

## On Parallel Computation of Blood Flow in Human Arterial Network Based on 1-D Modelling

B. V. Rathish Kumar<sup>1,3</sup>, Alfio Quateroni<sup>1,2</sup>  
Luca Formaggia<sup>1</sup>, and Daniele Lamponi<sup>1</sup>

Received December 12, 2002; revised June 16, 2003  
Published online: December 1, 2003  
© Springer-Verlag 2003

### Abstract

In this study, parallel computation of blood flow in a 1-D model of human arterial network has been carried out employing a Taylor Galerkin Finite Element Method. Message passing interface libraries have been used on Origin 2000 SGI machine. A Greedy strategy for load-distribution has been devised and data-flow graphs necessary for parallelization have been generated. The performance of parallel implementation measured in terms of speedup and efficiency factors is found to be good. Further, the parallel code is used in simulating the propagation of pressure and velocity waveforms in our 1-D arterial model for two different inflow pressure pulses. Also, the influence of consideration of terminal resistance on pressure and velocity waveforms have been analyzed.

*Keywords:* Parallel computation, blood flow, arterial network, Taylor Galerkin method, 1-D model

### 1 Introduction

The human vascular system is a complex network comprising of heart and blood vessels of various dimensions. It functions in such a way as to supply each organ with the proper amount of blood, which may vary depending on physiological conditions and organ demands. In order to understand the normal and the pathological behavior of the system, a detailed knowledge of blood flow and the response of blood vessels is required. Computational simulations based on suitable mathematical models describing the hemo-dynamics in a blood vessel is one of the effective ways for providing such information. Numerical computation of pulsatile blood flow dynamics in the entire arterial network based on realistic three dimensional arterial geometries remains an impractical task despite the rapid progress in computer speed and the emergence of robust numerical techniques because of the complexity of the problem, which has been highlighted in [18]. To analyze the pathological behavior of blood vessels researchers [1–4] have employed local 3-D models based on either realistic local vessel geometries or conceptual models. Less detailed, yet meaningful information is obtained by the use of simplified one-dimensional models [5–7] to represent the human arterial tree and to approximately solve the flow and the pressure field. Recently, geometrical multi-scale models have been advocated. Lumped (zero dimensional) and

one dimensional models describing arterial network are coupled together with localized complex 3-D models [8, 9]. The adoption of a geometrical multi-scale approach is a possible answer to the need of having rather detailed information in a particular region of interest while maintaining a global description of the cardiovascular system. The applications of such techniques have been shown to provide useful information to the practitioners [7, 10] and in perspective such models could provide a suitable tool for patient-specific medical planning of interventions. However, the computational complexities of the different “building blocks” that make up a geometrical multi-scale simulation are very different. On a scalar computer, lumped parameter model solution over several heart beats are obtained in a matter of seconds. Complex arterial trees simulated by a network of 1-D model may require simulation times of the order of several minutes upto hours for very complex geometries. Three-dimensional simulations, instead, require computing times of several hours, even days on very refined grids. Load balancing between the models is thus an important aspect. While, there is not much gain in making the lumped parameter computations parallel, there is an interest in parallelizing 1-D computations (in view of their coupling with lumped parameter network) as well as the three-dimensional schemes. High performance computing based on the paradigm of parallel or distributed computing on a cluster of PCs or on desk-side shared memory systems based (S2MPs) appear to be a viable means of providing an efficient solution in real-time as required in the clinical practice. As a step towards achieving such a goal authors in this work focus on the parallelization of a code for the computations of a 1-D modeling of arterial network.

In this study a simplified model of human arterial tree consisting of 55 arteries has been considered. Flow in each of the arteries is approximated by using a one-dimensional model [6, 7]. A second order Taylor-Galerkin Finite Element scheme has been used to solve the hyperbolic partial differential equations describing the flow and pressure fields inside the human arterial system. In Taylor-Galerkin Finite Element method high order time-stepping schemes, derived by using Taylor series expansion in the time increment, are coupled with accurate spatial discretization provided by Galerkin Finite Element method. The resulting scheme has the desired properties of accuracy and extended stability to cope up with the dominant role played by the characteristics in convection-dominated hyperbolic problems. It successfully precludes the so called spurious numerical oscillations which are usually encountered while applying low-order time stepping schemes in combination with the conventional Galerkin method to convection-dominated flows.

A Parallel algorithm based on the paradigm of domain decomposition has been developed and all the computations have been carried out on 8-node Origin 2000 machine using message passing interface libraries (MPI) under IRIX environment. A load-distribution strategy based on Greedy approach has been devised and is successfully used in distributing the computational load on various nodes of the parallel computer. The paper is organized in the following manner: (a) In section 2 a brief description of the one-dimensional model is given, (b) In section 3 numerical discretization of the one-dimensional model and the sequential algorithm for solving the problem is discussed, (c) In section 4 the parallelization

details including those of hardware are provided, (d) Section 5 provides the details of parallel computation in terms of speedup and efficiency plots. Also some of the results pertaining to the flow in the arterial tree are discussed.

### 2 Mathematical Model

The simplified human arterial tree with 55 main arteries adopted in this work is shown in Figure 1 and the data related to the arteries is provided in Table 1. Consider a simple compliant tube of length  $L$  as shown in Figure 2, as a model of an arterial segment of the human arterial tree. Following Brook et al. [11], Quarteroni and

