JOURNAL OF COMPUTATIONAL PHYSICS 37, 113–132 (1980)

Modeling Prosthetic Heart Valves for Numerical Analysis of Blood Flow in the Heart

CHARLES S. PESKIN AND DAVID M. MCQUEEN

Courant Institute of Mathematical Sciences, 251 Mercer Street, New York, New York 10012

Received August 2, 1979; revised October 18, 1979

This paper extends our previous work on numerical analysis of blood flow in the heart. In that work the boundary forces were evaluated by solving a fixed-point problem, which we now reformulate as a problem in optimization. This optimization problem, which involves the energy function from which the boundary forces are derived, is solved by Murray's modification of Newton's method. The energy function turns out to be an extremely useful tool in modeling prosthetic heart valves. To enforce a constraint on the valve, we use an energy function which is zero when the constraint is satisfied and positive otherwise. The energy function must be invariant under translation and rotation so that conservation of momentum and angular momentum will be satisfied. We use this technique to construct computer models of several prosthetic valves, and we study the flow patterns of these valves in our computer test chamber.

1. INTRODUCTION

In [1], Peskin described a numerical method for blood flow in the heart with special emphasis on the heart valves. This method is based on an Eulerian representation of the fluid on a fixed computational mesh and a Lagrangian representation of the immersed, elastic boundaries that interact with the fluid. An important aspect of the method is the computation of the boundary forces from the boundary configuration. To secure numerical stability, it is necessary to calculate these forces implicitly, as the solution of a certain nonlinear fixed-point problem.

This fixed-point problem is solved in [1] by Newton's method, and it is proved there that the relevant (symmetric) matrix is positive definite if the links that define the physical properties of the boundary satisfy certain sufficient conditions. One of these conditions states that the links do not resist compression.

The modeling of prosthetic heart valves requires links that resist compression, so the methods of [1] are not guaranteed to work in the case of prosthetic heart valves. This rather serious shortcoming is completely overcome in the present paper.

The way out of the difficulty described above was suggested to us by Olof Widlund, whose help we gratefully acknowledge.

The first step is to reformulate the fixed-point problem as a problem of finding a stationary point of a certain function of the boundary configuration. It can then be shown that this function has an absolute minimum, in the neighborhood of which the

0021-9991/80/100113–21$02.00/0

Hessian matrix is, of course, positive definite. There may be other regions, however, where this matrix is indefinite.

The modified Newton's method introduced by Murray [2] is specifically designed for this situation. Murray's method reduces to Newton's method in the positive definite regions and it generates a downhill search direction in the indefinite regions. The method requires a Cholesky factorization of the Hessian matrix by *columns*, and it modifies the matrix during the factorization if a negative (or small positive) element appears on the diagonal. In our case, it is very convenient to store the lower triangle of the Hessian matrix in the envelope or profile storage scheme [3] by *rows*. Accordingly, we need a special subroutine for the Cholesky factorization by columns when the matrix is stored in this way. It turns out that this can be achieved at the expense of doubling the amount of storage allocated for the matrix, and the resulting method is actually faster than a standard profile factorization because fewer tests are required.

The latter part of this paper is concerned with an extremely useful by-product of the reformulation of our fixed-point problem as a minimization problem. The key step in this reformulation is the recognition that the boundary forces are derivatives of an energy function. This observation makes it easy to generalize the notion of a link structure that was used in [1] and to introduce forces that are functions of the coordinates of triples of points instead of pairs. Such forces are needed to simulate the constraints on the motions of artificial valves that are imposed by the cages in which they move. Forces that depend on triples of points are also very convenient in the modeling of structures that resist bending. To impose a constraint (approximately) on a triple of points, we simply introduce an energy function of the triple that is invariant under translations and rotations and that achieves its minimum when the constraint is satisfied.

Finally, we use the methods developed in this paper to model a variety of specific prosthetic valves, and we study the flow patterns of these valves in a computer test chamber that models the left heart. For comparison, we also present the computed flow patterns of the natural mitral valve, and we give evidence that the conditions of pressure and flow in the computer test chamber are physiological.

2. The Fixed-Point Problem and Its Reformulation

In [1] we describe the immersed boundary by a collection of points $(\mathbf{x}_1 \cdots x_N)$ and the boundary forces by certain functions of the boundary configuration

$$\mathbf{f}_k(\mathbf{x}_1 \cdots \mathbf{x}_N), \qquad k = 1, 2, \ldots, N. \tag{2.1}$$

The forces actually applied to the fluid, however, are not obtained directly by evaluating these functions. Instead, a nonlinear system of the form

$$\mathbf{x}_k^* = \mathbf{x}_k^0 + \lambda \mathbf{f}_k(\mathbf{x}_1^* \cdots \mathbf{x}_N^*), \qquad k = 1, 2, \ldots, N, \tag{2.2}$$

with $\lambda > 0$ is solved by Newton's method, and the forces used are given by

$$\mathbf{f}_k^* = \mathbf{f}_k(\mathbf{x}_1^* \cdots \mathbf{x}_N^*). \tag{2.3}$$

In these equations $\mathbf{x}_k^0$ stands for $\mathbf{x}_k^n + \Delta t\, \mathbf{u}_k^n$, where the superscript n indicates $t = n\,\Delta t$ and where $\mathbf{u}_k^n$ is the velocity of the kth boundary point at this time. Thus $\mathbf{x}_k^0$ is known at the beginning of the time step from time level n to $n+1$. If λ is chosen appropriately, then $\mathbf{x}_k^*$ is approximately the position of the kth boundary point at the end of this step; it is a better approximation than $\mathbf{x}_k^0$ because it includes the effect of the boundary forces. Since we evaluate these forces at the configuration $(\mathbf{x}_1^*, \ldots, \mathbf{x}_N^*)$, we are using an (approximately) implicit scheme. This implicit method of calculating the boundary forces is needed for numerical stability. A more formal derivation of (2.2) is given in [1]. In this paper we are concerned with the technical details involved in the solution of (2.2).

Now suppose that the forces can be written in terms of an energy function $E(\mathbf{x}_1 \ldots \mathbf{x}_N)$ with continuous first derivatives. That is,

$$\mathbf{f}_k = -\partial_k E = -\left(\frac{\partial E}{\partial x_{k1}}, \frac{\partial E}{\partial x_{k2}} \right), \tag{2.4}$$

where ∂_k is the gradient with respect to the coordinates of the kth boundary point. It will be important in the following that E be bounded from below. In fact, we shall always have $E \geqslant 0$.

Let

$$\phi(\mathbf{x}_1 \cdots \mathbf{x}_N) = \tfrac{1}{2} \sum_{k=1}^{N} |\mathbf{x}_k - \mathbf{x}_k^0|^2 + \lambda E(\mathbf{x}_1 \cdots \mathbf{x}_N). \tag{2.5}$$

Then the stationary points of ϕ are precisely the solutions of the fixed-point problem (2.2).

Let x stand for the configuration $(\mathbf{x}_1 \cdots \mathbf{x}_N)$ and let $\| \ \|$ be the Euclidean norm. Then (2.5) reads

$$\phi(x) = \tfrac{1}{2} \|x - x^0\|^2 + \lambda E(x). \tag{2.6}$$

Now ϕ is a continuous function with the property $\phi(x) \to +\infty$ as $\|x\| \to \infty$, since E is bounded from below. It follows that ϕ has an absolute minimum at some point x^*. That is, $\phi(x^*) \leqslant \phi(x)$ for all x; we do not claim that x^* is unique. Since ϕ is differentiable, x^* is a solution of our fixed-point problem (2.2).

