INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS, VOL. 17, 711–729 (1993)

THE NUMERICAL SOLUTION OF THE FLOW IN A GENERAL BIFURCATING CHANNEL AT MODERATELY HIGH REYNOLDS NUMBER USING BOUNDARY-FITTED CO-ORDINATES, PRIMITIVE VARIABLES AND NEWTON ITERATION

ROLAND HUNT

Department of Mathematics, University of Strathclyde, Glasgow, G64 1DS, U.K.

SUMMARY

A numerical code has been implemented for the numerical solution of the steady, incompressible Navier–Stokes equations using primitive variables in a bifurcating channel. A boundary-fitted, numerically generated grid is placed onto the domain of the channel which is transformed into either a rectilinear C- or T-shaped region. The differenced equations are solved using Newton's iteration which makes upwinding at high Reynolds number unnecessary. Practical implications of inverting the huge Jacobian matrix of Newton's method are discussed. The results have relative error of $2–3 \times 10^{-3}$ at Reynolds number 100, with T-geometry being marginally but significantly more accurate than C-geometry. Results have been obtained for Reynolds numbers up to 1000 for three bifurcations one of which models the carotid arterial bifurcation in the human head. For this latter bifurcation the wall shear stress is calculated in connection with the onset of atherosclerosis. Finally, the results of flows having different daughter tube end pressures are presented.

KEY WORDS Primitive variables Newton iteration Boundary-fitted co-ordinates

1. INTRODUCTION AND GOVERNING EQUATIONS

The numerical solution of the fluid flow in a bifurcating channel has attracted much attention in, for example, the simulation of blood flow in an arterial bifurcation (see Reference 1 for a review). Since the boundary is irregular much of this work employs either finite element of finite volume techniques. However, more recently, traditional finite difference techniques have come into prominence by using boundary fitted co-ordinates pioneered by Thompson.[2, 18] Here the irregular domain is mapped onto a rectangle, the ensuing grid being generated numerically. The governing equations are similarly transformed and are solved on the rectangular domain permitting central differences to be used throughout. The distinct advantage of this method is that the ensuing algebraic system is second-order-accurate. This has been successfully employed for the steady flow in a symmetric bifurcation using the stream function/vorticity representation of the fluid equations.[3] However, the general non-symmetric bifurcation considered in this paper (Figure 1) does not readily map onto a rectangle if it is required that we have small cells throughout the domain. The approach used here is to map the domain onto a rectilinear region which more naturally reflects the topology of the bifurcation. Two regions or geometries are considered, namely, a rectangle with a cut (used by Friedman and Ehrlich[4] in modelling the aortic

0271–2091/93/200711–19$14.50

Received August 1991
Revised May 1993

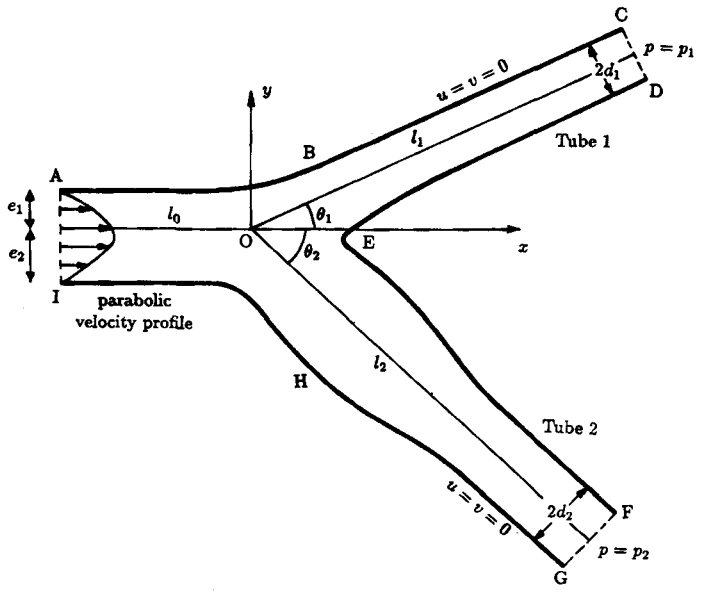

Figure 1. General bifurcation showing the boundary conditions. The curves ABC, DEF and GHI, the parameters $l_0, l_1, l_2, d_1, d_2, e_1, e_2, \theta_1, \theta_2, p_1$ and p_2 can be 'user specified'

bifurcation) and a T-shaped region which, as far as we are aware, is new (Figure 2). As we shall see, the latter is preferable having a higher concentration of cells near the apex.

The algebraic system of equations resulting from differencing the governing differential equations, is traditionally solved using relaxation techniques such as the Gauss–Seidel or SOR methods which can be accelerated using a multigrid formulation.[5,6] For such methods to converge at moderately high Reynolds number Re, which is typical for blood flow ($150 \leq Re \leq 600$),[1] it is usually necessary to incorporate an artificial viscosity or upwinding. The artificial viscosity proposed by Dennis and Hudson[7] is very attractive in that it is only applied locally and is second-order-accurate. However such an inclusion, whilst maintaining the nature of the flow, can increase the numerical error by a factor 10 or more.[8, 9] The advent of more powerful computers has made the more direct method of Newton's iteration feasible. The use of Newton's method for solving fluid flow equations was first considered by Fornberg,[10] who obtained results for the flow past a circular cylinder up to Reynolds number 600. The advantage of using Newton's method is not only that it converges quadratically, thus requiring only 3 or 4 iterations for convergence, but also, perhaps more importantly, does not need the use of artificial viscosity. The disadvantage of the method is that storage is required for the huge Jacobian matrix and that each iteration can use considerable CPU time. This is, however, outweighed by the extra accuracy available and the rapidity of the convergence. Usually, the solution obtained by Newton's method is accomplished on a supercomputer. However, such is the power of the modern PC and the capacity of its associated hard disk, that it has been possible to perform these calculations on such a machine using the disk to store the Jacobian.

Fluid calculations using Newton's method have so far only considered using the stream function/vorticity formulation of the governing equations; however, in this paper we chose to use the orginal, 'primitive' variables of velocity and pressure. This has two advantages, firstly, it is more accurate and, secondly, it is readily adaptable to a three-dimensional model. The big

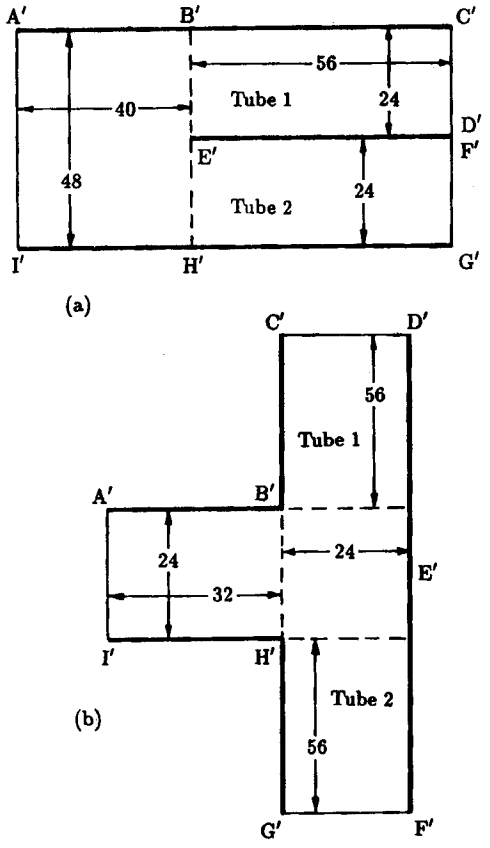

Figure 2. Rectilinear boundaries for (a) C-geometry and (b) T-geometry showing array dimensions used

disadvantage of the method is the difficulty of finding a suitable boundary condition for the pressure. We will overcome this difficulty by using a staggered grid (a modification of Harlow and Welch[11]) in which the pressure is not defined on the boundary.