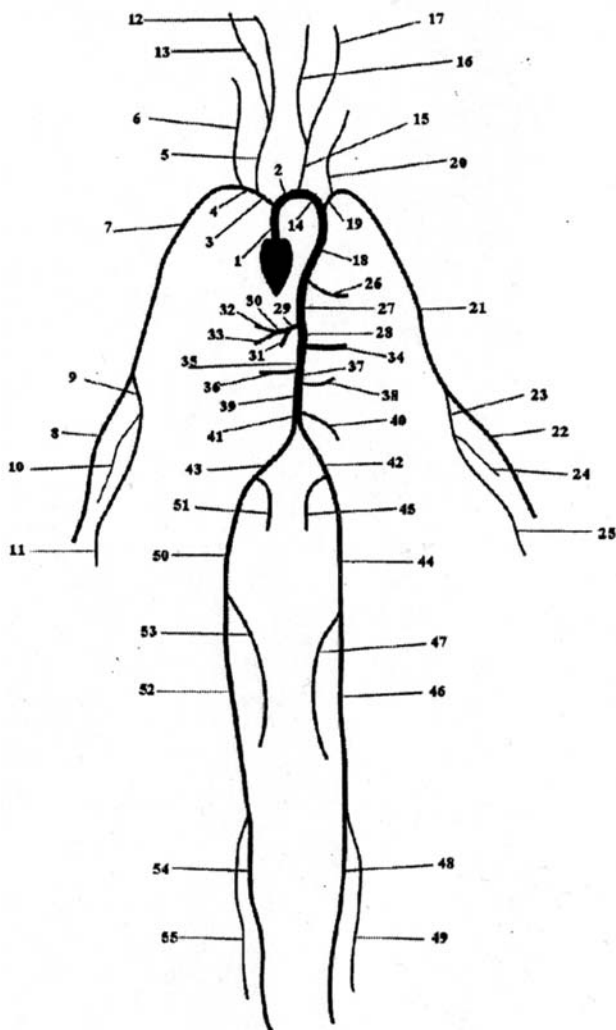


Fig. 1. Simplified human arterial tree with 55 arteries

**Table 1.** Data related to the simplified arterial tree with 55 main arteries

#	Artery	Length (cm)	Area (cm <sup>2</sup> )	$\beta$ (kg s <sup>-2</sup> cm <sup>-2</sup> )	$R_t$
1	Ascending Aorta	4.0	5.983	97	–
2	Aortic Arch I	2.0	5.147	87	–
3	Brachiocephalic	3.4	1.219	233	–
4	R. Subclavian I	3.4	0.562	423	–
5	R. Carotid	17.7	0.432	516	–
6	R. Vertebral	14.8	0.123	2590	0.906
7	R. Subclavian II	42.2	0.510	466	–
8	R. Radial	23.5	0.106	2866	0.82
9	R. Ulnar I	6.7	0.145	2246	–
10	R. Interosseous	7.9	0.031	12894	0.956
11	R. Ulnar II	17.1	0.133	2446	0.893
12	R. Internal Carotid	17.6	0.121	2644	0.784
13	R. External Carotid	17.7	0.121	2467	0.79
14	Aortic Arch II	3.9	3.142	130	–
15	L. Carotid	20.8	0.430	519	–
16	L. Internal Carotid	17.6	0.121	2644	0.784
17	L. External Carotid	17.7	0.121	2467	0.791
18	Thoracic Aorta I	5.2	3.142	124	–
19	L. Subclavian I	3.4	0.562	416	–
20	Vertebral	14.8	0.123	2590	0.906
21	L. Subclavian II	42.2	0.510	466	–
22	L. Radial	23.5	0.106	2866	0.821
23	L. Ulnar I	6.7	0.145	2246	–
24	L. Interosseous	7.9	0.031	12894	0.956
25	L. Ulnar II	17.1	0.133	2446	0.893
26	Intercostals	8.0	0.196	885	0.627
27	Thoracic Aorta II	10.4	3.017	117	–
28	Abdominal I	5.3	1.911	167	–
29	Celiac I	2.0	0.478	475	–
30	Celiac II	1.0	0.126	1805	–
31	Hepatic	6.6	0.152	1142	0.925
32	Gastric	7.1	0.102	1567	0.921
33	Splenic	6.3	0.238	806	0.93
34	Superior Mesenteric	5.9	0.430	569	0.934
35	Abdominal II	1.0	1.247	227	–
36	L. Renal	3.2	0.332	566	0.861
37	Abdominal III	1.0	1.021	278	–
38	R. Renal	3.2	0.159	1181	0.861
39	Abdominal IV	10.6	0.697	381	–
40	Inferior Mesenteric	5.0	0.080	1895	0.918
41	Abdominal V	1.0	0.578	399	–
42	R. Common Iliac	5.9	0.328	649	–
43	L. Common Iliac	5.8	0.328	649	–
44	L. External Iliac	14.4	0.252	1493	–
45	L. Internal Iliac	5.0	0.181	3134	0.925
46	L. Femoral	44.3	0.139	2559	–
47	L. Deep Femoral	12.6	0.126	2652	0.885
48	L. Posterior Tibial	32.1	0.110	5808	0.724
49	L. Anterior Tibial	34.3	0.060	9243	0.716
50	R. External Iliac	14.5	0.252	1493	–
51	R. Internal Iliac	5.1	0.181	3134	0.925
52	R. Femoral	44.4	0.139	2559	–
53	R. Deep Femoral	12.7	0.126	2652	0.888
54	L. Posterior Tibial	32.2	0.110	5808	0.724
55	R. Anterior Tibial	34.4	0.060	9243	0.716

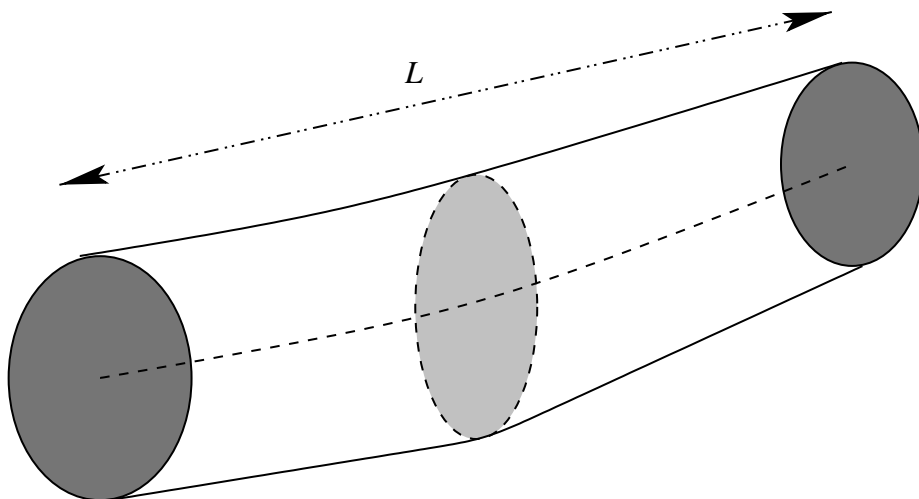


Fig. 2. A simple compliant tube of length  $L$

Formaggia [10] and Sherwin et al. [7] we write the system of equations representing conservation of mass and momentum in arterial segment  $[0, L]$  as:

$$\frac{\partial A}{\partial t} + \frac{\partial Q}{\partial z} = 0 \tag{1}$$

$$\frac{\partial Q}{\partial t} + \frac{\partial}{\partial z} \left( \alpha \frac{Q^2}{A} \right) + \frac{A}{\rho} \frac{\partial p}{\partial z} + K_R \frac{Q}{A} = 0 \tag{2}$$

where  $z$  is the axial direction,  $A = A(z, t) = \int_{S(z,t)} d\sigma$  is the area of the cross section  $S = S(z, t)$ ,  $Q = Q(z, t)$  is the mass flux across the section,  $\rho$  is the density of the blood which is taken to be a constant,  $p = p(z, t)$  is the internal pressure and  $u(z, t) = \frac{Q(z,t)}{A(z,t)}$  denotes the velocity of the fluid averaged across the section. The term  $\alpha$  is the momentum flux correction coefficient, defined as  $\alpha(z, t) = \frac{A}{Q^2} \int u^2 d\sigma$  and is assumed to be a constant. This implies that the velocity  $u$  is taken to be proportional to a given profile on each section  $S$ . A constant velocity profile would give  $\alpha = 1$ . Details may be found in [10].

In this study we will assume  $\alpha = 1$ . Experimental data [6, 7] have shown that blood velocity profiles are rather flat on average. Hence the crude assumption that  $\alpha = 1$  is reasonable and it further simplifies the analysis [7]. The term  $K_R$  is a strictly positive quantity, which represents the viscous resistance of the flow per unit length of the tube. There are three unknowns ( $p, A$  and  $Q$ ) in the above system (1-2) of two equations. To close the system the following static equilibrium relation between the pressure ( $p$ ) of the vessel and the vessel area ( $A$ ) is assumed:

$$p = p_{ext} + \frac{\beta(A - \sqrt{A_0})}{A_0} \quad (3)$$

where  $\beta = Eh_0\sqrt{\pi}$ .

Here  $h_0$  and  $A_0 = A_0(z)$  denote the reference vessel thickness and sectional area respectively,  $E = E(z)$  is the Young's modulus,  $p_{ext}$  is the external pressure. The reference state is taken to be that of the vessel at rest, i.e.  $Q = 0$ , and in equilibrium with the external pressure, i.e.  $p = p_{ext}$ .

By replacing  $p$  in (2) using (3) we obtain a system of differential equations, which may be written in conservation form as:

$$\frac{\partial U}{\partial t} + \frac{\partial F(U)}{\partial z} = B(U) \quad (4)$$

where  $U = [A, Q]^T$  are the conservative variables,  $F = [F_A, F_Q]$  the corresponding fluxes and  $B = [B_A, B_Q]^T$  a source term. These quantities are given by:

$$F(v) = \left[ \alpha \frac{Q^2}{A_0} + \frac{\beta}{3\rho A_0} A^{3/2} \right] \quad (5)$$

$$B(U) = \left[ K_R \frac{Q}{A} + \frac{A}{A_0} \frac{h_0 \sqrt{\pi}}{\rho} \left( \frac{2}{3} A^{1/2} - A_0^{1/2} \right) \frac{\partial E}{\partial z} - \frac{E h_0 \sqrt{\pi} A}{\rho A_0} \left( \frac{2}{3} A^{1/2} - \frac{1}{2} A_0^{1/2} \right) \frac{\partial A_0}{\partial z} \right] \quad (6)$$

Equation (4) may be written in quasi-linear form as:

$$\frac{\partial U}{\partial t} + H \frac{\partial U}{\partial z} = B(U) - \frac{\partial F}{\partial A_0} \frac{\partial A_0}{\partial z} - \frac{\partial F}{\partial \beta} \frac{\partial \beta}{\partial z} \quad (7)$$

where,

$$H(U) = \frac{\partial F}{\partial U} = \begin{bmatrix} \frac{\partial F_1}{\partial A} & \frac{\partial F_1}{\partial Q} \\ \frac{\partial F_2}{\partial A} & \frac{\partial F_2}{\partial Q} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\alpha \frac{Q^2}{A^2} + \frac{\beta}{2\rho A_0} A^{1/2} & 2\alpha \frac{Q}{A} \end{bmatrix} \quad (8)$$

For all allowable values of  $U$  (i.e.  $A > 0$ ) the matrix  $H$  has two real and distinct eigenvalues [10],

$$\lambda_{1,2} = \alpha \frac{Q}{A} \pm \sqrt{c_1^2 + \alpha(\alpha - 1) \frac{Q^2}{A^2}}, \quad c_1 = \sqrt{\frac{\beta}{2\rho A_0}} A^{1/4} \quad (9)$$

The system is hyperbolic. When  $\alpha = 1$ , the characteristic variables  $W(U)$  are given by:

$$W_{1,2} = \alpha \frac{Q}{A} \pm 4 \sqrt{\frac{\beta}{2\rho A_0}} A^{1/4} \quad (10)$$

From (10) one can get

$$A = \left( \frac{2\rho A_0}{\beta} \right)^2 \frac{(W_1 - W_2)^4}{8^4}, \quad Q = A \frac{W_1 + W_2}{2} \quad (11)$$

In this model approximately the presence of a prosthesis or a stent may be modeled by changing the value of the parameter  $\beta$ . Stenosis may be simulated by approximately choosing  $A_0(z)$  in the stenotic artery.