The significance of this result is as follows. First, it tells us that our fixed-point problem has at least one solution. There may be many other solutions including local maxima, saddle points, and local minima. Moreover, since the existence of at least one minimum is guaranteed, it makes sense to use an iterative method that searches for minima instead of a more complicated method that searches for stationary points in general.

3. The Modified Newton's Method and Its Implementation

In this section we briefly describe the modification of Newton's method that was introduced by Murray [2], and we discuss the implementation of Murray's algorithm in our case.

We seek a local minimum of a function $\phi(x)$, where ϕ is real and x is a point in R^n. Let g be the gradient of ϕ, and let $G = G^T$ be the Hessian matrix (matrix of second derivatives) of ϕ. The modified Newton's method is an iteration of the form

$$x^{(k+1)} = x^{(k)} + \alpha^{(k)} p^{(k)}, \tag{3.1}$$

where $p^{(k)}$ is the solution of

$$(G^{(k)} + D^{(k)}) p^{(k)} = -g^{(k)} \tag{3.2}$$

(with $D^{(k)}$ a diagonal matrix discussed below), and where $\alpha^{(k)}$ is a positive scalar chosen to minimize $\phi^{(k+1)}$. In the terminology of optimization, $p^{(k)}$ is a search direction, and the process of determining $\alpha^{(k)}$ is called line search.

System (3.2) is solved with the aid of the Cholesky factorization

$$G^{(k)} + D^{(k)} = L^{(k)}(L^{(k)})^T, \tag{3.3}$$

where $L^{(k)}$ is a lower triangular matrix. In fact, the matrix $D^{(k)}$ is constructed *during* the factorization, in such a way that the following criteria are satisfied

(i) $(G^{(k)} + D^{(k)})$ is positive definite. (Otherwise the factorization does not exist.)

(ii) The diagonal elements of $L^{(k)}$ are greater than some positive constant δ.

(iii) The off-diagonal elements of $L^{(k)}$ are bounded in magnitude by some preassigned constant β.

The positive definiteness of $G^{(k)} + D^{(k)}$ guarantees the fact that $p^{(k)}$ is a downhill search direction. To see this, just multiply both sides of (3.2) by $(p^{(k)})^T$. In fact, when the first property is satisfied, we may say (following Murray) that $p^{(k)}$ is a steepest descent direction with respect to the norm

$$\|y\|^2 = y^T(G^{(k)} + D^{(k)})y. \tag{3.4}$$

The remaining two properties are needed to ensure the numerical stability of the algorithm.

When $G^{(k)}$ is sufficiently positive definite, Murray's algorithm automatically sets $D^{(k)} = 0$, and the method reduces to Newton's method. This property of the algorithm depends on an appropriate choice of β, as discussed in [2].

The rest of this section will be concerned with an important detail of implementation. In Murray's algorithm [2], the construction of the lower triangular factor $L^{(k)}$ is carried out by *columns*, since a given diagonal element has a direct effect on the size of the off-diagonal elements in its column. In our case, however, it is very convenient to use the profile storage scheme [3] for the matrix $G^{(k)}$. This storage scheme uses two one-dimensional arrays A and $IDIA$. In A, the elements of the lower triangle of $G^{(k)}$ are stored by rows. In each row, however, only the elements from the first non-zero to the diagonal element are stored. We shall call these the active elements; they are precisely the elements that can be changed during the factorization. The array $IDIA$ is a list of pointers to the diagonal elements of the matrix. Thus the active element (I, J) is stored in $A(IDIA(I) - I + J)$

Unfortunately, this data structure is very inefficient for the task of locating all of the active elements in a given column. In our application, only a few rows (which are not adjacent) actually have active elements in any given column, but all of the rows would have to be tested to locate these active elements. This difficulty is overcome at the cost of doubling the storage requirements. We introduce two auxiliary one-dimensional arrays LRI and $LDIA$. The array LRI has the same length as A and it refers to the same active elements of the matrix. While A stores the active elements in their natural order by rows, however, LRI stores the row indices of the active elements in their natural order by columns. The array $LDIA$ is a list of pointers to the diagonal elements in the list LRI.

In our application [1], the graph of the matrices $G^{(k)}$ is independent of k. Moreover, the graph is also constant from one time step to the next of the overall numerical method. Accordingly, the arrays LRI, $LDIA$, and $IDIA$ are fixed integer arrays that can be initialized at the beginning of the run, so the cost of initialization is not important.

To illustrate the use of this data structure, we give the FORTRAN fragment corresponding to

$$L(I, J) = L(I, J) - L(I, K) * L(J, K), \tag{3.5}$$

which is the heart of the Cholesky factorization. In the following, the indices I and J are row indices of two active elements in column K. These elements have been located by generating two integers, LI and LJ, which satisfy

$$LDIA(K) < LI \leqslant LJ < LDIA(K+1). \tag{3.6}$$

Then,

$$I = LRI(LI), \tag{3.7}$$

$$J = LRI(LJ), \tag{3.8}$$

$$IJ = IDIA(I) - I + J, \tag{3.9}$$

$$IK = IDIA(I) - I + K, \tag{3.10}$$

$$JK = IDIA(J) - J + K, \tag{3.11}$$

$$A(IJ) = A(IJ) - A(IK) * A(JK). \tag{3.12}$$

With this data structure, the entire Cholesky factorization can be carried out by columns within the framework of the profile storage scheme. No IF statements are required except for the test for an empty column (i.e., a column with no active elements other than the diagonal element). It is also very easy to modify such a code so that it performs the factorization required by Murray's algorithm. The resulting subroutine is actually faster than a standard profile factorization by rows, since the latter method requires a significant amount of testing.

4. Three-Point Forces

In this section, we take further advantage of the fact that the boundary forces can be written as gradients of an energy function. In particular, we introduce a systematic procedure for modeling the constraints associated with rigid prosthetic valves. These constraints will not be imposed exactly, however. Instead, we introduce boundary forces which tend to drive the boundary points toward a configuration that satisfies the constraints.

In this context it is very convenient and sometimes indispensable to use forces that depend on the configuration of triples of points. Such forces are described by functions

$$\mathbf{f}_k(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3), \qquad k = 1, 2, 3. \tag{4.1}$$

The functions $\mathbf{f}_k$ are not arbitrary, however, but must be chosen to conserve momentum and angular momentum. That is,

$$\sum_{k=1}^{3} \mathbf{f}_k(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3) = 0, \tag{4.2}$$

$$\sum_{k=1}^{3} \mathbf{x}_k \times \mathbf{f}_k(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3) = 0. \tag{4.3}$$

These conditions are automatically satisfied if we choose an energy function $E(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3)$ that is invariant under translations and rotations and then set

$$\mathbf{f}_k = -\partial_k E, \tag{4.4}$$

as in Section 2.

The choice of three-point forces to enforce a constraint is therefore reduced to the choice of a three-point energy function. This function should be invariant under translation and rotation, it should take on the value zero when the constraint is satisfied, and it should be positive otherwise. The rest of this section will be concerned with specific functions of this kind that are useful in modeling prosthetic heart valves.