The computer program is coded to solve the flow for general boundaries, i.e. the boundaries ABC, DEF and GHI in Figure 1 can be specified by the user. This is effected by mapping the irregular domain onto the rectilinear domain in Figure 2 using a numerical grid generator (Section 2.1). The transformed fluid equations are differenced using central differences and the resulting system of algebraic equations is solved using Newton's method (Section 2.2). Technical details for efficient implementation (Section 2.3) and an analysis of accuracy (Section 2.4) are also given. Results and discussion are given in Section 3.

2. NUMERICAL SOLUTION OF THE GOVERNING EQUATIONS

The governing equations are the non-dimensionalized steady Navier–Stokes equations:

$$\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}=0, \tag{1a}$$

$$u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}+\frac{\partial p}{\partial x}-\frac{1}{Re}\nabla^2 u=0, \tag{1b}$$

$$u\frac{\partial v}{\partial x}+v\frac{\partial v}{\partial y}+\frac{\partial p}{\partial y}-\frac{1}{Re}\nabla^2 v=0, \tag{1c}$$

where u, v are the velocities in the x, y directions, respectively, p is the pressure and $\nabla^2 \equiv \partial^2/\partial x^2+\partial^2/\partial y^2$ is the Laplacian. Equations (1) are to be solved in the domain A–I (Figure 1) which is specified by the curves ABC, DEF and GHI and the parameters $l_0, l_1, l_2, d_1, d_2, e_1, e_2, \theta_1$ and θ_2. The boundary conditions are also shown in Figure 1 and are

(i) the no-slip condition $u=v=0$ on the retaining walls,
(ii) a parabolic velocity profile on entry given by $y=6(x-e_1)(x+e_2)/(e_1+e_2)^2$ which gives a central height of 1·5 and an average velocity of unity,
(iii) the pressure is specified on exit, i.e. $p=p_1$ on tube 1 and $p=p_2$ on tube 2.

2.1. Grid generation

Before solving the governing equations, a numerically generated grid is placed on the domain in Figure 1. As mentioned in the Introduction, mapping the domain onto a rectangle is unsuitable if we are to have small cells throughout the domain. Hence, we will map the domain onto two alternative geometries, namely the split rectangle (C-geometry) and a T-shaped region (T-geometry) as shown in Figure 2. If (x, y) are the co-ordinates of the original domain and (ξ, η) of the transformed, then, following Nakamura,[17] an orthogonal grid can be generated by solving the equations

$$\begin{aligned} \mathrm{D}x&=(\alpha k_\xi-\beta k_\eta)x_\xi+(\beta k_\xi-\gamma k_\eta)x_\eta,\\ \mathrm{D}y&=(\alpha k_\xi-\beta k_\eta)y_\xi+(\beta k_\xi-\gamma k_\eta)y_\eta, \end{aligned} \tag{2}$$

where D is the quasi-linear elliptic partial differential operator given by

$$\mathrm{D}\equiv\alpha\frac{\partial^2}{\partial\xi^2}-2\beta\frac{\partial^2}{\partial\xi\partial_\eta}+\gamma\frac{\partial^2}{\partial\eta^2} \tag{3}$$

and

$$\alpha=x_\eta^2+y_\eta^2, \qquad \beta=x_\xi x_\eta+y_\xi y_\eta, \qquad \gamma=x_\xi^2+y_\xi^2. \tag{4}$$

Here k is an arbitrary function of (ξ, η) which is related to the aspect ratio of an individual cell and is used to control the local concentration of cells. These equations will give an orthogonal grid provided the grid lines meet the boundary curve at right angles, i.e. if

$$x_\xi x_\eta+y_\xi y_\eta=0 \tag{5}$$

on the boundary. Equations (2) are solved numerically by replacing derivatives by standard second-order central differences in (2)–(4) at each interior nodal points. The boundary condition (5) is differenced on a typical boundary (Figure 3(a)) about the point $(i, \frac{1}{2})$ using

$$\begin{aligned} z_\xi&=\tfrac{1}{4}(z_{i+1,1}+z_{i+1,0}-z_{i-1,1}-z_{i-1,0}),\\ z_\eta&=z_{i,1}-z_{i,0},\\ g(x_{i,0}, y_{i,0})&=0, \end{aligned} \tag{6}$$

where z is either x or y in equation (5) and $g(x, y)=0$ is the equation of the boundary. For convenience, we have chosen ξ and η such that the grid spacings in the transformed domain are

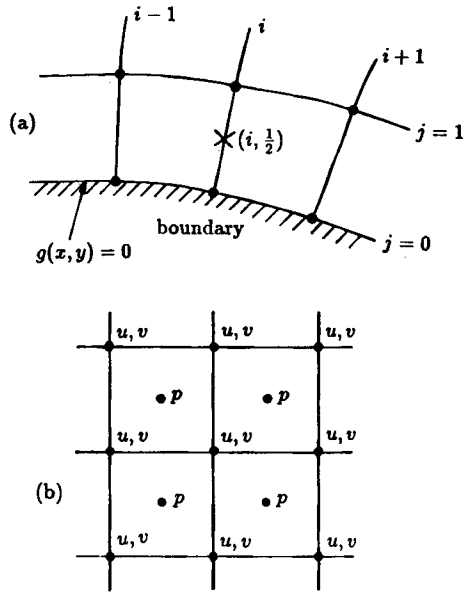

Figure 3. (a) Treatment of grid boundary. (b) Position of variables u, v and p relative to the grid

$\Delta\xi = \Delta\eta = 1$. Equations (6) can be adapted in a straightforward manner at internal corners E' in Figure 2(a) and B' and H' in Figure 2(b). Thus, associated with each nodal point (either on the boundary or in the interior), we have two non-linear equations. The full set of equations is solved iteratively using Newton's method (see Section 2.3).

We have generated 3 grids for T-geometry as shown in Figure 4(a)–4(c) and one for C-geometry (Figure 4(d)). Figure 4(a) and 4(d) are the same bifurcation having straight sides with curved corners and are used for comparison purposes. Figure 4(b) is similar except the corners are sharp and Figure 4(c) models the main carotid artery bifurcation.[12] Details of the grids including the function $k(\xi, \eta)$ are given in the appendix.

To solve equations (1) on a generated grid they are transformed to the (ξ, η) plane using the relationships

$$\frac{\partial}{\partial x} \equiv \frac{1}{J}\left(y_\eta \frac{\partial}{\partial \xi} - y_\xi \frac{\partial}{\partial \eta}\right), \qquad \frac{\partial}{\partial y} \equiv \frac{1}{J}\left(x_\xi \frac{\partial}{\partial \eta} - x_\eta \frac{\partial}{\partial \xi}\right),$$
$$\nabla^2 \equiv \frac{1}{J^2}\left(\alpha \frac{\partial^2}{\partial \xi^2} - 2\beta \frac{\partial^2}{\partial \xi \partial \eta} + \gamma \frac{\partial^2}{\partial \eta^2} + \sigma \frac{\partial}{\partial \eta} + \tau \frac{\partial}{\partial \xi}\right), \tag{7}$$

where

$$\sigma = \frac{1}{J}(y_\xi \mathrm{D}x - x_\xi \mathrm{D}_y), \qquad \tau = \frac{1}{J}(x_\eta \mathrm{D}y - y_\eta \mathrm{D}x), \tag{8}$$

with $\mathrm{D}, \alpha, \beta$ and γ as before and $J = x_\xi y_\eta - x_\eta y_\xi$. The resulting equations are long, with equations (1b) and (1c) comprising of over 100 terms each, and are therefore not given explicitly here. The size of these equations is considered later in connection with program accuracy.