### Boundary and Interface Conditions:

The arterial network will be handled by a domain decomposition approach, where each arterial element is modeled by using the formulation just described. Clearly, the differential system (4) must be augmented by proper conditions at the boundaries of the arterial network and at the interface between adjacent arterial elements. Consequently, in our simplified model of human arterial system one would encounter the following four types of situations, with respect to the inflow direction:

1. Artery with an inflow condition at its proximal end and a connection to another artery at its distal end.
2. Artery which is connected to other arteries both at its proximal and distal ends.
3. Artery which is connected to another artery at its proximal end and bifurcates into two other arteries at its distal end.
4. Artery which is connected to another artery at its proximal end and has a terminal distal end.

For introducing the boundary conditions considered in this study we would first discuss about characteristic extrapolation and about the boundary condition at the interface of two (three) arteries. As the two eigenvalues ( $\lambda_1, \lambda_2$ ) for the flow regimes in the human arterial tree are always of opposite sign, the differential problem associated with each arterial segment needs exactly one boundary condition at either ends of a segment. Important classes of boundary conditions, called non-reflecting conditions, are those that allow the simple wave solution associated to the outgoing characteristic to leave the domain. Following [12, 13] they may be written as:

$$L_1^T \left( \frac{\partial U}{\partial t} - B(U) \right)_{z=a} = 0, \quad L_2^T \left( \frac{\partial U}{\partial t} - B(U) \right)_{z=b} = 0 \quad (12)$$

where  $z = a, b$  denote either ends of an artery,  $L_1, L_2$  are the left eigen vector associated to  $\lambda_1$  and  $\lambda_2$  respectively. A boundary condition of this type is quite convenient at the distal section of terminal arteries. However, if we want to account for the terminal resistance caused by the smallest arterioles and the capillary bed, those conditions should be replaced, for instance, by the technique proposed by Sherwin et al. [7].

Although the differential problem requires only one boundary condition at either ends of an artery, the solution of the numerical problem calls for a full set of

values for  $A$  and  $Q$  at the ends of the computational grid. So in addition to the physical boundary conditions we need two extra compatibility conditions [16] to meaningfully define the values of  $A$  and  $Q$  at either ends. In view of the fact that  $A$  and  $Q$  can be expressed in terms of characteristic variables  $W_1, W_2$  the required compatibility conditions are here implemented through the so-called “characteristic extrapolation method” [14]. The idea is based on the well known fact that the characteristic variables satisfy a system of ODEs along the characteristic path. Indeed when  $B(U) = 0$  the characteristic variables are constant along the characteristics, so a first order approximation of the exiting characteristic variables at time  $t^{n+1}$  at proximal end ( $z = a$ ) and distal end ( $z = b$ ) is provided by:

$$W_2^{n+1}(z_a) = W_2^n(-\lambda_2^n(z_a)\Delta t) \tag{13}$$

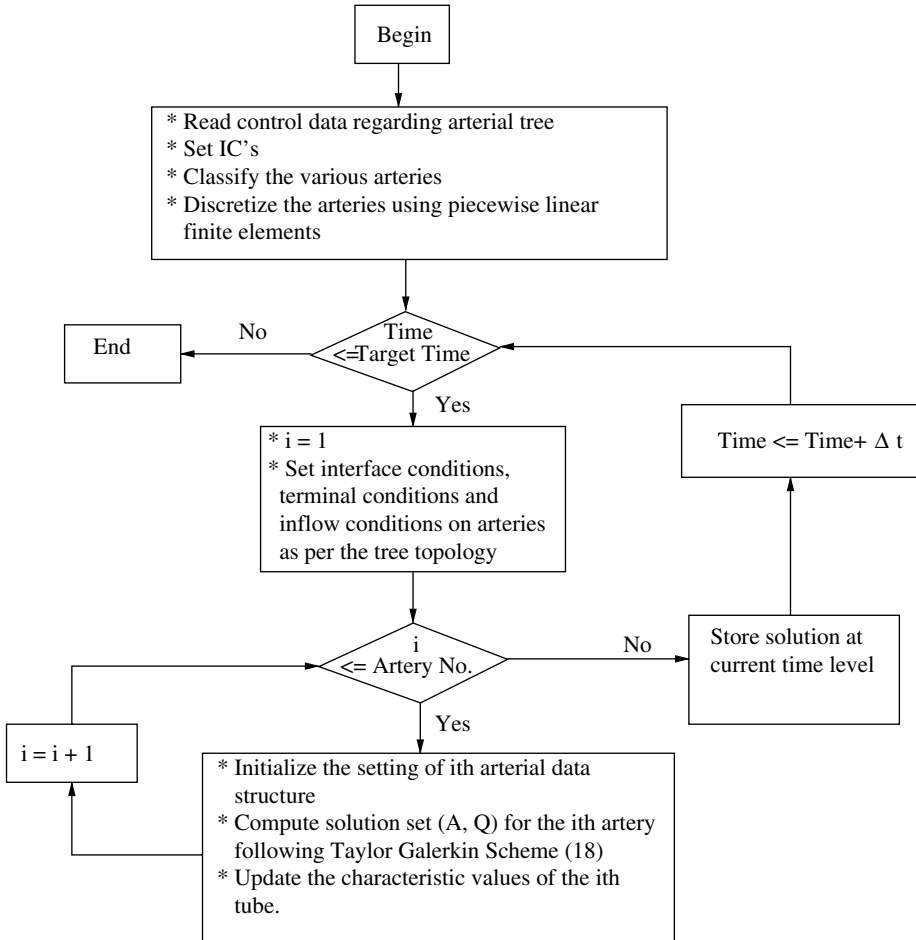


Fig. 3. Flowchart for the sequential algorithm



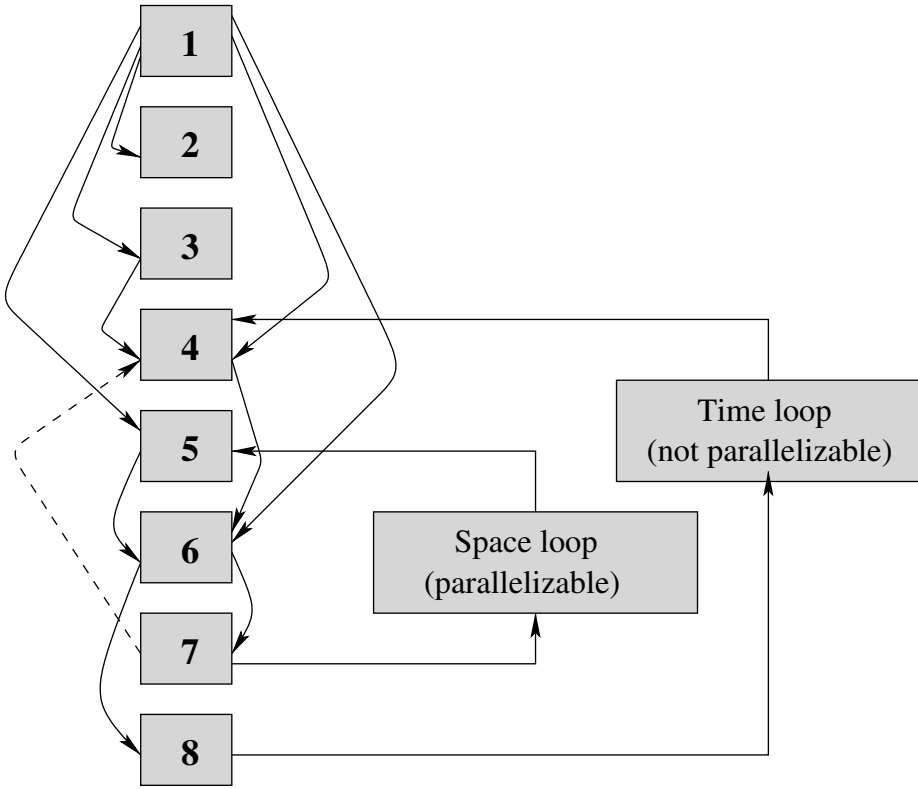


Fig. 4(a). Main Data Flow Graph depicting the data dependencies in solution procedure