First, we consider the cages that restrict the motions of prosthetic valves. Ball valves, for example, move in a cage that forces the ball to move along a line perpendicular to the plane of the valve ring. In a two-dimensional model, the valve ring is represented by two boundary points $\mathbf{x}_1$ and $\mathbf{x}_2$, We require that the center of the ball, $\mathbf{x}_3$, move along the perpendicular bisector of the line segment joining $\mathbf{x}_1$ and $\mathbf{x}_2$. Let

$$\mathbf{a} = \mathbf{x}_2 - \mathbf{x}_1, \tag{4.5}$$

$$\mathbf{b} = \mathbf{x}_3 - \tfrac{1}{2}(\mathbf{x}_1 + \mathbf{x}_2). \tag{4.6}$$

The constraint described above can then be formulated as $\mathbf{a} \cdot \mathbf{b} = 0$, and an energy function which enforces this constraint is

$$E(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3) = \tfrac{1}{2}C(\mathbf{a} \cdot \mathbf{b})^2, \tag{4.7}$$

where C is a constant. For large C, the constraint will be rather strictly enforced. In practice, stability considerations set an upper limit on C. Obviously, E is invariant under translation and rotation. It takes on the value zero if the constraint is satisfied, and it takes on positive values otherwise. Note that E is a polynomial in the coordinates of the points $\mathbf{x}_1$, $\mathbf{x}_2$, and $\mathbf{x}_3$. This makes it easy to compute the first and second derivatives as required for Newton's method.

Next, consider the problem of keeping a sequence of points equally spaced along a straight line, as in the two-dimensional representation of a rigid disc. An energy function that achieves this purpose can be written as a sum of three-point functions:

$$E(\mathbf{x}_1 \cdots \mathbf{x}_N) = \frac{1}{2} C \sum_{k=1}^{N-1} |\mathbf{x}_{k+1} + \mathbf{x}_{k-1} - 2\mathbf{x}_k|^2 h^{-3}, \tag{4.8}$$

where $h = 1/N$ and C is a constant. Again, E has all of the required properties. In particular, $E = 0$ when the points $\mathbf{x}_0 \cdots \mathbf{x}_N$ are equally spaced along a straight line. The scale of the configuration of points satisfying $E = 0$ is arbitrary, however, since E is a homogeneous (quadratic) function of the coordinates. This degeneracy can be removed, for example, by introducing a link that fixes the distance between $\mathbf{x}_0$ and $\mathbf{x}_N$.

It may be worth mentioning that (4.8) is an approximation to the formula

$$\mathrm{E} = \frac{1}{2} C \int_0^1 \left| \frac{d^2\mathbf{x}}{ds^2} \right|^2 ds, \tag{4.9}$$

which is the energy function of a linear beam.

The energy function in each term of (4.8) keeps three points equally spaced along a straight line. This can be generalized in two different ways. Suppose that the point $\mathbf{x}_3$ should be on the line joining $\mathbf{x}_1$ and $\mathbf{x}_2$, but at some point other than the midpoint of that line. Then we use

$$E = \tfrac{1}{2}C\,|\mathbf{x}_3 - \alpha\mathbf{x}_1 - (1 - \alpha)\,\mathbf{x}_2|^2. \tag{4.10}$$

This construction is useful in the representation of a pivoting disc valve, which pivots about a point in the plane of the valve ring.

The other generalization involves a departure from the straight-line configuration. Suppose that the point $\mathbf{x}_3$ should be at the apex of an isosceles triangle whose base is the line segment joining $\mathbf{x}_1$ and $\mathbf{x}_2$. Let β be the required ratio of height to base, and let $\hat{\mathbf{z}}$ be a unit vector perpendicular to the plane. The required energy function is

$$E = \tfrac{1}{2}C\,|\mathbf{x}_3 - \tfrac{1}{2}(\mathbf{x}_1 + \mathbf{x}_2) - \beta\hat{\mathbf{z}} \times (\mathbf{x}_2 - \mathbf{x}_1)|^2. \tag{4.11}$$

This is very useful in the representation of curved, rigid valves.

Finally, we remark that one-sided (inequality) constraints can also be enforced in this way. Suppose that the point $\mathbf{x}_3$ should lie in the half-space to the right of the vector from $\mathbf{x}_1$ to $\mathbf{x}_2$. That is,

$$D = \hat{\mathbf{z}} \cdot (\mathbf{x}_3 - \mathbf{x}_1) \times (\mathbf{x}_2 - \mathbf{x}_1) \geqslant 0. \tag{4.12}$$

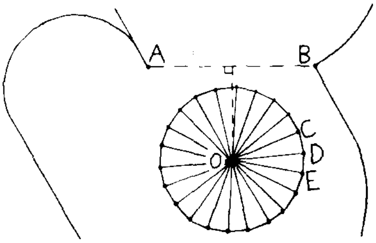

FIG. 1. Computer model of the caged ball valve. The circular shape is maintained by circumferential links (CD, DE,...) and radial links (OC, OD, OE,...). The lateral constraint of the cage is modeled by a three-point force which has the effect of keeping point 0 on the perpendicular bisector of AB; see Eqs. (4.5)–(4.7). The downward excursion of the ball is limited by links AO and BO, which do not resist compression. The upward excursion is limited by a one-sided three-point force which prevents the point O from entering the half-plane above the line AB; see Eqs. (4.12)–(4.13). Link AB maintains the diameter of the mitral ring.

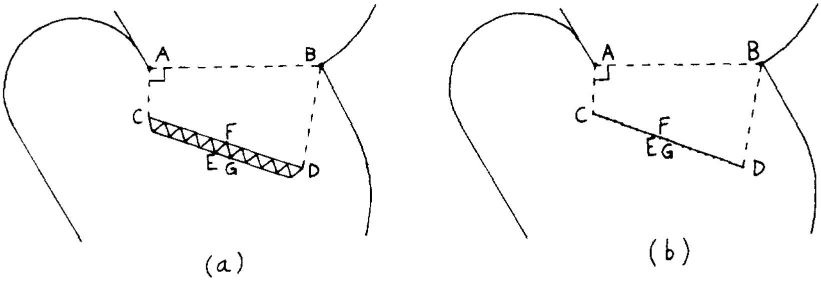

FIG. 2. Computer models of the eccentric monocusp valve. In (a) the rigidity of the valve is maintained by a triangular network of links (EF, FG, EG,...) in a "railroad bridge" arrangement. In (b) it is maintained by three-point forces involving the triples EFG,...; see Eq. (4.8). As remarked following Eq. (4.8), a link CD is also needed in (b) to fix the length of the occluder. In both cases, the point C is constrained by a three-point force to slide along a line perpendicular to AB at A, and the limits of downward excursion are fixed by links AC and BD, which do not resist compression. Link AB maintains the diameter of the mitral ring.

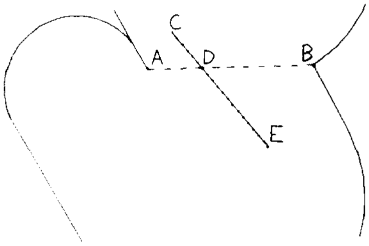

FIG. 3. Computer model of the pivoting disc valve. The points of the occluder are held in a straight line by three-point forces with a single link fixing the overall length as in Fig. 2b. The pivot point D is held at a fixed point along the line AB by a three-point force of the type defined by Eq. (4.10). Point E is prevented from entering the half-plane above the line AB by a one-sided three-point force; see Eqs. (4.12), (4.13). We do not set any limitation on the angle of opening this valve, but such a limit could be set by a link BE. Link AB maintains the diameter of the mitral ring.

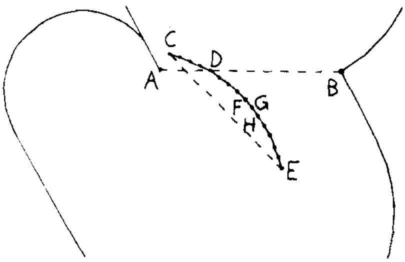

FIG. 4. Computer model of the curved pivoting disc valve. The points of the occluder (FGH,...) are held in a circular arc by three-point forces of the type defined by Eq. (4.11), and a link CE is used to fix the overall length of this arc. Other features are the same as in Fig. 3.