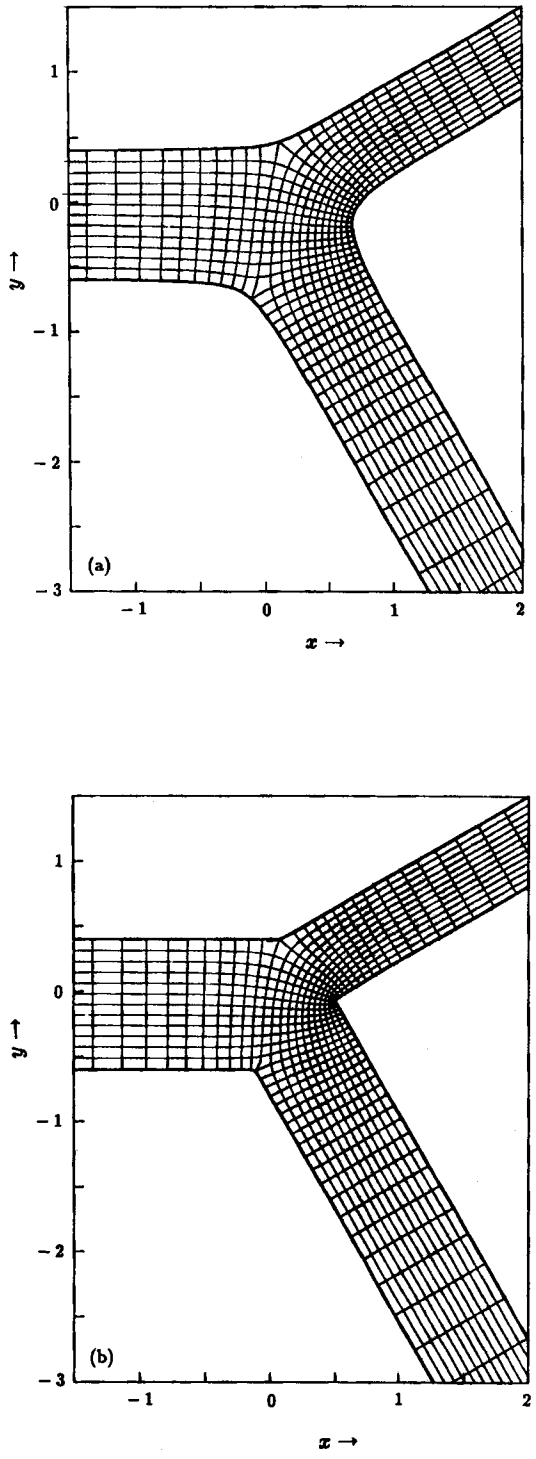

Figure 4. Generated grids: (a) Grid A with curved corners in T-geometry; (b) Grid B with sharp corners in T-geometry, (c) Grid C, the Bharadvaj bifurcation in T-geometry and (d) Grid D with curved corners in C-geometry

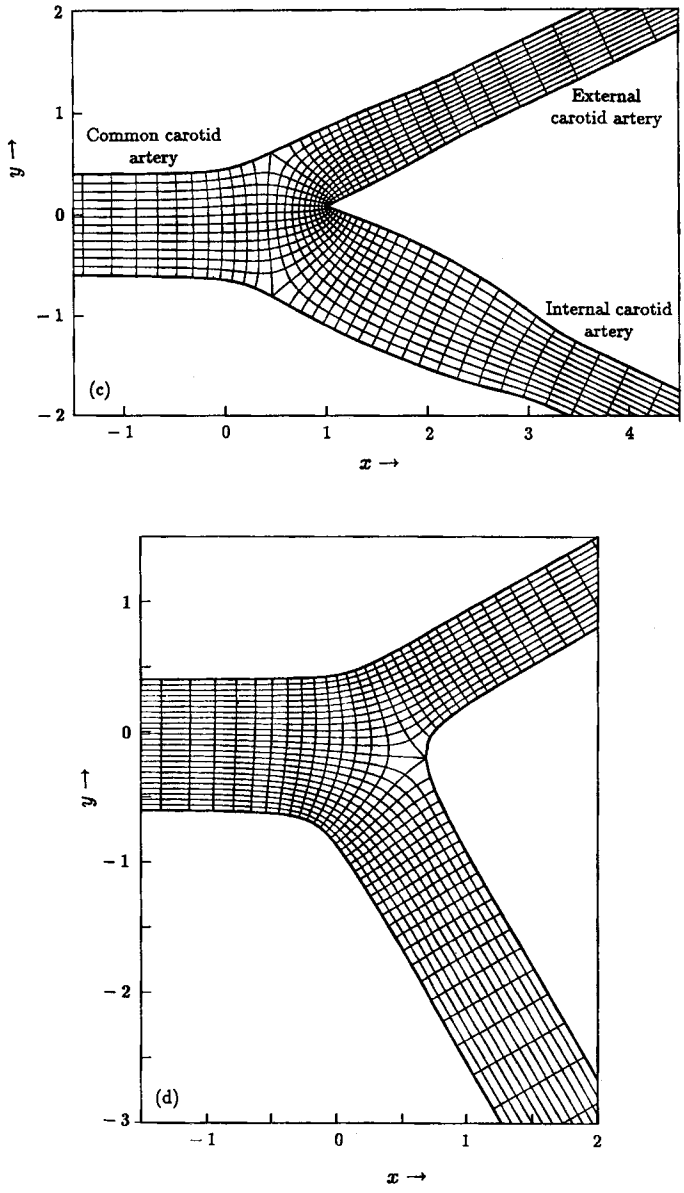

Figure 4. (Continued)

2.2. *Differencing and Newton's method*

Owing to the difficulty of providing a pressure boundary condition which could easily be applied, Harlow and Welch[11] devised a grid in which the velocity nodal points are situated at the sides of the cell with pressure at the centre. The pressure being defined at the centre of the cell, avoids the need to supply a pressure boundary condition. However, when equations (1) are mapped onto the (ξ, η) plane they are not readily discretized on such a staggered grid, and we use

a grid in which variables u, v are at cell vertices but retaining p at cell centres (see Figure 3(b)) as proposed by Kuznetsov.[13]

The transformed equation of (1a) is differenced about the point $(i+\frac{1}{2}, j+\frac{1}{2})$ using

$$\begin{aligned} z_\xi &= \tfrac{1}{2}(z_{i+1,j} + z_{i+1,j+1} - z_{i,j} - z_{i,j+1}), \\ z_\eta &= \tfrac{1}{2}(z_{i,j+1} + z_{i+1,j+1} - z_{i,j} - z_{i+1,j}), \end{aligned} \tag{9}$$

where z is either u or v. Equations (1b) and (1c) are differenced about (i, j) using the standard second-order central differences except for the pressure derivatives which use formulae similar to (8) shifted to centre on (i, j).

Specifying the velocity profile on entry of the main tube and pressure on exit of the daughter tubes are normally sufficient for a unique solution. However, the corresponding differenced equations have too few equations for the number of unknowns. In order to close the system we also specify the velocity on exit.[14] This is determined from the flow in the interior, by setting

$$\begin{aligned} &z_{N,j} = z_{N-1,j} \quad \text{(C-geometry)}, \\ &\left.\begin{aligned} z_{N,j} &= z_{N-1,j} \\ z_{0,j} &= z_{1,j} \end{aligned}\right\} \quad \text{(T-geometry)}, \end{aligned} \tag{10}$$

where z represents either u or v. The differenced equations and differenced boundary conditions now form closed set of non-linear algebraic equations. These can be represented as $\mathbf{f}(\mathbf{u}) = 0$ where $\mathbf{u}$ is the vector of unknowns and $\mathbf{f}$ is a vector containing the left-hand side of the equations. These equations are solved iteratively using Newton's method which is given by

$$\mathbf{u}^{(s+1)} = \mathbf{u}^{(s)} + \Delta\mathbf{u}^{(s)}, \qquad s = 0, 1, 2, \ldots, \tag{11}$$

where $\Delta\mathbf{u}^{(s)}$ is the solution of

$$A\,\Delta\mathbf{u}^{(s)} = -\mathbf{f}(\mathbf{u}^{(s)}), \tag{12}$$

and $A = \partial\mathbf{f}/\partial\mathbf{u}$ is the Jacobian matrix evaluated at $\mathbf{u} = \mathbf{u}^{(s)}$. The matrix A is a huge banded matrix with shape and dimensions as shown in Figure 5 for the two geometries. Inversion of (12) is effected by Gaussian elimination, the implementation of which is discussed in the next subsection. To commence the iteration, $\mathbf{u}^{(0)}$ must be specified. For high Reynolds number this initial iterate must be fairly accurate and accordingly results are obtained using continuation. For $Re = 10$ the initial iterate $\mathbf{u}^{(0)}$ is chosen to satisfy the boundary conditions with intermediate points set using