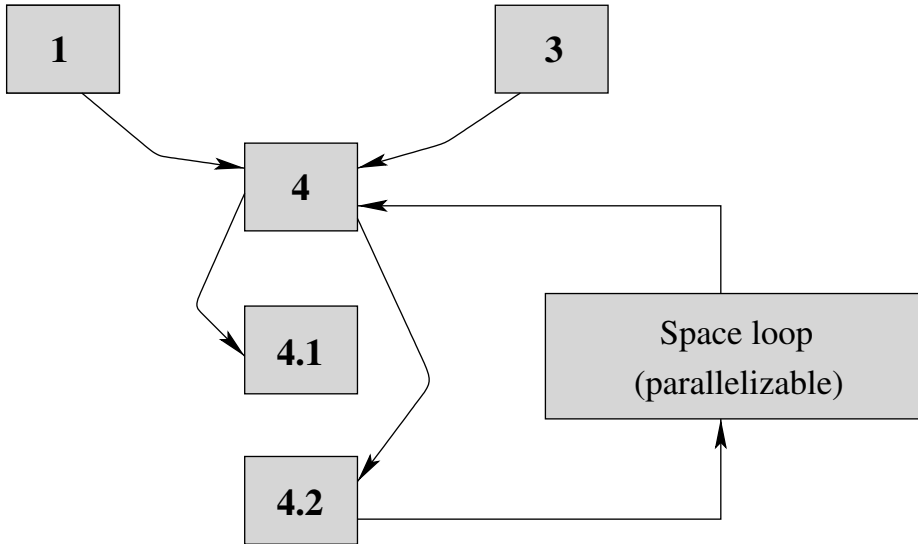
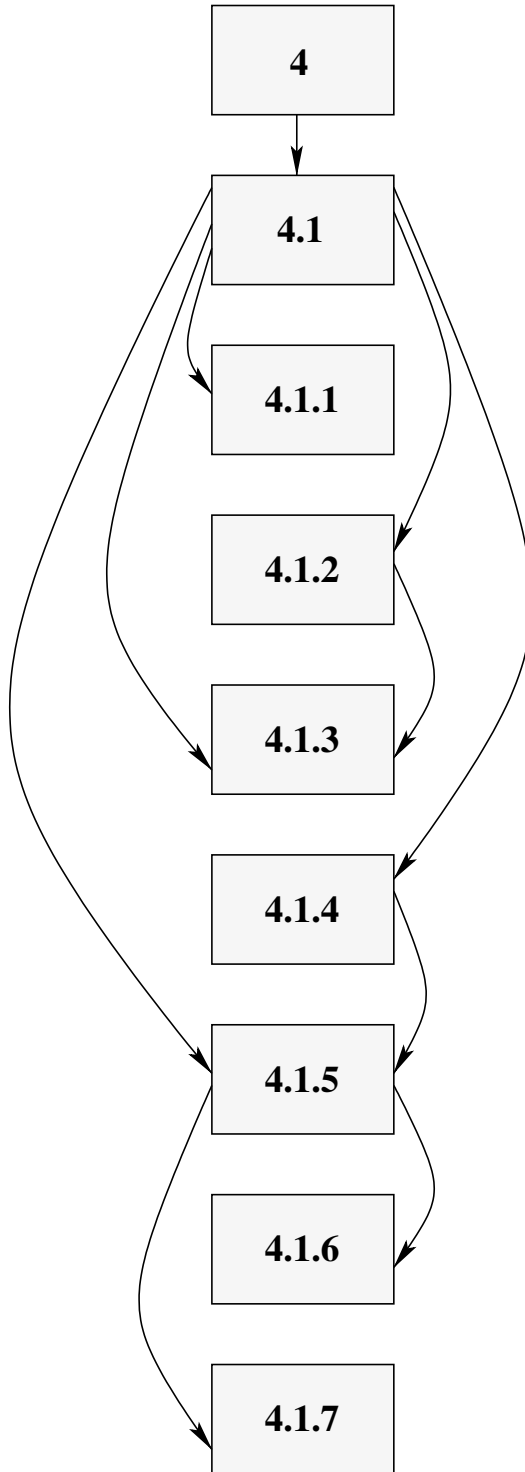


Fig. 4(b). Data Flow Graph depicting the data dependencies in setting interface boundary conditions during the temporal evolution of the solution





**Fig. 4(c).** Data Flow Graph depicting the data dependencies while setting interface boundary condition between two arterial segments

$$W_1^{n+1}(z_b) = W_1^n(z_b - \lambda_1^n(z_b)\Delta t) \tag{14}$$

When  $B(U) \neq 0$  the values of  $W_2^{n+1}(z_a), W_1^{n+1}(z_b)$  will have to be approximated by numerically solving the associated ODE system. The values of  $W_2^{n+1}(z_a)$  and  $W_1^{n+1}(z_b)$  together with the boundary conditions effectively complement the linear system provided by the discretization of the hyperbolic differential equations, which will be detailed in Section 3. Let us now consider the conditions at the interfaces between arteries and at bifurcations. Now suppose that  $\Omega_1 = [z_a, z_\Gamma]$  and  $\Omega_2 = [z_\Gamma, z_b]$  denote two arteries with an interface at  $z = z_\Gamma$ . At the interface the mass flux ( $Q$ ) and the total pressure ( $p_t$ ) are assumed to be continuous i.e.

$$Q_1 = Q_2, \quad p_{t,1} = p_{t,2}, \quad \text{at } z = z_\Gamma, \quad t > 0 \tag{15}$$

here  $p_t = p + \frac{1}{2}u^2$ .

While conservation of mass justifies the first interface condition, the second condition guarantees the energy inequality for the coupled system [5]. In order to compute the values of  $Q_i^{n+1}, A_i^{n+1}, i = 1, 2$  at  $z_\Gamma$ , conditions (15) are supplemented by the compatibility conditions in the form of extrapolation of characteristic variables exiting domain  $\Omega_1$  and  $\Omega_2$  at  $z = z_\Gamma$  are used to obtain the suitable values for  $A_i^{n+1}, Q_i^{n+1}, i = 1, 2$  at  $z = z_\Gamma$  [6]. In the case of bifurcation or branching we have three arterial segments, say  $\Omega_1 = [z_a, z_B]$ ,  $\Omega_2 = [z_B, z_b]$  and  $\Omega_3 = [z_B, z_c]$  where  $\Omega_1$  denotes the main branch and  $\Omega_2, \Omega_3$  denote the two branches of  $\Omega_1$ . Again, for the reason mentioned earlier, mass flux ( $Q$ ) and total pressure ( $p_t$ ) are assumed to be continuous across the branching, i.e.

$$Q_1 = Q_2 + Q_3, \quad p_{t,1} = p_{t,2} = p_{t,3} \quad \text{at } z = z_3, \quad t > 0 \tag{16}$$

These two conditions together with the three compatibility conditions permit to define  $A_i^{n+1}, Q_i^{n+1} (i = 1, 2, 3)$  at the bifurcation point i.e.  $z = z_B$  [6]. It may be noted that at the proximal end of the first artery in our arterial model pressure (i.e.  $A_i$ )

**Table 2.** Time profile of sequential execution for  $\Delta t = 0.000025, \Delta x = 0.1$

State no. in Data-flow graph	Actual time taken in seconds	% of total time (28980 s)	Parallelization status
1, 2, 3	100	0.0034	No
4	648	2.2	No
5	1772	6.11	Yes
6	23400	80.7	Yes
7	1424	4.91	Yes
8	268	0.09	Yes
Sys. Time	1368	4.72	No

has been imposed. The exact details of the imposed pressure pulse will be provided subsequently in results and discussion section.

### 3 Numerical Discretization

Let  $\Omega = [z_a, z_b]$  denote an artery of our network. Following a second order Taylor-Galerkin Scheme [15] we discretize the system (14) in time to obtain the following semi-discrete form:

$$\begin{aligned}
 U^{n+1} = U^n - \Delta t \frac{\partial}{\partial z} \left[ F^n + \frac{\Delta t}{2} H^n B^n \right] - \frac{\Delta t^2}{2} \left[ B_U^n \frac{\partial F^n}{\partial z} - \frac{\partial}{\partial z} \left( H^n \frac{\partial F^n}{\partial z} \right) \right] \\
 + \Delta t \left( B^n + \frac{\Delta t}{2} B_U^n B^n \right) \quad n = 0, 1, 2, \dots
 \end{aligned}
 \tag{17}$$

where  $\Delta t$  denotes the time step and the superscript ‘ $n$ ’ denotes the time level.  $F^n$  stands for  $F(U^n)$  and  $B_U^n$  stands for  $\left(\frac{\partial B}{\partial U}\right)^n$ .

Suppose that  $V_h$  be the space of piecewise linear finite element functions. Let  $V_h = [v_h]^2$ ,  $V_h^0 = \{v_h \in V_h | v_h = 0 \text{ at } z = a, b\}$  and  $(u, v) = \int_a^b u \cdot v dz$  the  $L^2(a, b)$  scalar product. After carrying out the spatial discretization by linear finite elements the problem reduces to finding a solution  $U_h^{n+1} \in V_h$  s.t.