An energy function which maintains this constraint is

$$\begin{aligned} E &= \tfrac{1}{2}CD^2, \quad & D < 0, \\ &= 0, & D \geqslant 0. \end{aligned} \tag{4.13}$$

This construction can be used to model the stops that limit the motion of prosthetic valves.

We have used the energy functions specified above in the computer modeling of several prosthetic mitral valves. These are depicted in Figs. (1–4) and their computed flow patterns are shown in the next section.

5. Results and Discussion

In [1], we described the representation of the left side of the heart for our computer studies of the mitral valve. The computer test chamber that we used in that paper will also be used here. It consists of an atrium and a ventricle, the walls of which have the mechanical properties of heart muscle. There is a source of blood in the atrium which represents the return of blood to the left heart through the pulmonary veins. The entire structure is floating in fluid and there is a sink around the edges of the domain which accepts the volume displaced as the heart fills. We have checked that the pressure outside the heart is essentially constant (zero) in these computations, so we feel confident that the external fluid is not having any important influence on the results. Computer models of the mitral valve, natural or artificial, are mounted at the junction of the atrium and the ventricle.

The physiology of our computer experiments may be described as follows. Each computation begins during the rapid relaxation of the ventricle at the moment when the falling ventricular pressure crosses the atrial pressure. As the ventricular muscle continues to relax, the ventricular pressure falls below that of the atrium, mitral flow begins, and the mitral valve opens. Mitral flow reaches a peak and then slows down as the ventricular and atrial pressures equilibrate. The next major event is atrial systole (contraction of the atrium) which produces a second peak of mitral flow and which is rapidly followed by ventricular systole and valve closure. The computation ends early in ventricular systole, before the ventricular pressure is high enough to open the aortic valve. For this reason we can avoid modeling the aortic valve for these computations, and the outflow tract of our model ventricle has a permanently closed end.

In choosing the parameters of our computer test chamber we made an effort to establish physiological flow conditions. A detailed report on this effort will be published elsewhere, but the following remarks may be helpful here. First, we used the dog as a standard rather than man, since experiments on heart valves are usually done in dogs. In our choice of parameters, we were guided by the results of such experiments. In particular, we used unpublished records of left atrial pressure, left ventricular pressure, and mitral flow as functions of time from the laboratory of E. L. Yellin; see [4].

Unfortunately, we cannot model these experiments directly. The major restrictions are low Reynolds number and two space dimensions. We lower the Reynolds number by scaling all lengths and times down by the factor $\gamma = 1/25$ from their physiological values. Since we retain the physiological density and viscosity of blood, this also scales the Reynolds number down by the factor γ. The scaling operation changes the problem, so there is no rigorous reason to believe that the results are applicable to the dog heart. Nevertheless, it is important to compare our results with experiments. We can make a reasonable comparison if we first apply the inverse scaling to the computed solution. When this is done the theoretical and experimental results agree rather well (see below). This suggests, among other things, that the scaling procedure does not actually change the problem in an important way. It would be better, of course, if the computation could be performed directly at physiological Reynolds numbers. An important step in this direction has already been taken by McCracken and Peskin [5], where we use a vortex method and perform computations for the natural mitral valve at Reynolds numbers up to the physiological value. In the present work on prosthetic valves we use a finite difference method [6, 7] for the fluid dynamics, and this imposes the restriction on the Reynolds number that has been described above. We see no reason why the methods of this paper could not be combined with vortex methods, but this has not yet been tried.

The restriction of two space dimensions introduces a certain difficulty with respect to the use of mitral flow records. The quantity measured in Yellin's experiments is the volume of blood per unit time crossing the plane of the mitral ring. The corresponding quantity in a two-dimensional calculation has units of area per unit time. This introduces an arbitrariness in the scale of the flow when the computed and experimental flows are compared.

There is no such ambiguity in the pressure, and we set up the initial conditions for our computation in such a way that the common atrial and ventricular pressure at the initial time agrees with the experiental value. The time scale is also unambiguous, and we choose the time of onset of atrial and ventricular systole to correspond with the experimental records.

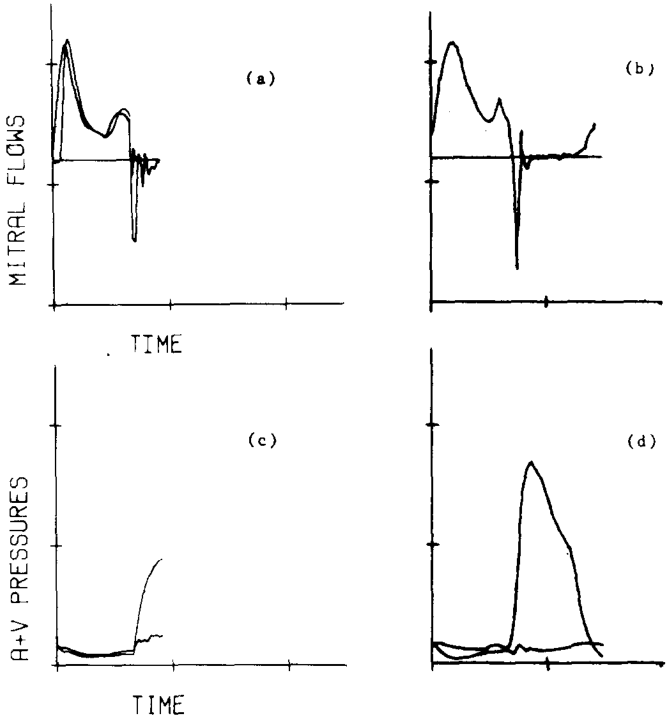

FIG. 5. Comparison of theory and experiment. Natural mitral valve. This figure shows computed (a, c) and experimental (b, d) records of flow and pressure as functions of time. Time scales and origin are the same in all cases. Pressure scales are also the same, but flow scales are arbitrary. Experimental records cover a full cardiac cycle, but the computed records stop in early ventricular systole. The two computed curves in (a) are records of flux through the valve ring and the tips of the leaflets, respectively. The latter flow (which is not available experimentally) starts later and shows no backflow during closure. Ex-

In the following we give the results of computer experiments on the natural mitral valve and on various prosthetic valves. The numerical parameters for these experiments are the same as in [1] except that we refine the time step by a factor of 2 during ventricular systole. Because of several improvements in the program, some of which are described in this paper, the computer time required for each run has been reduced from 120 to 40 min on a CDC 6600.