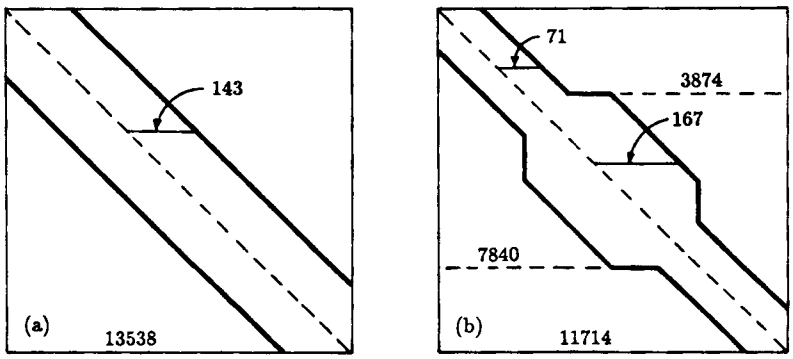

Figure 5. The shape and dimensions of the Jacobian used in Newton's iteration: (a) C-geometry and (b) T-geometry

linear interpolation. Then flows were obtained for $Re = 20, 50, 100, 200, 500$ and 1000 for which $\mathbf{u}^{(0)}$ is set to be the results from the previous value of Re. Typically, 4 iterations ensure an accuracy of 10^{-6}.

2.3. *Implementation*

These calculations were performed on a Nimbus VX386 microcomputer using the FTN77/386 Fortran compiler from the University of Salford. The matrix A is stored on hard disk and requires 9–15 Mbytes. The bulk of the computer time was taken up in solving equation (12). To solve this efficiently, the following were adopted:

(i) After a row of A is calculated, that row is immediately reduced using forward elimination and then transferred to the hard disk. This minimizes the number of disk transfers required.
(ii) Transfers to the hard disk are done in large batches.
(iii) The elimination process consists of three nested DO loops, the inner loop being the most expensive. This inner loop is only performed when it makes non-zero changes (i.e. only for non-zero multipliers) and then only for those elements which require a non-trivial change.

The adoption of these measures reduced the time taken by approximately a factor five. The computer code can be switched to run with or without partial pivoting. It was found that pivoting is unnecessary except when the pivotal element is zero thus saving CPU time. A typical iteration took between 1 and 2 h, and hence solutions were obtained overnight.

There are certain other matters worthy of note particular for those seeking to solve such problems by this method. These are listed below.

(i) For the results to be correct the vector $\mathbf{f}$ must be correct. The substitution of equations (7) and (8) into (1) results, as we have seen in equations (1b) and (1c), having the order of 100 terms. The discretization of these equations gives algebraic equations having several hundred terms. To avoid errors the program must be as transparent as possible. In particular, the equations used should look similar to their mathematical counterparts. For instance, if suitably named variables hold x_ξ, x_η, y_ξ and y_η then program statements for α, β and γ can be made to look like the formulas in (4). In the same way, program expressions for Dx and Dy in (3) and the equations to be solved (2) can be made similarly transparent. This greatly assists checking and therefore helps maintain program accuracy.
(ii) To obtain second-order convergence, which is the main virtue of Newton's method, every element of A must be correct. If one element is incorrect the convergence almost invariably reverts to first order. It is therefore useful to monitor the order achieved by the program as the calculation proceeds, which can be done as follows. Let E_s be the absolute value of the largest element in $\Delta\mathbf{u}^{(s)}$ which is a measure of the error in $\mathbf{u}^{(s)}$. If the convergence is of order α then

$$E_{s+1} \simeq kE_s^\alpha \quad \text{and} \quad E_s \simeq kE_{s-1}^\alpha \tag{13}$$

for some constant k. Eliminating k gives

$$\alpha \simeq \frac{\ln n(E_{s+1}/E_s)}{\ln n(E_s/E_{s-1})}. \tag{14}$$

For results in this paper, typically $1{\cdot}9 < \alpha < 2{\cdot}1$. Mistakes in A were usually revealed by an α value in the range $0{\cdot}99 < \alpha < 1{\cdot}01$.

(iii) Owing to the difficulty in obtaining A correctly, one might be tempted to find A dynamically, i.e. use

$$\frac{\partial f_i}{\partial u_j} \simeq \frac{f_i(u_0, \ldots, u_j+\varepsilon, \ldots, u_n) - f_i(u_0, \ldots, u_j, \ldots, u_n)}{\varepsilon}, \tag{15}$$

where f_i and u_j refer to the elements of $\mathbf{u}$ and $\mathbf{f}$, and ε is small parameter, for example, $\varepsilon = 10^{-6}$. This should be resisted since, not only is it costly in computer time, but it can affect the convergence. It was found that for large A and a grid which contains very small cells (e.g. near the apex of the bifurcation) the iteration failed to converge when A is calculated dynamically but converged readily when A is found algebraically. For similar reasons it is necessary to work to double precision if programming in Fortran. However, formula (15) can be used to check the correctness of A.

2.4. Accuracy

To estimate the accuracy we obtain a further set of results on a grid with half the number of cells in each direction. If, at a common location, the solution on the original grid is U_F and on the coarser grid is U_M, and if, further, U is the exact solution then for a second-order method we have

$$U - U_F \simeq K h^2 \quad \text{and} \quad U - U_M \simeq K(2h)^2 \tag{16}$$

for some constant K where h is a typical cell size. Eliminating U, the error on the fine grid (given by $\simeq K h^2$) at that location is approximately

$$\tfrac{1}{3}(U_F - U_M). \tag{17}$$

Here U is u or v for the velocities and the average of the four nearest neighbours for the pressure. Table I lists the maximum and average errors for a Reynolds number of 100 for both C- and T-geometry on grid A. It will be observed that average errors are in the range $2\text{–}4 \times 10^{-3}$. The results on the T-geometry are slightly but significantly more accurate than those on the C-geometry (50% at maximum and 20% on average). This is because there are more cells near the apex (E in Figure 1) in T-geometry than in C-geometry. For this reason the results in the next section use T-geometry.

