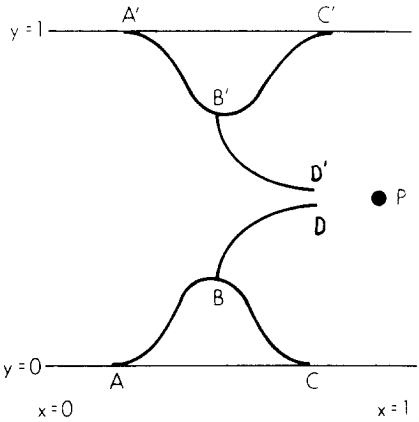

FIG. 3. Two-dimensional representation of the natural mitral heart valve. Elastic cushions ABC, $A'B'C'$ represent the narrowing between atrium and ventricle. Infinitely flexible valve leaflets BD, $B'D'$ are connected by elastic cords (not shown) to the point P.

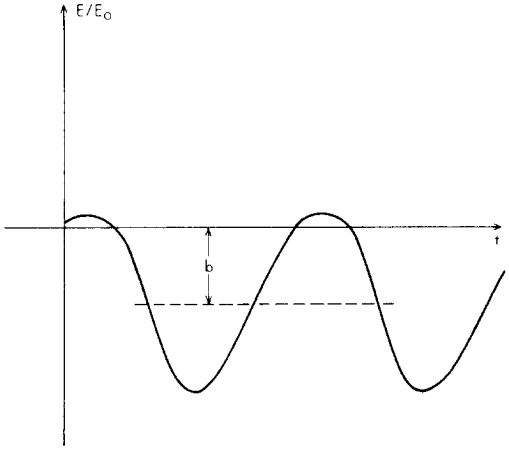

FIG. 4. Applied driving force. The off-center sine wave roughly represents the pressure difference observed between atrium and ventricle with a normal mitral valve.

force which is constant throughout space and points always in the x direction. Its time variation is given by

$$E = E_0(-b + \sin(\omega t + \varphi)) \qquad 0 < b < 1, \tag{5.1}$$

which is plotted in Fig. 4.

Note that the physics of the problem would be unchanged (except for a change in the pressure field) if we did not apply this body force but instead required the pressure to obey a boundary condition of the form

$$p(0, y, t) = p(X, y, t) + XE(t), \tag{5.2}$$

where $E(t)$ is defined by Eq. (5.1) and X is the length of the domain.

Physiologically reasonable parameters for the problem are as follows: $X = 5$ cm, $E_0 = 1.3 \times 10^4$ dyn/cm^3, $b = 0.9$, $\omega = 2\pi$/sec, density $\rho = 1$ gm/cm^3, viscosity $\nu = 0.04$ cm^2/sec. These yield a valve diameter of 3 cm, a maximum pressure difference across the closed valve of about 100 mm Hg, and a peak forward pressure difference of 5 mm Hg. Unfortunately these parameters cannot be used without modification because of inequality (3.7) which restricts the Reynolds number which can be handled by this technique. The physiologic Reynolds number for the human heart is about 2000; from (3.7) we deduce that 10^3 points are needed in each space direction for a true representation of the flow. By raising the viscosity to $\nu = 4$ and leaving the other physical parameters unchanged, we lower the Reynolds number sufficiently to make a practical computation possible. The results, however, are applicable to a whole class of problems; namely, those which differ from the stated problem by a similarity transformation in which $(X^2/\nu T)$ is held constant. The results of Fig. 5, then, are applicable to either a valve of 3-cm diamter in a very viscous fluid $\nu = 4$ cm^2/sec, or to a valve of 0.3-cm diameter in blood of normal viscosity $\nu = 0.04$ cm^2/sec. The latter interpretation is more interesting, because the hearts of different species of mammals certainly differ in size and in Reynolds number over a considerable range, while their valves are in some cases excellent scale models of each other (T. McMahon, R. Frater, personal discussions). Thus we have some reason to hope that the important features of the flow, from the point of view of valve design, are not too sensitive to Reynolds number.

The numerical parameters are as follows: The calculation is performed on a 20×20 mesh. The heartbeat is divided into 2000 time steps, and the computer time required for one heartbeat is about 5000 sec on a CDC-6600. Improvement in the latter figure is to be sought along the following lines. First, we have used 30 sweeps of successive overrelaxation per time step for the solution of Poisson's equation. This unusually large requirement may be connected with the highly localized nature of the boundary forces, since these appear on the right-hand