Figure 5 shows the computed and experimental records of mitral flow and of atrial

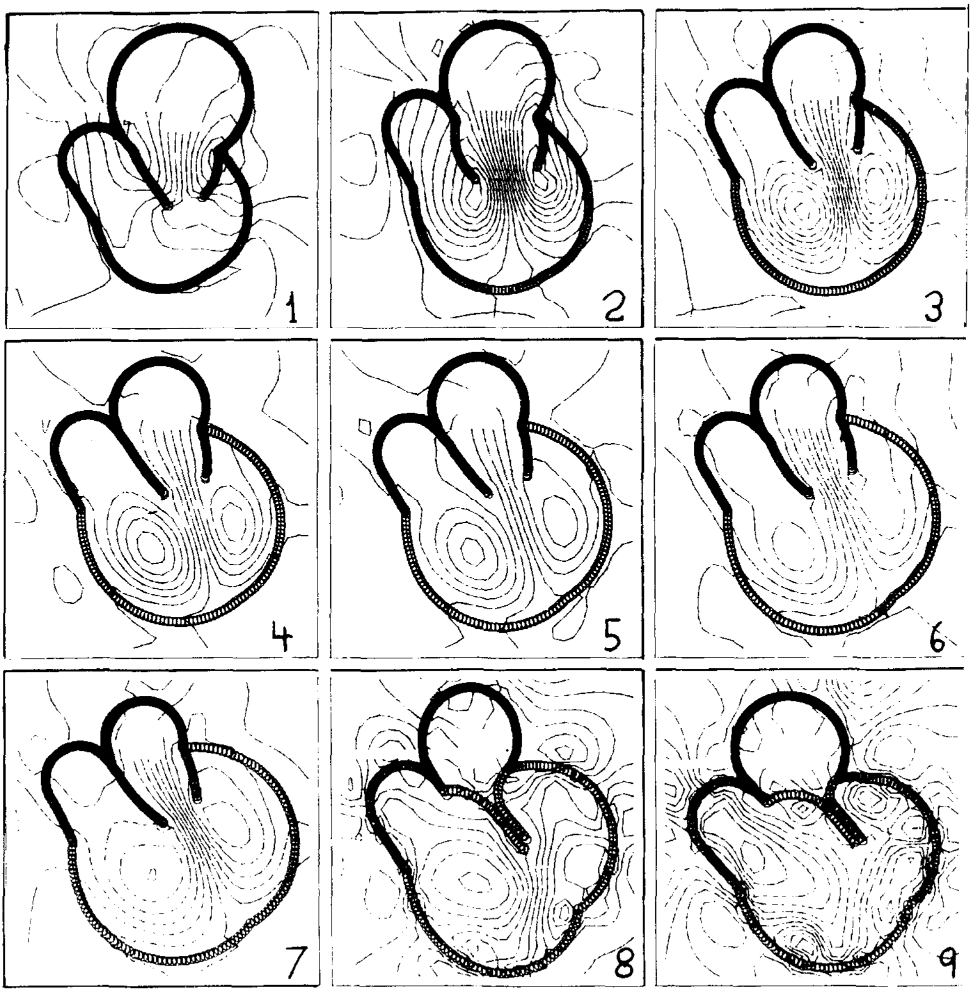

FIG. 6. Flow patterns of the natural mitral valve. This figure and those that follow show computec streamlines at equally spaced times. In frame (1) the valve is opening; note that streamlines cross thε leaflets. Vortices are shed from the tips of the leaflets in frame (2), and the vortex streamlines move th‹ leaflets toward closure. Frames (3–5) show the characteristic flow pattern of ventrcular filling; note tha fluid moves toward the ventricle on both sides of the larger (anterior) leaflet. In frames (6–7) atria systole strenghtens the jet of forward flow and reinforces the vortex system. Closure has just begun ir frame (7); it is completed by ventricular systole in frame (8).

and ventricular pressure for the natural mitral valve. Two flow waveforms are plotted in the computed case (Fig. 5a). One curve gives the flow crossing the valve ring; this is the quantity that is measured in experiments. The other gives the flow passing between the tips of the leaflets. These computed flows agree with each other during most of ventricular diastole, but the flow at the tips starts later when the valve is opening and it shows almost no trace of backflow during the closure movement of the valve. This proves that the spurt of backflow that appears at the valve ring involves fluid displaced by the moving leaflets, not fluid escaping between the tips of the leaflets.

In general, the waveform of the computed mitral flow is similar to the experimental

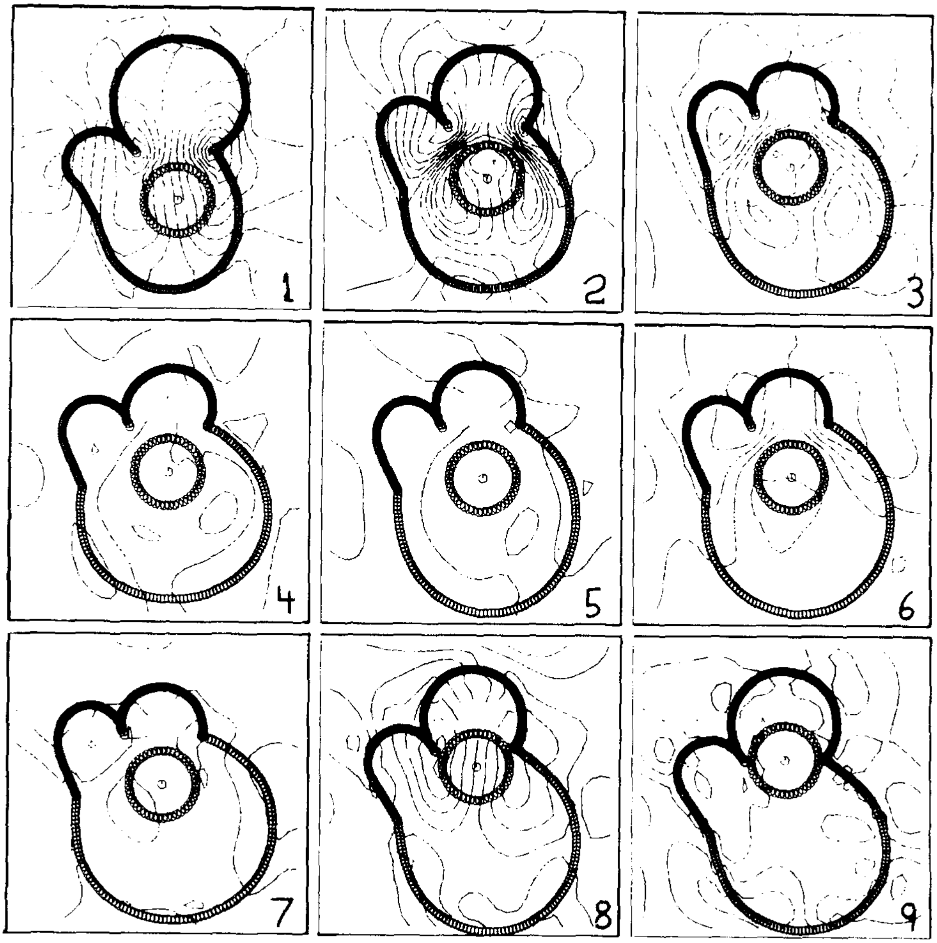

FIG. 7. Flow patterns of the caged ball valve. The computer model of this valve is shown in Fig. 1. Note the distinction between the flow pattern when the ball is moving forward (1) and after it has hit the end of the cage (2). Similarly, compare the closing pattern (8) with the closed flow pattern (9).

waveform. The computed flow reaches is peak at an earlier time, however, and then falls more rapidly.