Another useful indicator of program performance is to calculate the observed order of the method. To find this we need a further set of results from a grid having a quarter of the number of cells in each direction which we denote by U_C. If the order is r then similar to (16) we have

$$U - U_F \simeq K h^r, \qquad U - U_M \simeq K(2h)^r, \qquad U - U_C \simeq K(4h)^r. \tag{18}$$

Table I. Maximum and average relative errors (in units of 10^{-3}) and the perceived order of the method

	C-geometry			T-geometry		
	Max	Average	Order	Max	Average	Order
Horizontal velocity u	10·9	2·5	2·0	7·2	2·0	1·9
Vertical velocity v	14·0	2·9	1·7	7·4	1·7	2·1
Pressure p	10·6	7·3	3·2	6·7	2·5	0·9

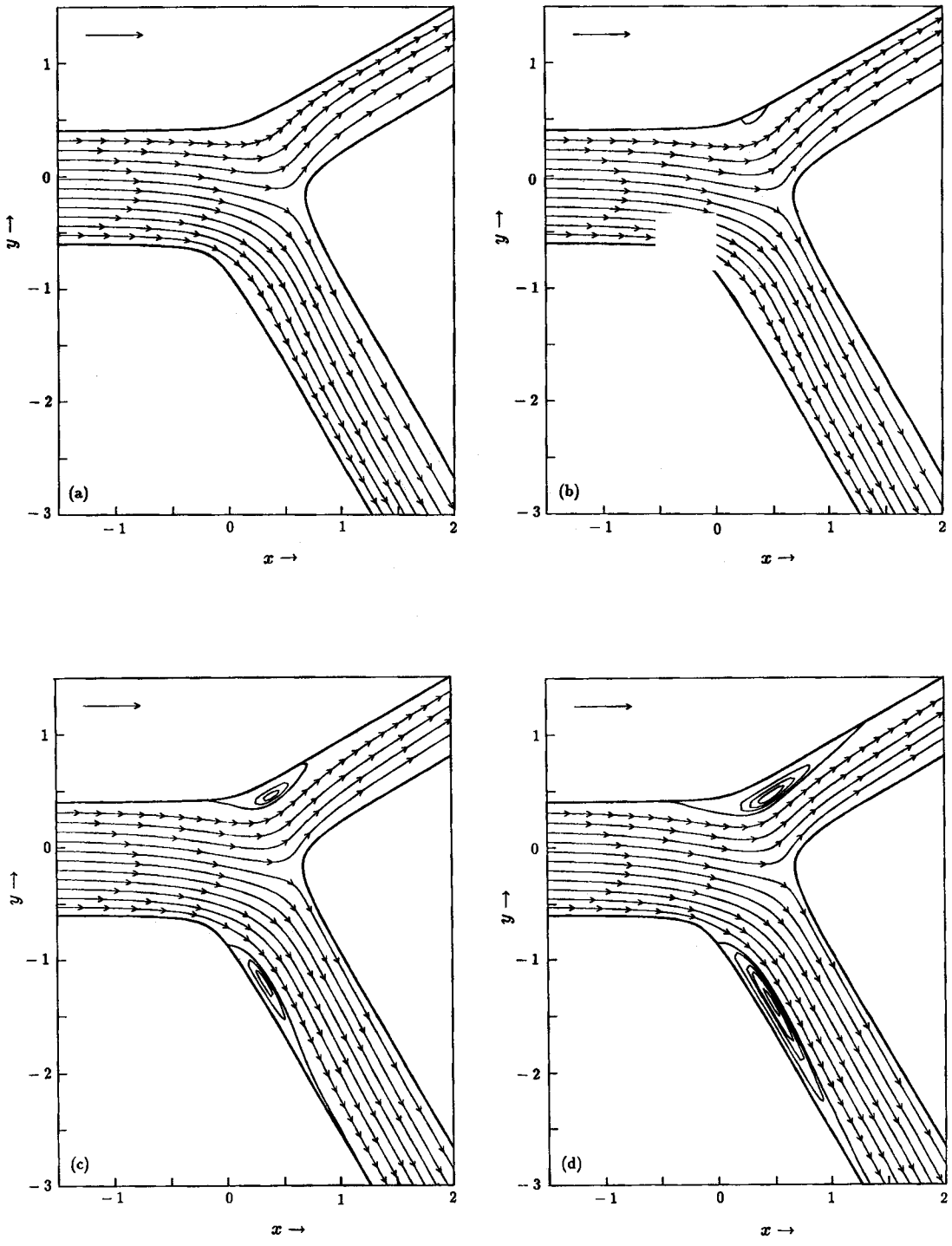

Figure 6. Streamlines for (a) $Re = 100$, (b) $Re = 200$, (c) $Re = 500$ and (d) $Re = 1000$, in bifurcation A. The distance between arrowheads is proportional to the velocity at the tail-end of the arrow. The arrow in the top left-hand corner corresponds to unit velocity. Arrowheads are not shown in the circulation regions for clarity since the velocities here are very small

Eliminating U and K give the empirical order as

$$r=\ln\left(\frac{U_F-U_M}{U_M-U_C}\right)\Big/\ln 2. \tag{19}$$

The average order is shown in Table I. For the two velocities we have good confirmation of the second-order nature of the results; however, the pressure is disappointing. It should however be appreciated that this is a very stiff test.

3. RESULTS

The results are depicted graphically from Figure 6 onwards. In each case T-geometry is used. Figure 6 shows the streamlines for the flow in the bifurcation A for $Re=100, 200, 500, 1000$. The distance between arrowheads is proportional to the velocity magnitude at the tailend of the arrow. As expected, the circulation regions develop and enlarge as Re increases, the approximate length and width of each region is shown in Table II. Figure 7 shows the corresponding pressure contours for $Re=100$, 500 in units of 0·05. As expected the maximum pressure occurs at the stagnation point near the apex and the pressure drop along the tubes is approximately inversely proportional to Re. Figure 8 shows the streamlines for the bifurcation B with sharp corners for $Re=100$. Unfortunately, the Newton iteration failed to converge for $Re>125$. This is almost certainly due to the presence of re-entrant angles at B, H and E at which the vorticity is singular, with point E being the most severe. To obtain solutions for Re significantly greater than 100 it would be necessary to use a much finer grid with a high concentration of cells near the corners.

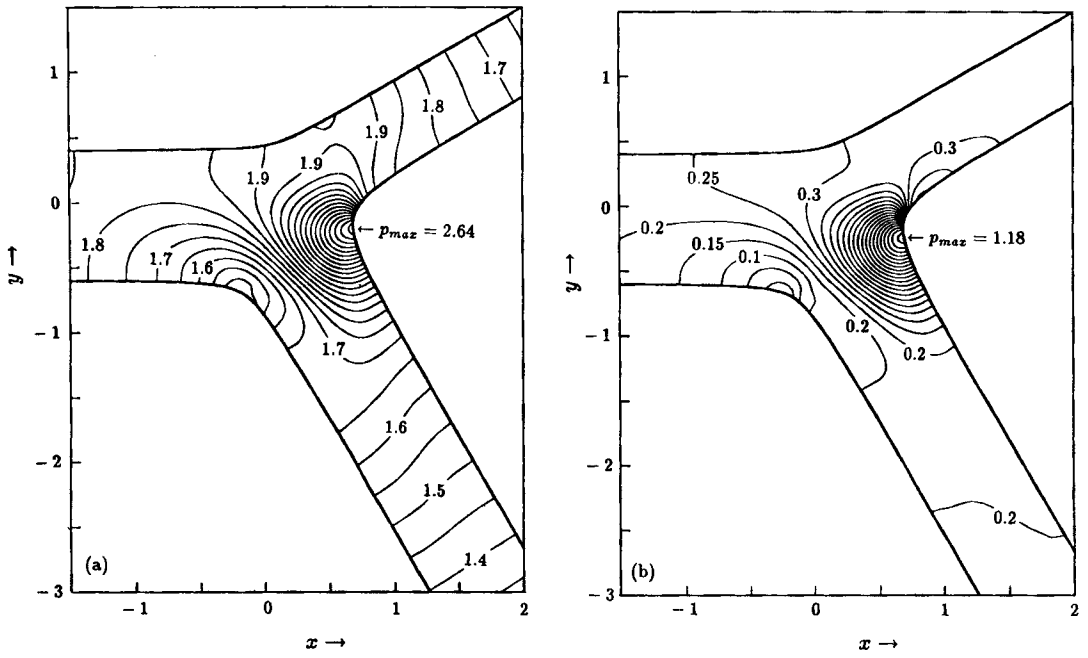

Figure 7. Pressure contours for (a) $Re=100$ and (b) $Re=500$ for bifurcation A. Contours are spaced in units of 0·05

Table II. Approximate sizes of circulation regions

		Upper		Lower	
	Re	Length	Width	Length	Width
Bifurcation A	200	0·31	0·08	0·25	0·01
	500	0·93	0·18	1·52	0·14
	1000	1·86	0·24	2·80	0·18
Bifurcation C	100	—	—	0·74	0·10
	200	0·16	0·03	1·84	0·31
	500	1·16	0·18	2·64	0·44