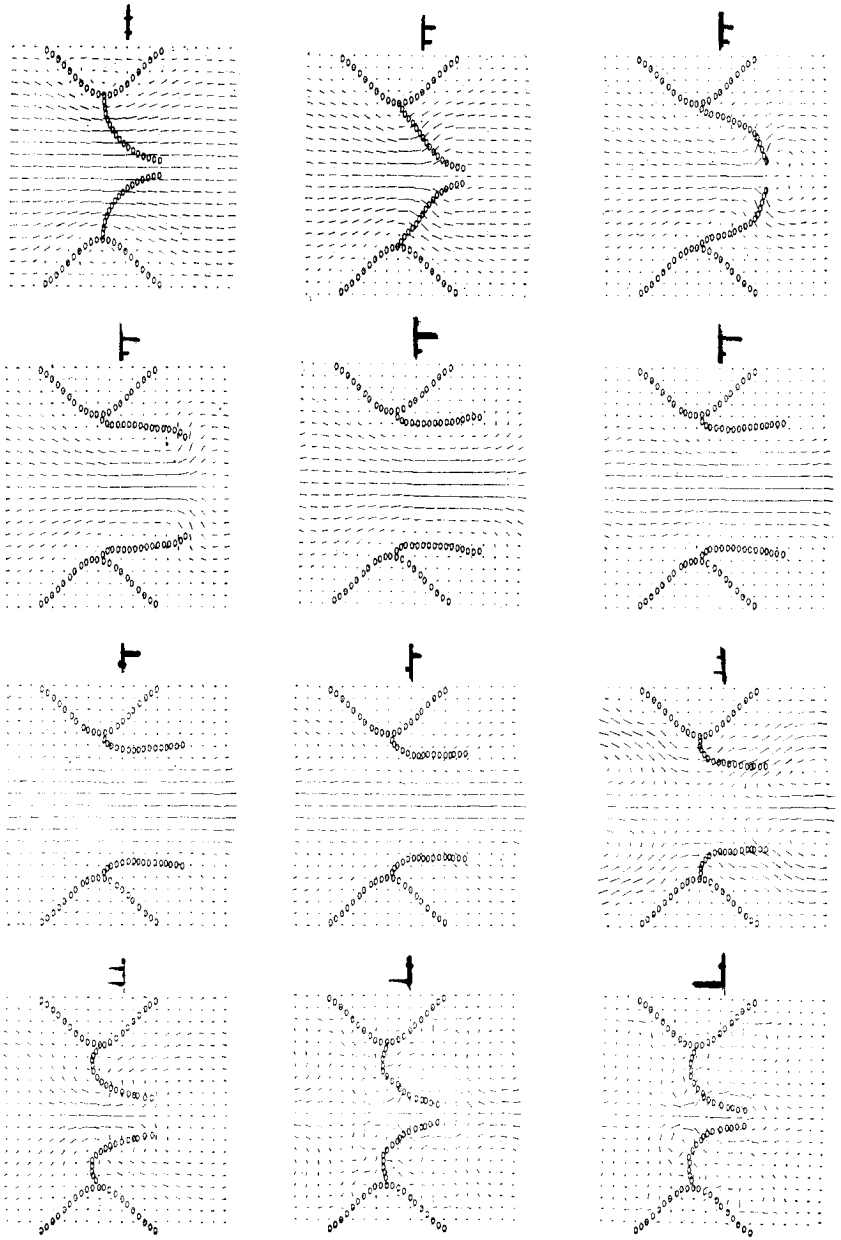

FIG. 5. Opening and closing of the simulated heart valve. Vectors above each frame are net flow (upper) and driving force (lower). Just prior to valve closure, net flow continues forward against adverse driving force. Forward flow persists in the center while backflow begins early at the sides. Hence all the backflow associated with closure is "caught" by the leaflets.

side of Poisson's equation for the pressure. These considerations suggest that direct methods would be preferable; their use would not require any other changes in the algorithm. Second, in the iterative procedure for finding the implicit boundary forces, Eq. (4.9), the range of γ that yields convergence depends on the state of the boundary. Roughly, a highly stressed boundary requires small γ. We are currently working on procedures for having the program choose the optimum γ at each time step.

The opening and closing of the valve is shown in Fig. 5. The frames shown are separated by 40 time steps of computation. In each frame the velocity vectors are normalized to the maximum component of velocity *for that frame*. Thus absolute values of velocity cannot be compared from frame to frame. A pair of horizontal vectors above each frame indicate the magnitude and direction of the flux between $y = 0$ and $y = 1$ (upper vector) and the externally applied force (lower vector). The scale of the latter two vectors is fixed over time so that frame-to-frame comparisons can be made. The period of time shown encompasses the period of forward flow and includes the opening and closing of the valve. Note that the cushions act as an effective boundary as expected though they are represented in the numerical scheme only by the forces they produce. Also note the flow pattern during valve closure. Backward flow begins outside of the main stream at a time when net flow is still positive. Finally net backward flux does occur, but almost all of it is flux through the instantaneous position of the valve leaflet, and very little escapes around the tips of the leaflets. Another way to say this is that the valve leaflets "catch" most of the backflow associated with valve closure. It will be interesting to see whether this is also true for artificial valves.