**Table 3.** Sample data related to the load distribution achieved by Greedy approach

Number of processing elements	Processing element index	Assigned load (# of arteries)	Assigned DOF	Assigned arteries numbers
Two	1	28	2822	7,8,15,52,16,12,25,6,49,48, 27,24, 32,9,33,42,43,28, 18,40,14,19,3,51, 45,29,30,41
-do-	2	27	2822	21,22,5,46,13,17,11,20,55,54, 39,26, 10,23,31,34, 50,44,47,53,1,4,38,36, 2,37,36
Four	1	13	1408	7,46,11,20,48,32,31,43, 47,1,4,51,29
-do-	2	13	1410	21,52,12,55,54,10,33, 50,18,14,19, 36,35
-do-	3	15	1414	8,15,16,25,49,27,24,9,42, 28,53,3,45, 37,41
-do-	4	14	1404	22,5,13,17,6,39,26, 23,34,44,40,38, 2, 30
Eight	1	7	708	7,39,9,28,14, 37,41
-do-	2	6	704	21,27,23,44,3,29
-do-	3	7	704	8,11,54,33,18,19,45
-do-	4	7	704	22,25,48,31,47,38,51
-do-	5	7	704	15,12,49,32,43,4,2
-do-	6	7	702	5,17,55,26,34,53,35
-do-	7	7	700	46,16,20,10,50,40,30
-do-	8	7	712	52,13,6,24,42,1,36

$$\begin{aligned}
 (U_h^{n+1}, \psi_h) &= (U_h^n, \psi_h) + \Delta t \left( F_{LW}(U_h^n), \frac{d\psi_h}{dz} \right) - \frac{\Delta t^2}{2} \left( B_U(U_h^n) \frac{\partial F(U_h^n)}{\partial z}, \psi_h \right) \\
 &\quad - \frac{\Delta t^2}{2} \left( H(U_h^n) \frac{\partial F(U_h^n)}{\partial z}, \frac{d\psi_h}{dz} \right) + \Delta t (B_{LW}(U_h^n, \psi_h)) \quad \forall \psi_h \in V_h^0
 \end{aligned}
 \tag{18}$$

subjected to the boundary conditions defined earlier.

Here,

$$F_{LW}(U) = F(U) + \frac{\Delta t}{2} H(U)B(U) \text{ and } B_{LW}(U) = B(U) + \frac{\Delta t}{2} B_U(U)B(U)$$

Linear stability analysis [17] suggests that the following condition should be satisfied:

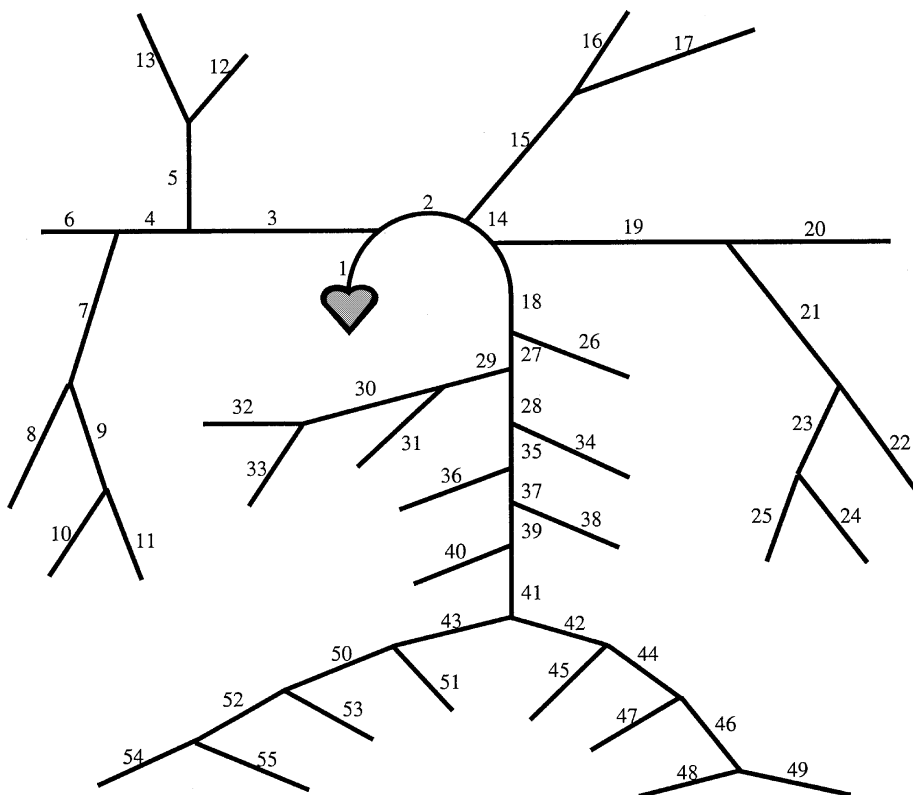
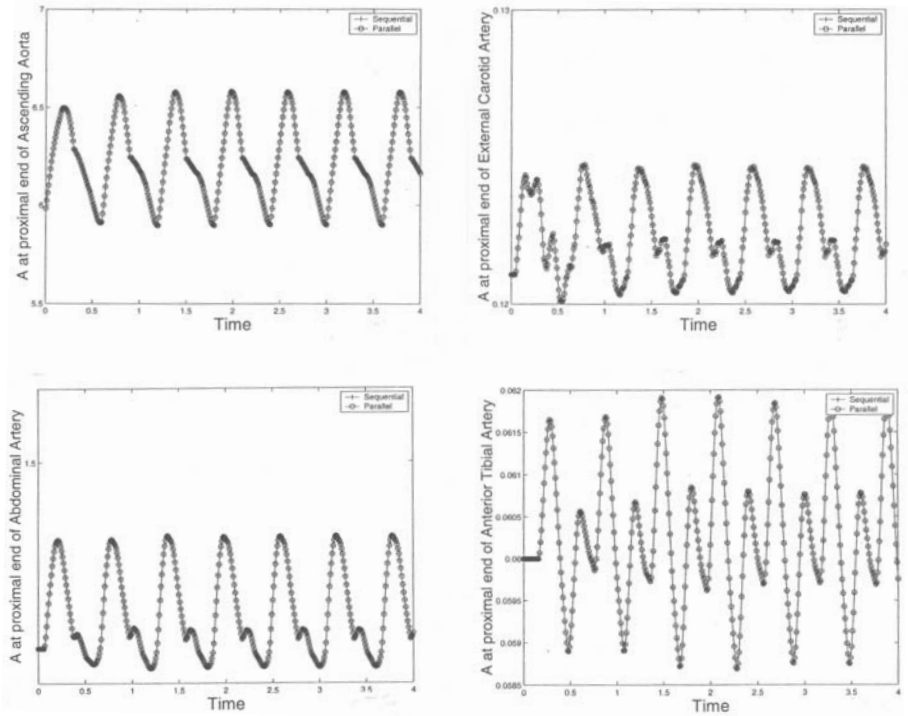


Fig. 5. Actual arterial tree model employed in numerical computations



**Fig. 6.** Comparison of 'A' values obtained from sequential and parallel computation corresponding to the proximal ends of (a) Ascending Aorta, (b) External Carotid Artery and (c) Abdominal Artery and (d) Anterior Tibial Artery