The computed and experimental pressure waveforms also have several features in common. In both cases, the pressures start together, separate as the ventricle relaxes and mitral flow begins, equilibrate as the mitral flow decreases, and separate again at the onset of atrial systole and the second peak of mitral flow. When the ventricular pressure rises dramatically at the onset of ventricular systole, the atrial pressure is protected from rising by the closed mitral valve, but the atrial pressure waveform shows some characteristic bumps that can be seen in both cases. The most important

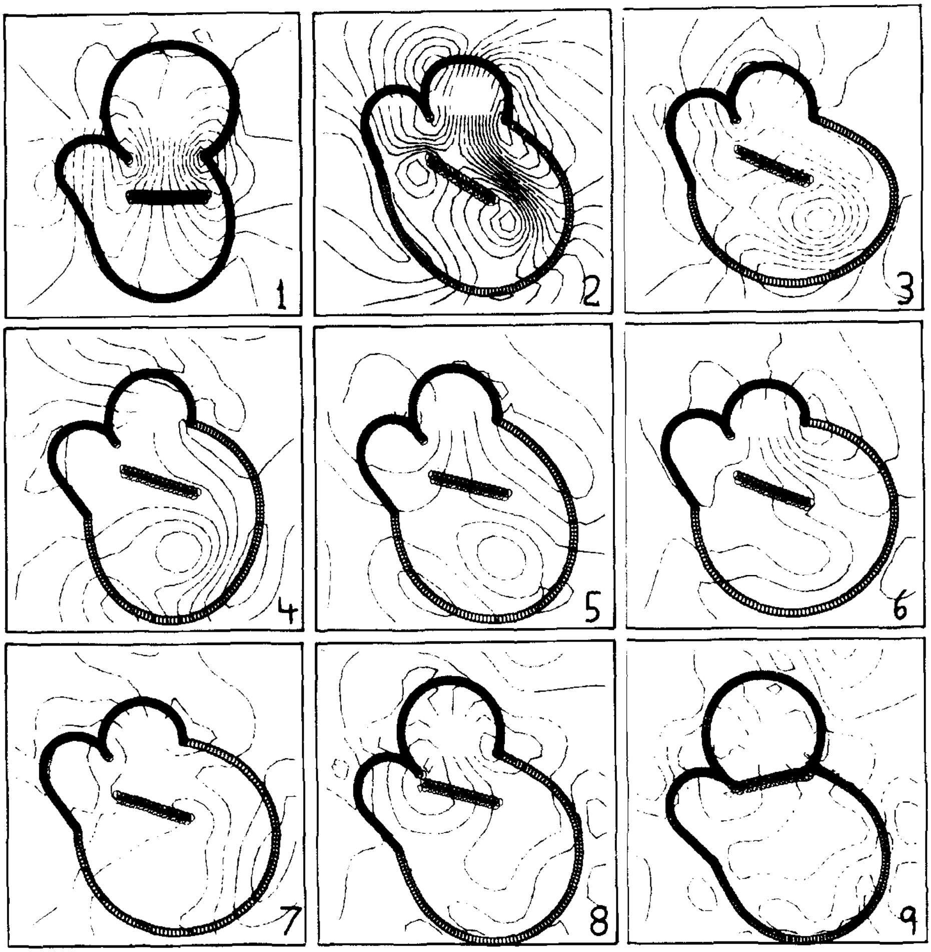

FIG. 8. Flow patterns of the eccentric monocusp (a). The computer model is shown in Fig. 2a. This valve slides forward until it hits a stop; then it rotates to a maximum angle. Note that flow goes around both sides of the occluder (2). There is a definite change of shape of the ventricle in response to the strong jet striking the posterior wall in frame (3). Although a vortex is formed (2–5), it is not helpful in closing the valve, which closes late (9).

quantitative difference is that the computed atrial and ventricular pressures are closer together than the corresponding experimental pressures during ventricular diastole. If we interpret the flows as being the same (which is questionable for reasons that have been discussed above), this means that the model valve offers less resistance to forward flow than the real mitral valve. This could be a consequence of using a two-dimensional model. It is possible, however, that part of the artifact is on the experimental side, since the flow meter that is used to make the measurements contributes an unknown amount to the resistance of the valve.

Computed flow patterns of the natural mitral valve are shown in Fig. 6, and computed flow patterns for the various prosthetic valves which are the subject of this paper are shown in Figs. 7–12.

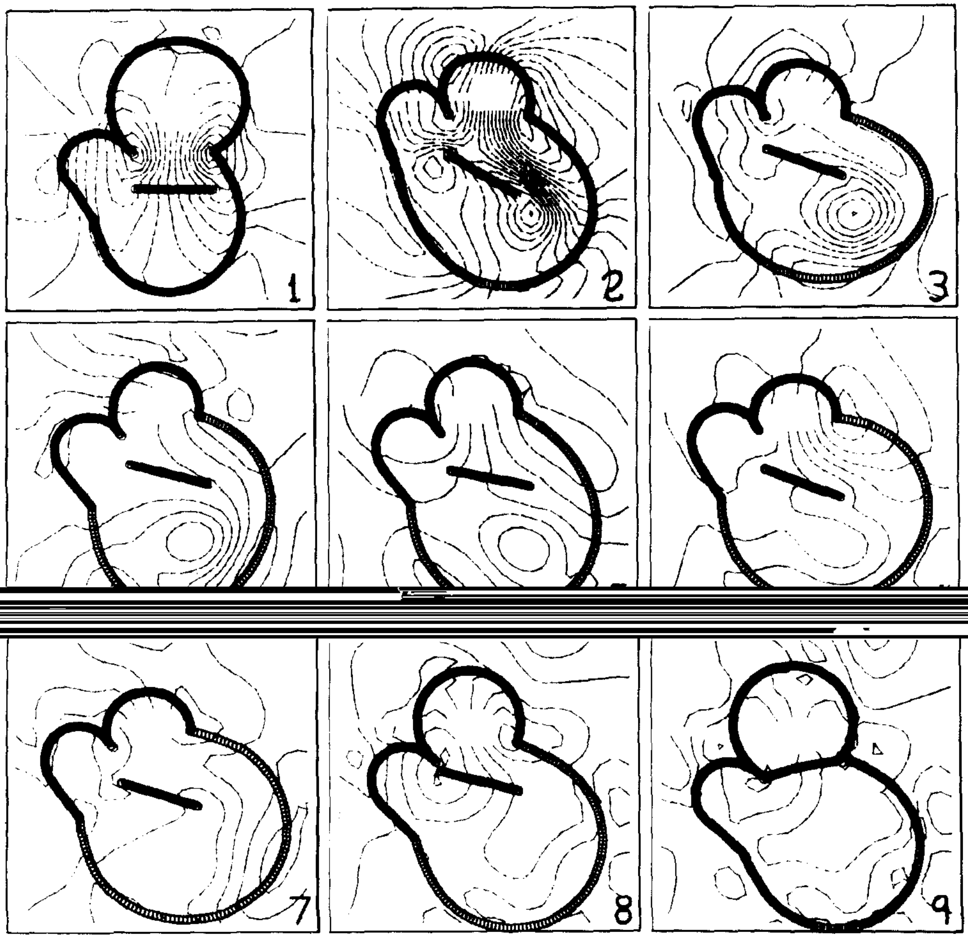

FIG. 9. Flow patterns of the eccentric monocusp (b). The computer model is shown in Fig. 2b. This is a different model of the same prosthetic valve as in Fig. 8; the results are nearly identical.