In Fig. 6 the driving force and the flux between $y = 0$ and $y = 1$ are plotted as functions of time. The following qualitative features are evident:

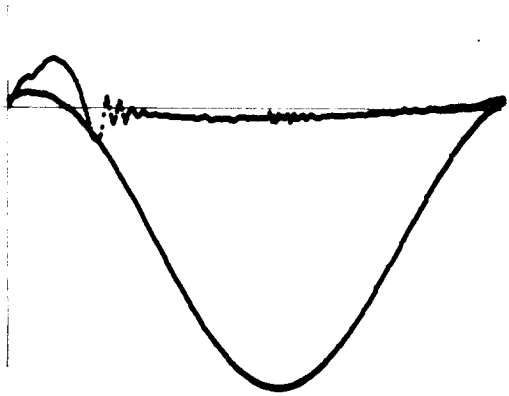

FIG. 6. Driving force and net flow. The oscillation following closure is due to the elasticity of the valve and cords coupled with the mass of the fluid.

(1) The time of peak forward flow is after the time of peak forward driving force and before the time of zero driving force.

(2) Forward flow continues after the driving force has become negative.

(3) Following valve closure there is a damped oscillation in the flow curve. This oscillation is due to the coupling of the valve elasticity to the mass of the fluid.

(4) At the end of the cycle there is a slight shoulder of forward flow which occurs before the driving force becomes positive. This represents the discharge of fluid stored in the ballooned valve leaflets.

All of these features have been previously observed in measurements on the natural mitral valves of dogs [9].

VI. Summary and Conclusions

In the present paper we have extended a previously existing method for solving the Navier–Stokes equations [1, 2] so that it is now possible to find numerical solutions to problems involving immersed boundaries which are moved by the fluid and which simultaneously exert forces on the fluid which moves them. In particular it is hoped that this work will provide a tool for studying flow patterns around heart valves and will therefore aid in the design of artificial valves. The essence of our technique is that the immersed boundary is replaced by an almost singular force field which is calculated at each instant from the boundary configuration. This force field takes on nonzero values only at those mesh points which lie in the immediate neighborhood of the immersed boundary. Otherwise these mesh points are not treated differently from other mesh points of the domain.

The main difficulty has been the calculation of the boundary forces in such a way as to avoid numerical instability in the overall scheme. We have offered a solution to this problem, based on an iterative method for solving a fixed-point problem for the boundary forces.

The technique of this paper uses the primitive variables of pressure, velocity, and boundary configuration. It can, therefore, be generalized at once to encompass three-dimensional problems. On the other hand, the method, at least as presently programmed, is very expensive in computer time and space, so that three-dimensional applications may prove to be impractical.

Considerations of computer time and space also limit the Reynolds number which can be handled by this technique, since the Reynolds number determines the required resolution of the mesh. As discussed in Section V, one simply has to hope that flow at some lower Reynolds number will be an adequate model of

the true flow; support for this idea comes from the variety of Reynolds numbers actually existing in different species of mammals.

In closing we emphasize that the main strength of this technique is the absence of special conditions imposed along the immersed boundary. Because of this, the algorithm is considerably simplified, and the class of problems which can be studied is enlarged.

Acknowledgments

This work was supported by the National Institutes of Health (PHS grant #5T5GM1674) and the Atomic Energy Commission [Contract AT(30-1)-1480] and was closely supervised by Professor A. J. Chorin of the Courant Institute of Mathematical Sciences (NYU). The author is deeply indebted to Professor Chorin for his continuing advice and support on this project, and to the Courant Institute generally for making available computer time and office space. The author also wishes to acknowledge the influence of countless discussions with Dr. E. Yellin of the Albert Einstein College of Medicine concerning the physics and the physiology of blood flow around heart valves.

References

1. A. J. Chorin, On the convergence of discrete approximations to the Navier–Stokes equations, *Math. Comp.* **23** (1969), 341–353.
2. A. J. Chorin, Numerical solution of the Navier–Stokes equations, *Math. Comp.* **22** (1968), 745–762.
3. J. A. Viecelli, A method for including arbitrary external boundaries in the MAC incompressible fluid computing technique, *J. Comput. Phys.* **4** (1969), 543–551.
4. J. A. Viecelli, A computing method for incompressible flows bounded by moving walls, *J. Comput. Phys.* **8** (1971), 119–143.
5. B. L. Buzbee, G. H. Golub and R. W. Nielson, On direct methods for solving Poisson's equation, *SIAM J. Numer. Anal.* **7** (1970), 627–655; R. W. Hockney, The potential calculation and some applications, *Methods Comput. Phys.* **9** (1970), 135–211.
6. R. D. Richtmyer and K. W. Morton, "Difference Methods for Initial Value Problems," 2nd. ed., pp. 198–201, Wiley, New York, 1967.
7. A. S. Householder, "The Theory of Matrices in Numerical Analysis," pp. 123–124, Blaisdell Publishing Co., New York, 1965.
8. R. S. Varga, "Matrix Iterative Analysis," pp. 97–129, Prentice Hall, Englewood Cliffs, N.J., 1962.
9. E. L. Yellin, R. W. M. Frater, C. S. Peskin, and W. H. Epstein, Dynamics of flow across natural mitral valve, ASME 92nd Winter Meeting 71-WA/BHF-2.