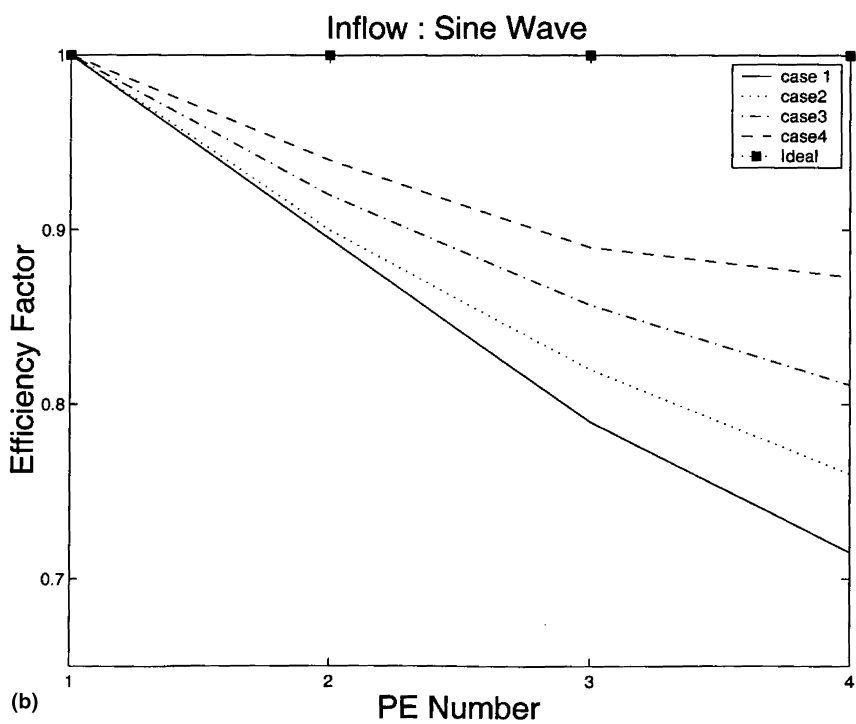
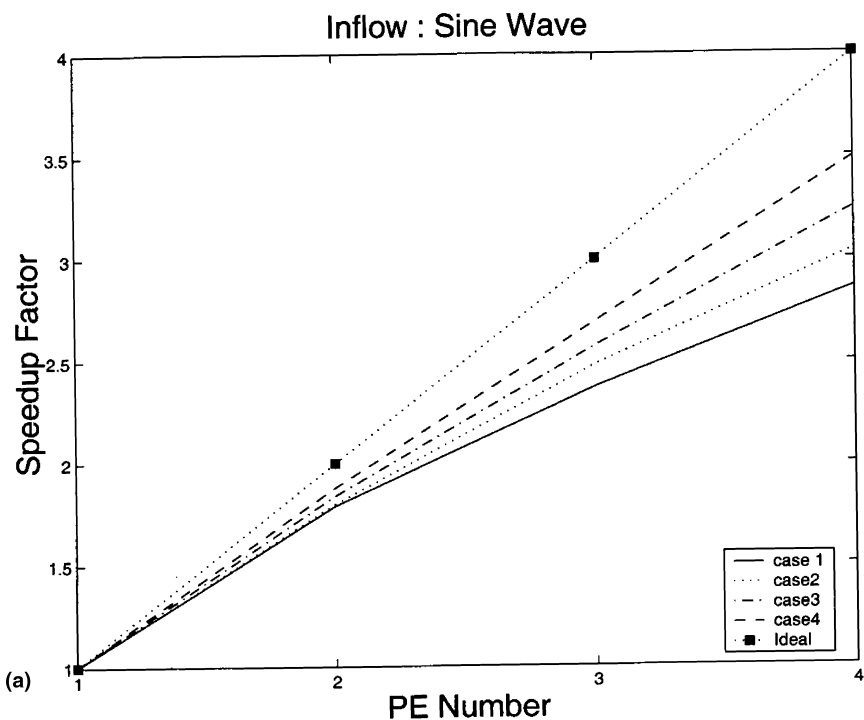
$$\Delta t \leq \frac{\sqrt{3}}{3} \min_{0 \leq i \leq N} \left[ \frac{h_i}{\max(\lambda_{1,i}, \lambda_{1,i+1})} \right] \tag{19}$$

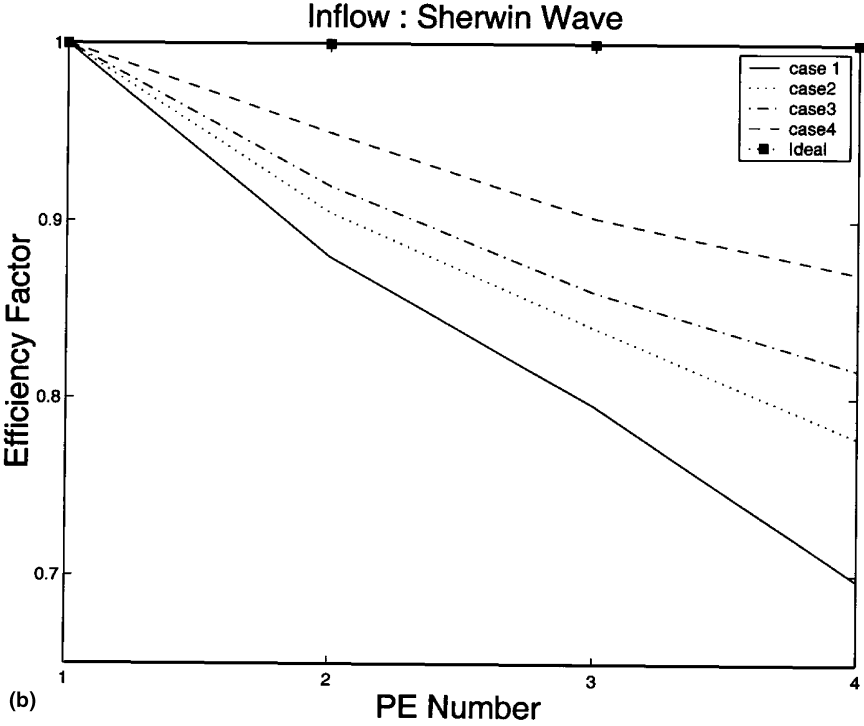
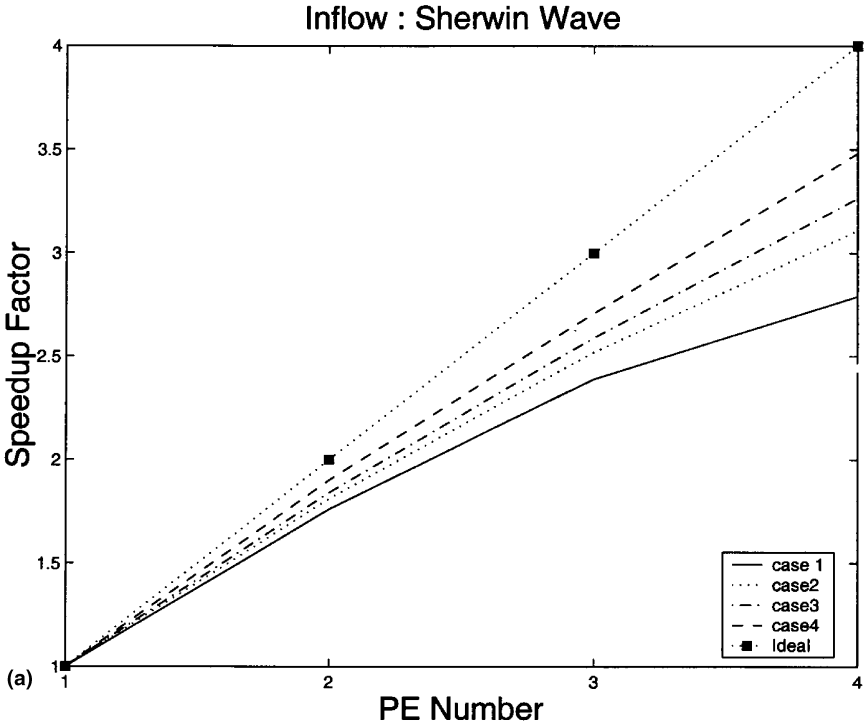
where  $\lambda_{1,i}$  denotes the value of  $\lambda_1$  at  $i^{th}$  mesh node. The interval  $[z_a, z_b]$  is divided into  $N$  linear elements  $[z_i, z_{i+1}], i = 0, 1, \dots, N$  with  $z_i = a + ih_i$ ,  $h_i$  being local element size.

The various steps involved in the sequential computation of the solution of  $(A, Q)$  for the entire arterial tree are presented schematically through the computational flow diagram in Figure 3.

**Fig. 7(a).** Speedup factor vs. PEs number obtained in the parallel computation with sinusoidal inflow waveform

**Fig. 7(b).** Efficiency factor vs. PEs number obtained in the parallel computation with sinusoidal inflow waveform









**Fig. 8a.** Speedup factor vs. PEs number obtained in the parallel computation associated with realistic inflow waveform

**Fig. 8b.** Efficiency factor vs. PEs number obtained in the parallel computation associated with realistic inflow waveform