There are certain features of these streamline plots which require explanation. First, we have a source in the atrium so the region where $\nabla \cdot \mathbf{u} = 0$ is not simply connected and the stream function is not single-valued. We avoid this difficulty by introducing a cut which extends from the source in the middle of the atrium to the sink around the edges of the domain. We compute the stream function ψ from the velocity field by quadrature, and we choose the path of integration so that it never crosses the cut. Once the values of ψ have been computed on the mesh, contour lines corresponding to equally spaced values of ψ are plotted by interpolation. The

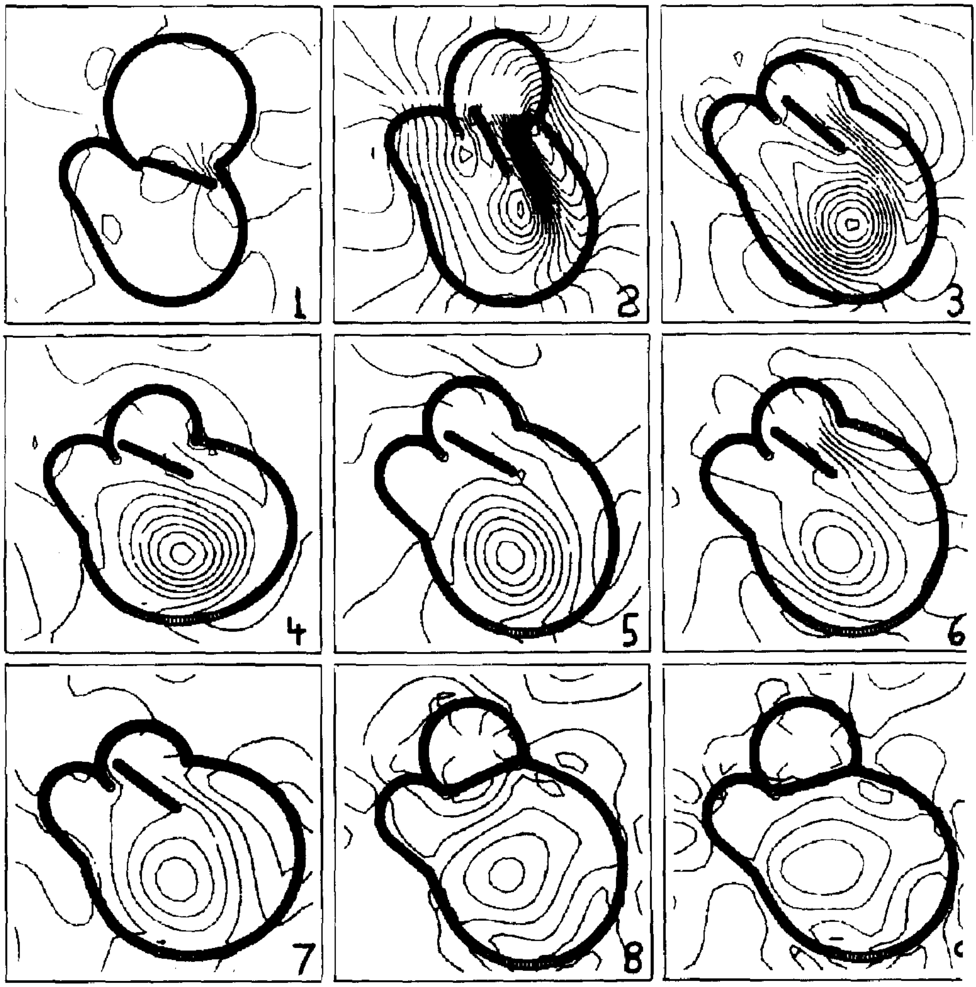

FIG. 10. Flow patterns of the pivoting disc (wide-angle). The computer model is shown in Fig. 3. Th dominant feature of the flow pattern is the prominent vortex which forms at the tip of the disc in fram (2). The pivoting disc behaves very much like the anterior leaflet of the natural valve. In this computa tion there is no constraint on the angle of opening, which is determined entirely by the fluid dynamic

program that computes ψ and constructs the contour lines was written by Antoinette Wolfe.

Another unusual aspect of these figures is that streamlines cross boundaries. This is, however, a simple consequence of the motion of our immersed boundaries, which move at the local fluid velocity [1]. It certainly does not mean that fluid particles cross boundaries. To check that they do not, we have run the following sensitive test. At $t = 0$, we put a fluid marker midway between each pair of adjacent boundary points. Like the boundary points, these markers were moved at the local fluid velocity using the same interpolation formula in both cases. Throughout the run, all of these

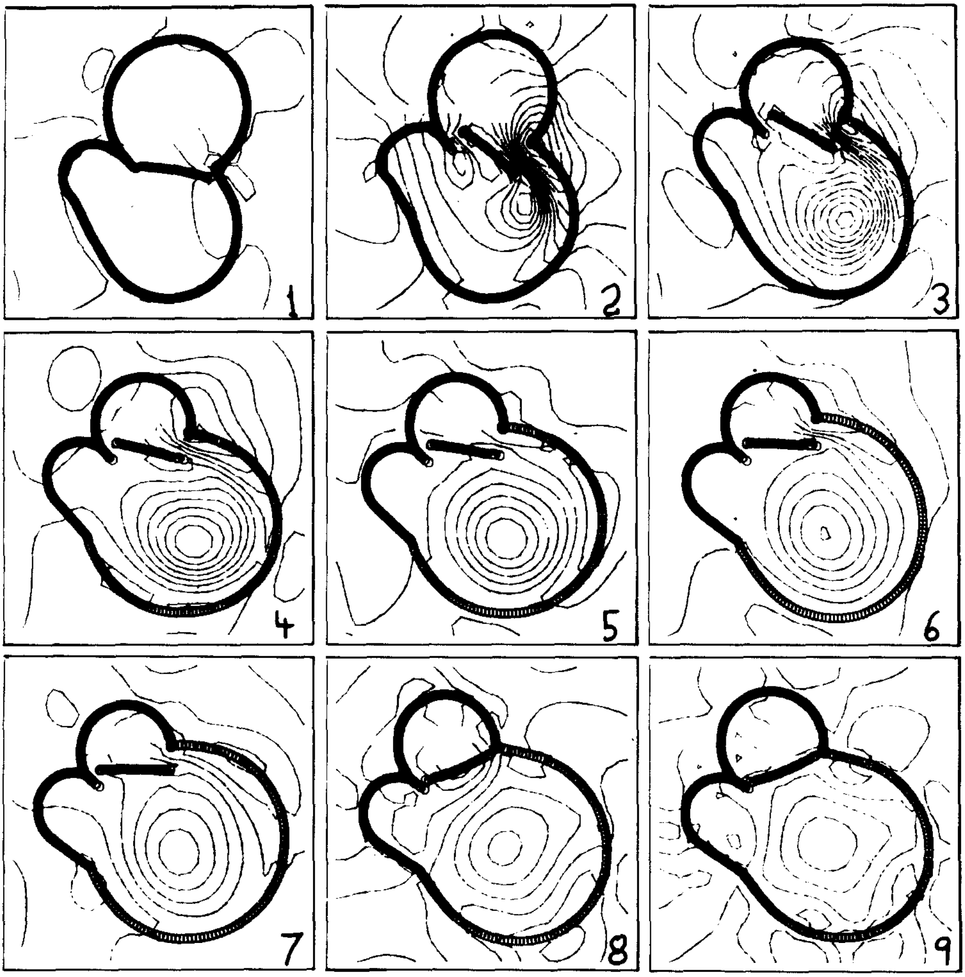

FIG. 11. Flow patterns of the pivoting disc (narrow-angle). The valve is the same as in Fig. 10 except that the position of the pivot point has been moved closer to the center of the disc. As before, there is no constraint on the angle of opening. Despite this, the maximum angle of opening is much less than in Fig. 10, and the space available for forward flow is very much reduced.

markers moved as though they were part of the boundary with the exception of two markers: the marker nearest the tip of each leaflet was carried away by the flow. This result confirms that we have placed the boundary points close enough together with respect to the fluid mesh.