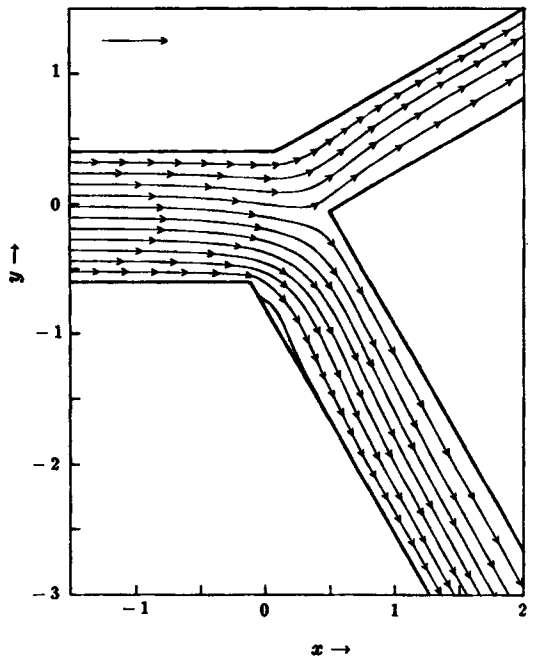

Figure 8. Streamlines for $Re = 100$ in bifurcation B

Figure 9 shows the streamlines of the flow in the Bharadvaj bifurcation C for $Re = 100, 200, 500$ and Figure 10 shows the corresponding pressure contours for the case $Re = 200$. As Re increases a large circulation region develops in the internal carotid artery with the consequence of slow flow and small wall shear stress. It is known that the onset of atherosclerosis is connected with arterial wall shear stress,[15, 16] but the connection is unclear at present with Fry predicting that high stresses will induce atherogenesis, whereas Caro and his co-workers predicting low stresses. Figure 11 shows the wall stress $\sigma_n = \partial u_t/\partial n$ where u_t is the tangential component of velocity near the wall and $\partial/\partial n$ is the normal derivative. From these we have high stresses near the apex and low stresses at the circulation regions and hence we expect bifurcations to be regions of likely

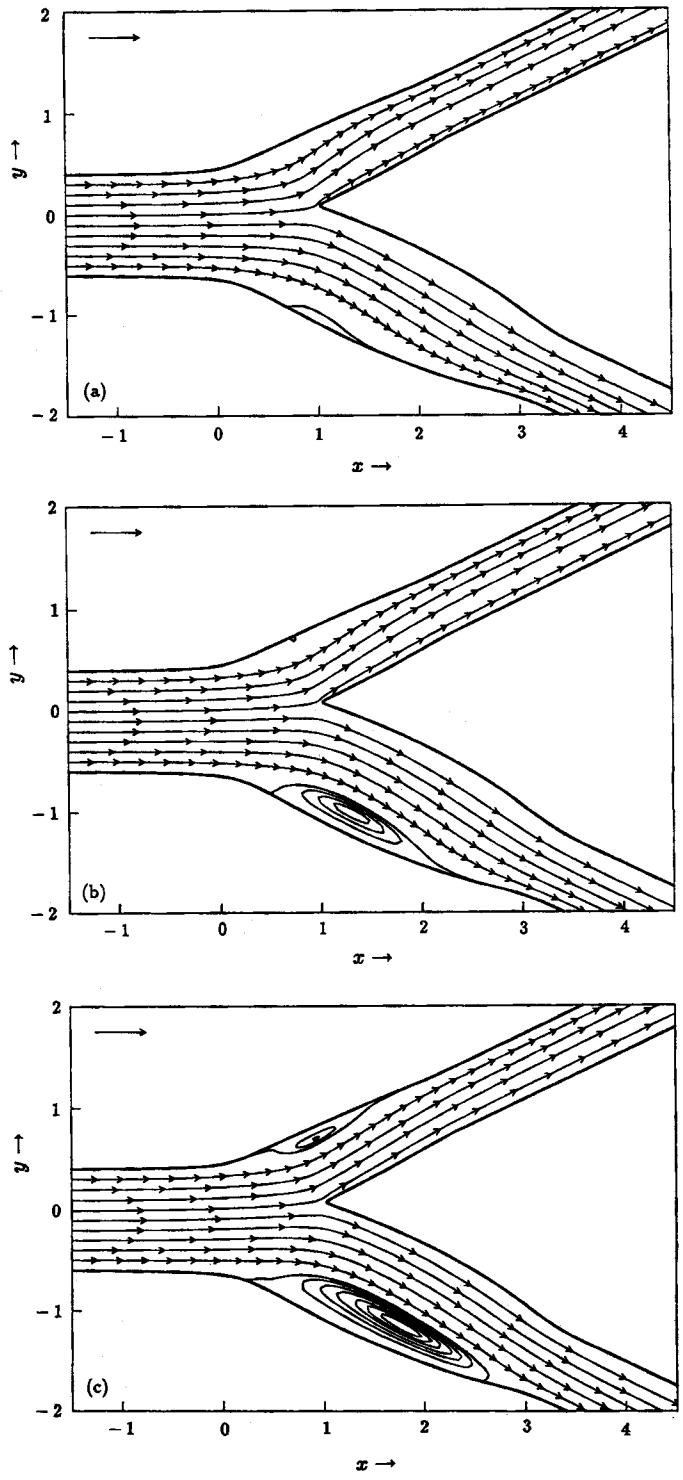

Figure 9. Streamlines for (a) $Re = 100$, (b) $Re = 200$ and (c) $Re = 500$ for the Bharadvaj bifurcation C

atherosclerosis development. The presence of spurious oscillations in Figure 11(c) for $x > 3$ along the curve GHI indicates that the outlet boundary condition in future models will need improving.

In all the results so far the pressures p_1 and p_2 at the ends of the daughter tubes have been the same. Figure 12 shows the flows at $Re = 100$ for the bifurcation A with $\Delta p \equiv p_2 - p_1 = -5, 5, 15$. For $\Delta p = -5$ and 15 the pressure is sufficient to reverse the flow in one of the daughter tubes. In the case $\Delta p = 5$ the flow is slowed in the lower tube producing a wide circulation region.

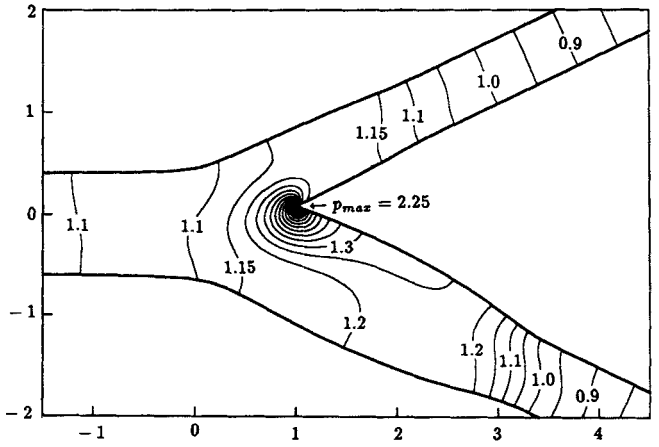

Figure 10. Pressure contous for $Re = 200$ in Bharadvaj bifurcation C. Contours are spaced in units of 0·05