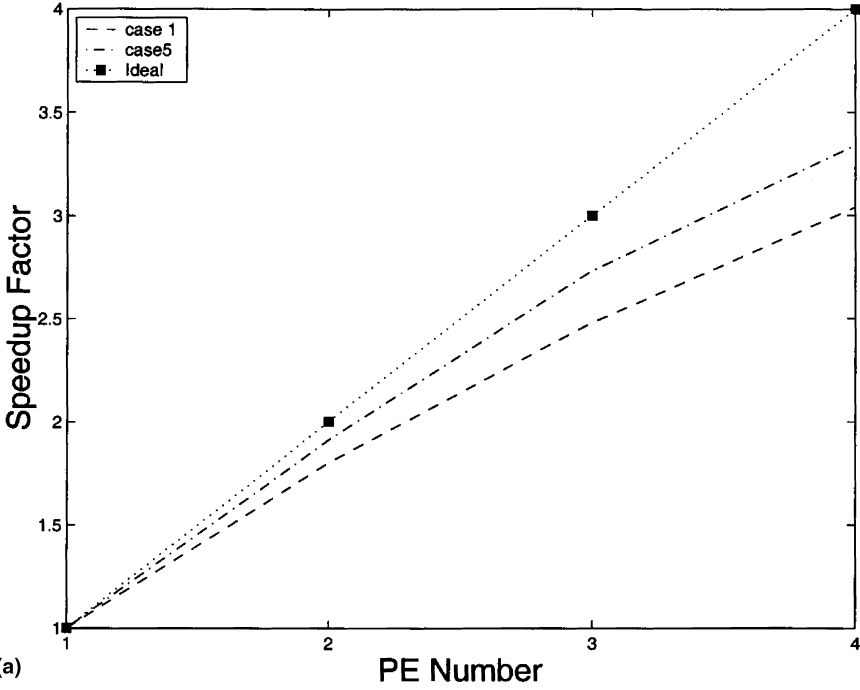
#### 4 Parallel Implementation

In view of the explicit nature of the computational scheme discussed under Section 2 and the geometry of the domain (Arterial tree) under consideration, one may think of two different strategies for the parallel computation of the solution to the mathematical model, namely, (a) Data Parallelization and (b) Algorithmic Parallelization. Under the former approach one would essentially partition the grid and distribute the same among various Processing Elements (PEs) to concurrently carry out all the computations related to those grid points primarily on those PEs with the necessary communications. Whereas, under Algorithmic Parallelization one would to begin with segregate the computations related to various field variables of the model under study, which in this case happens to be two (i.e.  $A, Q$ ), and assign a cluster of PEs to carry out all the computations pertaining to each of these variables. Further, on each of these clusters a data parallelization strategy is adopted. This approach would lead to memory economic computations on PEs. But assigning the  $A$  and  $Q$  component of a grid node to different processors would result in a lot of communication. Typically, one needs to that communication cost should grow asymptotically slower than computation cost for an increasing problem size (number of dofs). This can be achieved by the data parallel approach. Also, owing to the 1-D nature of the model and memory not being an issue we have adopted the former strategy in preference to the later, as it would avoid the extra book-keeping related to the grid partitioning on different clusters and the need for a special communication layer for inter and intra data transactions among the various PEs sitting in different clusters.

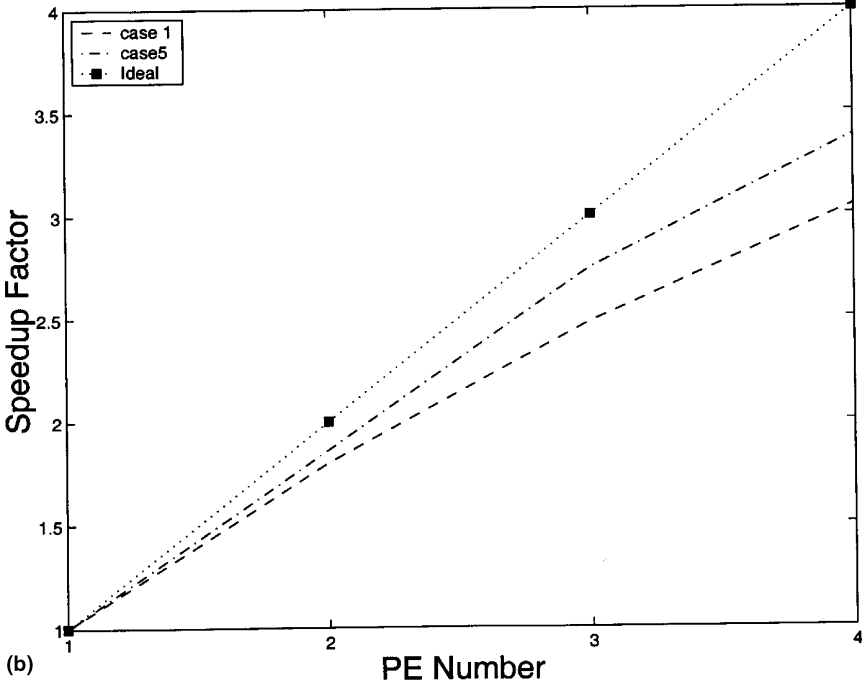
As a first step towards the development of a suitable MIMD parallel algorithm under the paradigm of data parallelism approach one has to carry out a through data flow analysis to trace out the exact data dependencies between the various stages of the sequential computation procedure. For this purpose data flow graphs depicting the data dependencies among various modules constituting the sequential code have been generated. Under the current sequential solution strategy, the data dependency graphs for either of the field variables (i.e.  $A, Q$ ) turn out to be one and same. Hence, it is enough to trace the dependency graphs w.r.t. one of these variables say,  $A$ . The various states in the data flow graphs for the variable  $A$  in Figure 4(a-c) correspond to various modules in source code as follows:

1. i/o modules: (1) read-data (artery geometry, arterial wall properties, control parameters, inflow details etc.)
2. Solution Process modules: (2) prep\_bc (setting arterial network connectivity for applying boundary conditions), (3) define artery geometry (discretizing the

### Inflow : Sine Wave



### Inflow : Sherwin Wave





- Fig. 9(a).** Speedup factor vs. PEs number obtained in the parallel computation to assess the influence of CFL number with sinusoidal inflow waveform
- Fig. 9(b).** Speedup factor vs. PEs number obtained in the parallel computation to assess the influence of CFL number with realistic inflow waveform

domain, checking the presence of a stenosis or a vascular prosthesis or a stent etc.), (4) `put_bc` (setting the interface boundary conditions for various arteries to maintain the continuity of solution in the arterial tree during temporal evolution of periodic solution), (5) `init-modelle1D` (initializing various data structures related to special computations during temporal evolution of periodic solution), (6) `Solver` (carries out the actual Taylor-Galerkin sequential scheme on various arterial segments), (7) `update_W12_n` (update the characteristic variables after every special computation before progressing to the next time step), (8) `store_sol` (stores the solution history).

The data flow graph depicting the data dependencies between the eight modules listed above is shown in Figure 4(a). Here the arrows emerging from state '*i*' to '*j*' indicate the data dependency of state '*j*' on '*i*'. The space and the time control loops decide the modules involved in temporal evolution of spatial solutions. Here, firstly one would investigate the possible parallelism in the modules under time and space control loops. Under the time-marching numerical scheme, the temporal loop is not parallelizable. However the data-dependencies among the modules 5, 6 and 7 under spacial loop are local, i.e. confined to individual arteries in the arterial tree, and hence are amenable to data parallelization. One has to investigate the presence of any data parallelism within other left out modules (i.e. 1, 2, 3, 4 and 8). As modules 1 and 8 are related to i/o operations, no further investigation is needed other than checking the hardware parallel i/o capability on the target parallel computing platform, if it is an i/o intensive computation. Clearly the current 1-D calculations are not an i/o intensive computation. Modules 3 and 4 are to be further investigated. Module 4 being inside the time loop will be the first one to be considered. It has two sub-modules, namely, (4.1) `W_interface_2t` (to fix the boundary conditions at the interface between the two arteries) and (4.2) `W_interface_3t` (to fix the boundary conditions at the interface between the main artery and the two arteries bifurcating from it). Each of these sub-modules again has the similar set of sub-sub-modules. The sub-sub-modules under (4.1) and (4.1.1) `compute-eig` (computes eigen values), (4.1.2) `interpolate` (carries out interpolation of characteristic values), (4.1.3) `var_to_Riemann` (evaluates characteristic variables from  $A$  and  $Q$ ), (4.1.4) `calculate_F2` (Evaluates flux terms), (4.1.5) `calculate-JF` (evaluates Jacobian associated with flux calculations), (4.1.6) `DGETRF`, (4.1.7) `DGETRI` (LAPACK routines). The data flow graphs corresponding to the modules under (4) are provided in Figures 3(b-c). From Figure 3(b) one may notice that the modules (4.1) and (4.2) may be executed concurrently. Data flow graph in Figure 3(c) depicts very little scope for further parallelism under module (4.1).

The second step towards the development of parallel algorithm is to build a time profile table for the sequential code. The time profile data of the sequential execution when  $\Delta t = 0.000025, \Delta x = 0.1$  is presented in Table 2. From the time profiles it is clear that most of the execution time is devoted to the time loop and of this more than 90% of time is taken by the space loop consisting of modules 5, 6 and 7. Dataflow graphs suggest that space loop is amenable to data parallelism. Now an appropriate load-distribution strategy has to be devised to maximize the computation on all PEs and to simultaneously minimize the communications among PEs.