The simple fact that streamlines cross moving boundaries is very important in understanding the flow patterns of blood in the heart. This remark is best illustrated by Fig. 7, which shows the streamlines of a caged ball, but it applies to all heart valves including the natural valve. When the ball valve is opening or closing, the streamlines pass through the moving ball. Accordingly the ball is not an obstacle to

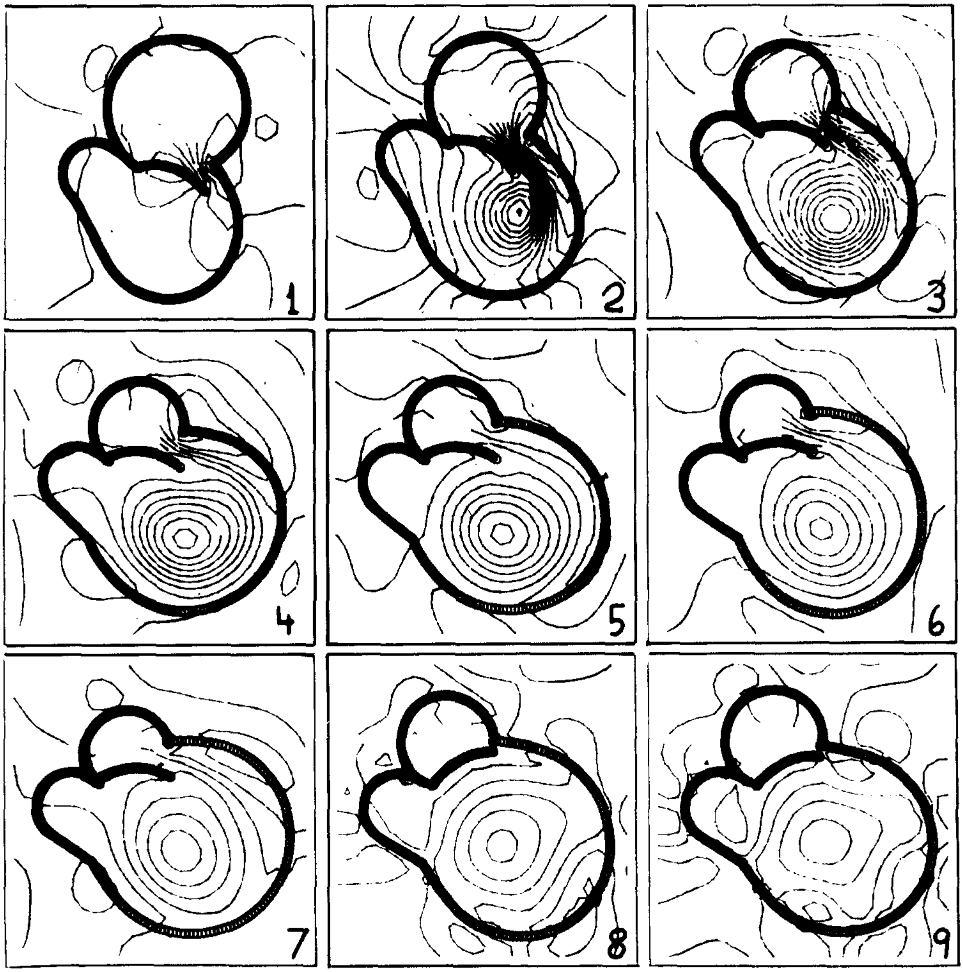

FIG. 12. Flow patterns of the curved pivoting disc. The computer model is shown in Fig. 4. As in the preceding two figures, the angle of opening is unconstrained. The valve is pivoted at the same point as in Fig. 10, but its maximum angle of opening is substantially smaller. This may be a consequence of lift on the curved valve.

flow at these times. It is only after the ball has reached its open or closed position that it acts as an obstacle. The flow patterns of valve opening and valve closure are very different from a sequence of steady flow patterns past the various intermediate positions of the valve. In a similar way, the streamlines of ventricular filling are very different from the streamlines of flow in a fixed cavity with the shape of the ventricle. In ventricular filling the streamlines intersect the moving wall. In a fixed cavity, they do not.

Figs. 8 and 9 show computed flow patterns for two different models of the same prosthetic valve. The two models are shown in Fig. 2. One model uses a double row of points with a triangular array of links to provide bending rigidity. The other model achieves the same result with a single row of points and three-point forces. The flow patterns and valve positions at corresponding times are practically identical, even with regard to small details. This is a convincing demonstration that constraints can be enforced in different ways with the same results.

The methods of this paper can be used for parametric studies of prosthetic valve design. We shall illustrate this with one example. Pivoting disc valves have a built-in constraint on the maximum angle of opening. It sometimes happens, however, that these valves fail to open to as wide an angle as the constraint would allow. This suggests that fluid dynamic effects can limit the angle of opening. To study this, we used a model valve with no constraint on the maximum angle of opening, and we systematically varied the position of the pivot point point. It might be thought that whenever the pivot is off-center the valve will open to an angle of 90°, but this is not the case. In fact, the angle of opening is always less than 90°, and it varies smoothly [redacted] valve opens widely enough that flow can go smoothly down both sides of the disc. In this configuration the disc appears to offer little resistance ot the forward flow. In Fig. 11, the pivot point is closer to the center of the disc. The angle of opening is substantially less, and the valve appears to be badly stenotic (high resistance) despite the absence of a mechanical constraint on the angle of opening.

Finally, in Fig. 12, we study a curved pivoting disc valve. The curvature may generate lift, as with an airfoil. This would reduce the angle of opening, but it could help in valve closure. The valve is pivoted at the same point as in Fig. 10 ("wide angle") but the maximum angle of opening is much less, and the flow patterns bear a greater resemblance to Fig. 11. A systematic study on the effect of curvature is planned.

6. Conclusion

We have overcome a fundamental difficulty in the application of [1] to prosthetic heart valves, and we have developed convenient techniques for modeling prosthetic mitral valves of any design so that their performance can be studied in our computer test chamber. In the process, we have substantially reduced the computer time required for each run.

Despite the limitations of the model, we have established reasonable agreement with experimental data for the case of the natural mitral valve. This makes us confident that the conditions of pressure and flow established by our computer test chamber are physiological and also that our natural valve results can be used as a standard of reference when we test prosthetic valves in the computer.

In the area of applications, we have begun a series of detailed parametric studies aimed at improvements in prosthetic valve design. We are also using the computational method to study the normal and pathological function of the natural mitral valve. In terms of development of the method, the greatest challenge is to remove the limitations of low Reynolds number and two space dimensions.

ACKNOWLEDGMENTS

We are indebted to Olof Widlund for the crucial suggestion that led to the present work, to Edward Yellin for experimental data and for many stimulating discussions, and to Antoinette Wolfe for the computer program that generates the streamline plots. This work was supported by the National Institutes of Health under Research Grant HL-17859. Computation was also supported in part by the Department of Energy under Contract EY-76-C-02-3077 at the Courant Mathematics and Computing Laboratory of New York University.

REFERENCES

1. C. S. PESKIN, *J. Comput. Phys.* **25** (1977), 220–252.
2. W. MURRAY, "Numerical Methods for Unconstrained Optimization," pp. 64–68, Academic Press, New York, 1972.
3. A. JENNINGS, *Comput. J.* **9** (1966), 1–23.
4. S. LANIADO, E. L. YELLIN, H. MILLER, AND R. W. M. FRATER, *Circulation* **47**(1973), 1006–1014.
5. M. F. MCCRACKEN AND C. S. PESKIN, *J. Comput. Phys.*, in press.
6. A. J. CHORIN, *Math. Comput.* **22**(1968), 745–762.
7. A. J. CHORIN, *Math. Comput.* **23** (1969), 341–353.
8. J. KÖHLER, *Proc. European Soc. Artificial Organs* **2** (1975), 33–35.