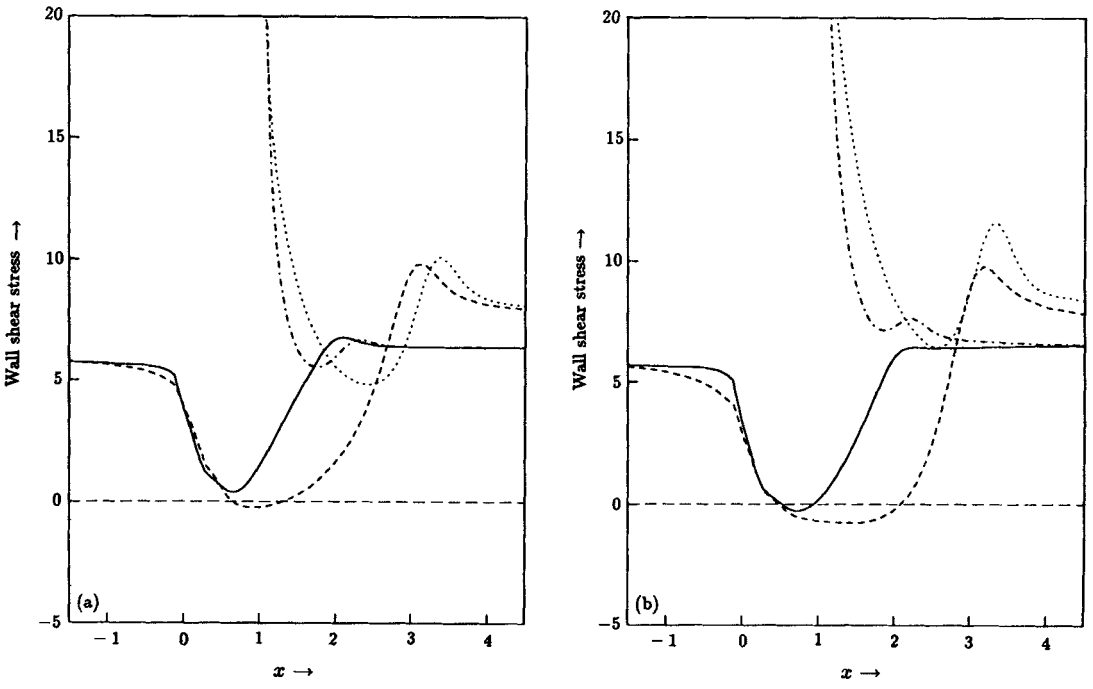

Figure 11. Wall shear stress in the Bharadvaj bifurcation C: (—) along curve ABC ; (– – –) along curve IHG, (– · – · –) along curve ED; (· · · · · ·) along curve EF for (a) $Re = 100$, (b) $Re = 200$ and (c) $Re = 500$

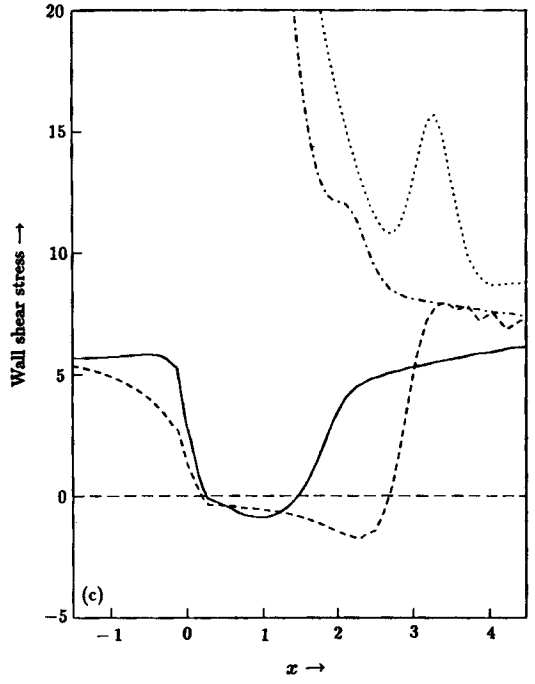

Figure 11. (Continued)

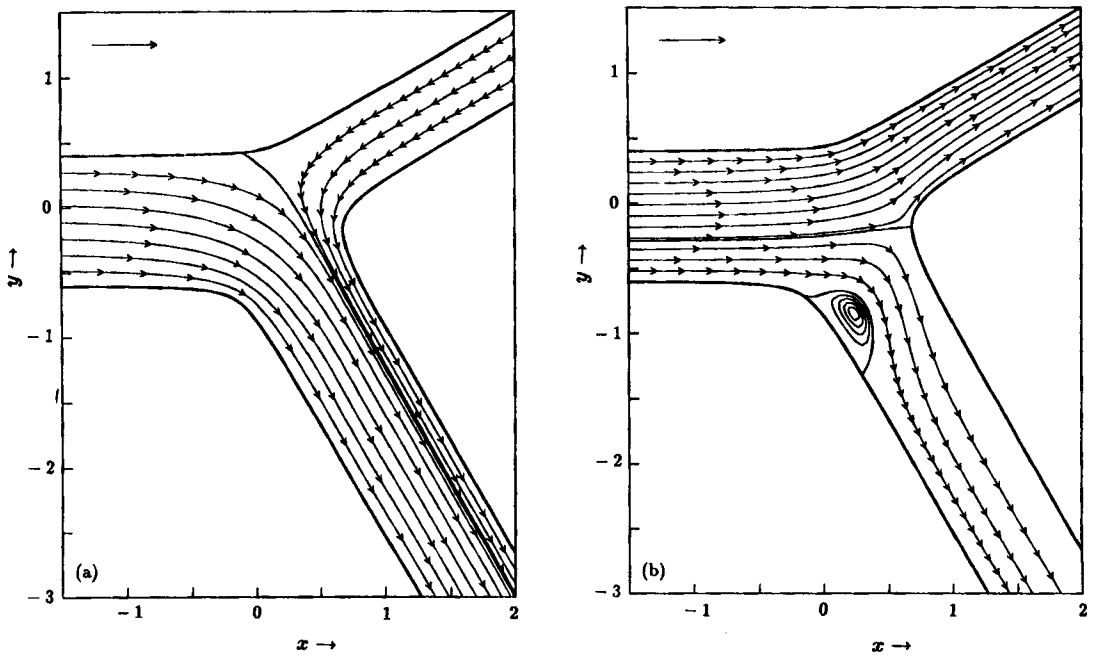

Figure 12. Streamlines for $Re = 100$ in bifurcation A with pressure difference $p_2 - p_1$ (a) -5, (b) 5 and (c) 15. The dividing streamline is shown in each case

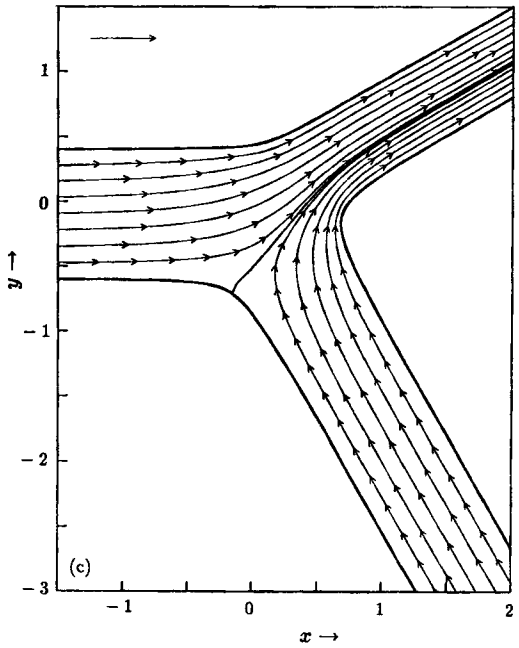

Figure 12. (Continued)

4. CONCLUSIONS

To conclude we summarize what we consider are the original elements used in these calculations.

(i) Mapping the physical domain onto a T-shaped region or grid using numerical grid generation. The ensuing grid is orthogonal having an even distribution of cells with a local concentration near the apex. Such a grid proved eminently suitable for the accurate calculation of the fluid flow through the bifurcation. The C-grid, also considered, produced results that were slightly but significantly less accurate.

(ii) The use of Newton's method with 'primitive' variables. This iteration makes the use of artificial viscosity unnecessary which substantially improves the accuracy of the results. We obtained results with relative accuracy of $2\text{–}3 \times 10^{-3}$ at moderately high Reynolds number for three non-symmetric bifurcations, one of which models the carotid arterial bifurcation.

(iii) The solution of the flow in a general non-symmetric bifurcation using finite differences.

(iv) The implementation of such a large calculation on a small computer.

This paper is very much a forerunner to future work. Of particular interest is the solution of the flow in a three-dimensional bifurcation since it is known experimentally that secondary velocity components which are necessarily absent in a two-dimensional model are not negligible.[1] Using 'primitive' variables instead of the more usual stream function/vorticity formulation means that the techniques discussed here can readily be adapted to a three-dimensional model. Also the code could be adapted to include time dependence for the study of pulsatile flow and the inclusion of a non-Newtonian fluid would model blood more precisely.

APPENDIX

The equations of the boundaries in Figure 1 are as follows:

$$\begin{aligned}
&\text{Curve ABC:} \quad \sinh[c(y-e_1)]\sinh[c(-x\sin\theta_1+y\cos\theta_1-d_1)]=h_1,\\
&\text{Curve GHI:} \quad \sinh[c(y+e_2)]\sinh[c(x\sin\theta_2+y\cos\theta_2+d_2)]=h_2,\\
&\text{Curve DEF:} \quad \sinh[c(-x\sin\theta_1+y\cos\theta_1+d_1]\sinh[c(x\sin\theta_2+y\cos\theta_2-d_2)]=h_3.
\end{aligned} \tag{20}$$

At large distances from the bifurcation the curve become virtually straight lines as given by setting the expressions in parantheses equal to zero. The sinh functions ensure that the curves rapidly become straight, away from the branch point. For Grid A,

$$\begin{aligned}
&h_1=0{\cdot}0243, \quad h_2=0{\cdot}0927, \quad h_3=0{\cdot}1911,\\
&\theta_1=30°, \quad \theta_2=60°, \quad l_0=4, \quad l_1=12 \quad l_2=12,\\
&e_1=0{\cdot}4, \quad e_2=0{\cdot}6, \quad c=3, \quad d_1=0{\cdot}3, \quad d_2=0{\cdot}4.
\end{aligned} \tag{21}$$

For Grid B the parameters are the same except $h_1=h_2=h_3=0$. The Bharadvaj bifurcation (Grid C) can be satisfactorily modelled using equations (20) by adding a suitable expression to represent the bulbous elements in the bifurcation. This is effected by replacing $d_j, j=1,2$ in (20) by

$$d_j+V_j\,\mathrm{sech}[z(1+0{\cdot}3z^4)],$$

with

$$z=(x\cos\theta_j-(-1)^j y\sin\theta_j-p_j)/w_j, \tag{22}$$

where V_j, w_j and p_j are the depth, width and centre of the bulbous portion, respectively. The variations parameters are given by

$$\begin{aligned}
&h_1=0{\cdot}0365, \quad h_2=0{\cdot}0927, \quad h_3=0{\cdot}0019,\\
&\theta_1=25{\cdot}1°, \quad \theta_2=25{\cdot}4°, \quad l_0=4, \quad l_1=12, \quad l_2=12,\\
&e_1=0{\cdot}4, \quad e_2=0{\cdot}6, \quad c=3, \quad d_1=0{\cdot}29 \quad d_2=0{\cdot}345,\\
&V_1=0{\cdot}055, \quad w_1=1{\cdot}00, \quad p_1=0{\cdot}964,\\
&V_2=0{\cdot}21, \quad w_2=1{\cdot}23, \quad p_2=1{\cdot}790.
\end{aligned} \tag{23}$$

Grids A, B and C all use T-geometry. Grid D is the same as Grid A except that it uses C-geometry.

The function $k(\xi,\eta)$ used to control the local density of grid cells is given by

$$\begin{aligned}
&k_\xi=\frac{1}{384}(\xi-4) \quad k_\eta=0 \quad \text{for C-geometry,}\\
&k_\xi=-\frac{1}{392}\xi \quad k_\eta=\frac{5}{3136}\eta \quad \text{for T-geometry.}
\end{aligned} \tag{24}$$

The lengths l_0, l_1 and l_2 are chosen to be sufficiently large so that the inlet and outlet conditions simulate uniform parabolic Poiseuille flow. It was found that setting $l_0=4$ and $l_1=l_2=12$ is sufficient for all the cases considered.

REFERENCES

1. F. N. Van de Vosse, 'Numerical analysis of carotid artery flow', *Ph.D. Thesis*, Eindhoven University, 1987.
2. J. F. Thompson, F. C. Thames and C. W. Mastin, 'Automatic numerical generation of body-fitted co-ordinates curvilinear co-ordinate system for field containing any number of arbitrary bodies', *J. Comput. Phys.*, **15**, 299–319 (1974).
3. J. S. Bramley and D. M. Sloan, 'Numerical solution for two-dimensional flow in a branching channel using boundary-fitted co-ordinates', *Comput. Fluids*, **15**, 297–311 (1987).
4. M. H. Friedman and L. W. Ehrlich, 'Numerical simulation of aortic bifurcation flows: the effects of flow divider curvature', *J. Biomech.*, **17**, 881–888 (1984).
5. G. Longdale, J. S. Bramley and D. M. Sloan, 'A nonlinear multigrid algorithm and boundary-fitted co-ordinates for the solution of a two-dimensional flow in a branching channel', *J. Comput. Phys.*, **78**, 1–14 (1988).
6. W. Hackbusch and U. Trottenberg, *Multigrid Methods*, Lecture Notes in Matehmatics, Vol. 960, Springer, Berlin, 1982.
7. S. C. R. Dennis and J. D. Hudson, 'A difference method for solving the Navier–Stokes equations', *Proc. 1st Conf. on Numerical Methods in Laminar and Turbulent Flow*, Pentech Press, London, 1978.
8. J. S. Bramley and D. M. Sloan, 'A comparison of an upwind difference scheme with a central difference scheme for moderate Reynolds number', *Internal Report No. 3*, Department of Mathematics, Strathclyde University, 1988.
9. R. Hunt, 'The numerical solution of the laminar flow in a constricted channel at moderately high Reynolds number using Newton iteration', *Int. j. numer. methods fluids*, **11**, 247–259 (1990).
10. B. Fornberg, 'Steady viscous flow past a circular cylinder up to Reynolds number 600', *J. Comput. Phys.*, **61**, 297–320 (1985).
11. F. H. Harlow and J. E. Welch, 'Numerical calculation of time-dependent viscous incompressible flow of fluid', *Phys. Fluids*, **8**, 2182–2189 (1965).
12. B. K. Bharadvaj, R. F. Mabon and D. P. Giddens, 'Steady flow in a model of the human carotid bifurcation', *J. Biomech.*, **15**, 363–378 (1982).
13. B. G. Kuznetsov, *Fluid Dyn. Trans.*, **4**, 85–89 (1968).
14. W. Y. Soh, 'Time-marching solution of incompressible Navier–Stokes equations for internal flow', *J. Comput. Phys.*, **70**, 232–252 (1987).
15. D. L. Fry 'Hemodynamic forces in atherosclerosis', in Scheinberg, P. (ed.), *Cerebrovascular Diseases*, Raven Press, New York, 1976, pp. 77–95.
16. C. G. Caro, J. M. Fitzgerald and R. C. Schoter, 'Atheroma and arterial wall shear. Observation, correlation and proposal of a shear mass transfer mechanism for atherogenesis', *Proc. Roy. Soc. Lond. B*, **177**, 109–159 (1971).
17. S. Nakamura, 'Generation of orthogonal grids by boundary grid relaxation', *NASA Contracter Report No. 166523.*
18. J. F. Thompson, Z. U. A. Warsi and C. Wayne, 'Boundary-fitted co-ordinates systems for numerical solution of partial differential equations—a review', *J. Comput. Phys.*, **47**, 1–108 (1982).