Greedy Algorithmic approach has been used to devise a load distribution strategy. A Greedy algorithm or a “single minded” algorithm is an algorithm that gobbles up all its favorites first. The idea behind it is to perform a single procedure in the recipe over and over again until it can’t be done any more and see what kind of results it will produce. It is one of the ways of approaching the problem and it may lead to very good results at times. Sometimes it may not completely solve the problem too. In the current situation of distributing arteries with different lengths and with different degrees of freedom to PEs a greedy load distribution strategy seems to work. The idea behind this Greedy approach is to first arrange the arteries in a descending order with respect to their associated degrees of freedom

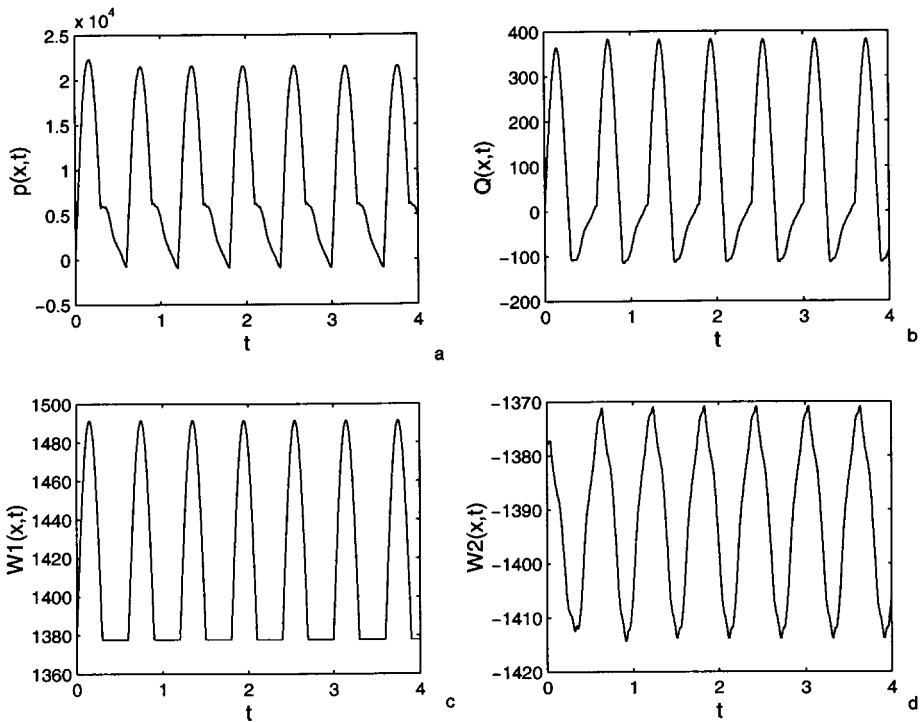


Fig. 10. Time history plots of (a)  $p(x,t)$ , (b)  $Q(x,t)$ , (c)  $W_1(x,t)$  and (d)  $W_2(x,t)$  over a period of  $4T$  associated with Ascending Aortic Artery in the absence of terminal resistance

(or length) and keep assigning the load (i.e. arteries) to the first identified PE with least load (i.e. First PE with least load is allowed to snatch the load for computation). This Greedy approach is seen to lead to an acceptable and nearly even load-distribution in the present situation. Table 3 presents the load-distribution achieved by this approach when applied to our simplified arterial tree model. In Table 3 load distribution pertaining to the arterial tree model with total degree of freedom (i.e.  $2818 \times 2$ ) on 2, 4, 8 PEs are presented. Here one may note that the Greedy load distribution strategy has been implemented in a static manner i.e. load-distribution module is run as a pre-processing module on all the PEs. Further, it assumes that there are more arteries than processors and load difference between the arteries is not too much. In case of any significantly large variation in the load between arteries, for parallel computations one may think of segmenting such arteries with very large loads, with due consideration made to the communication overheads, into smaller ones with suitable interface conditions. It is also to be noted that the module 4 has not been parallelized because of the following reasons- (1) it takes less than 8% of the time control loop execution time, (2) its parallelization demands a involved communication transactions which may effectively lead to a loss in speedup factor as quantum of data-parallelism under it is insignificant.

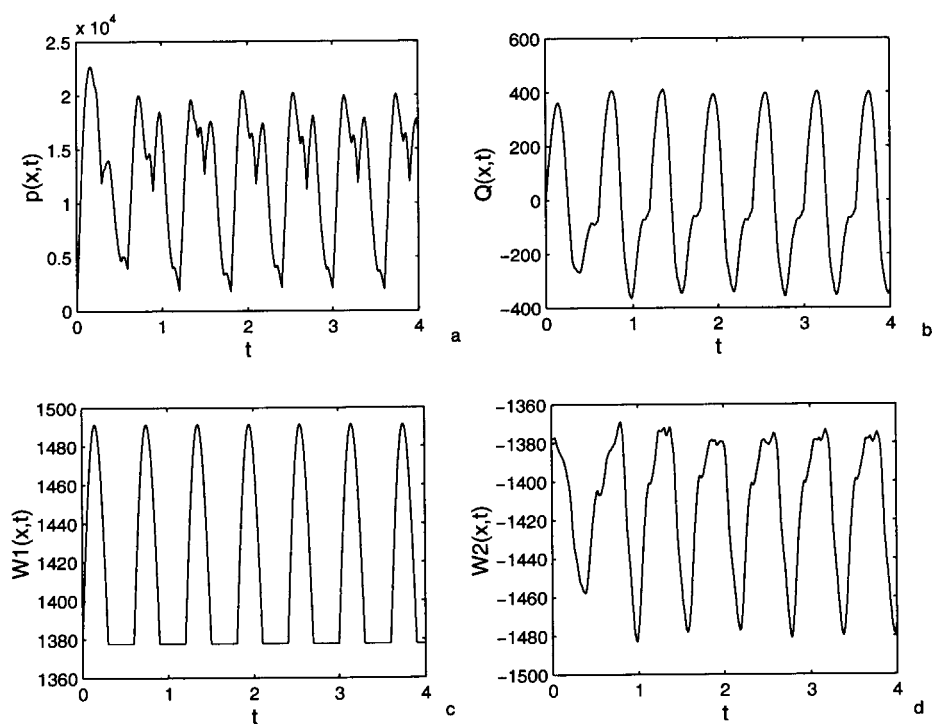


Fig. 11. Time history plots of (a)  $p(x,t)$ , (b)  $Q(x,t)$ , (c)  $W_1(x,t)$  and (d)  $W_2(x,t)$  over a period of  $4T$  associated with Ascending Aortic Artery in the presence of terminal resistance

The parallel computations have been carried out on parallel computer MASG68 at EPFL, Lausanne. MASG68 is a Silicon Graphics Deskside Origin 2000 class of machine with eight R10000 Processing Elements under IRIX 6.5.4f operating system. It uses Silicon Graphics distributed shared-memory multi-processing architecture i.e. S2MP. The current parallel implementation is primarily based on Message Passing Interface libraries (MPI 3.1.X.15) as on MASG68 machine.

### 5 Results and Discussions

The actual human vascular tree model employed in the numerical computations is depicted in Figure 5. One may note here that in the flow model the arteries are treated as linear segments and no bifurcation angle is accounted for. For a discussion on how to include bifurcation angle into consideration, the reader may refer to [6]. For the numerical simulations in our simplified human vascular model two different inflow pressure impulses [6, 7], have been considered at the proximal end of Ascending Aorta. In the discussion to follow the first of these inflow conditions, defined by:

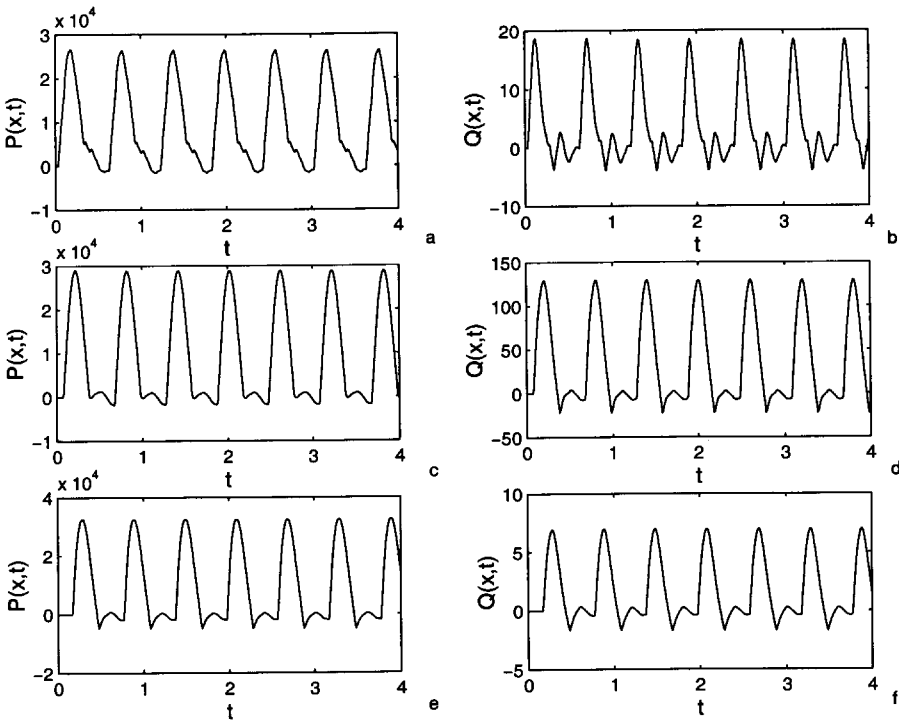


Fig. 12. Time history plots of  $p(x,t)$  corresponding to (a) Right Carotid Artery, (c) Abdominal Artery, (e) Left Femoral Artery with sinusoidal inflow waveform in the absence of terminal resistance. Corresponding  $Q(x,t)$  plots are in Figures 12(b, d, f)

$$A(0, t) = \frac{(P_0 + 20000 \sin(400\pi(t - t_0)))}{4\beta^2} \tag{19}$$

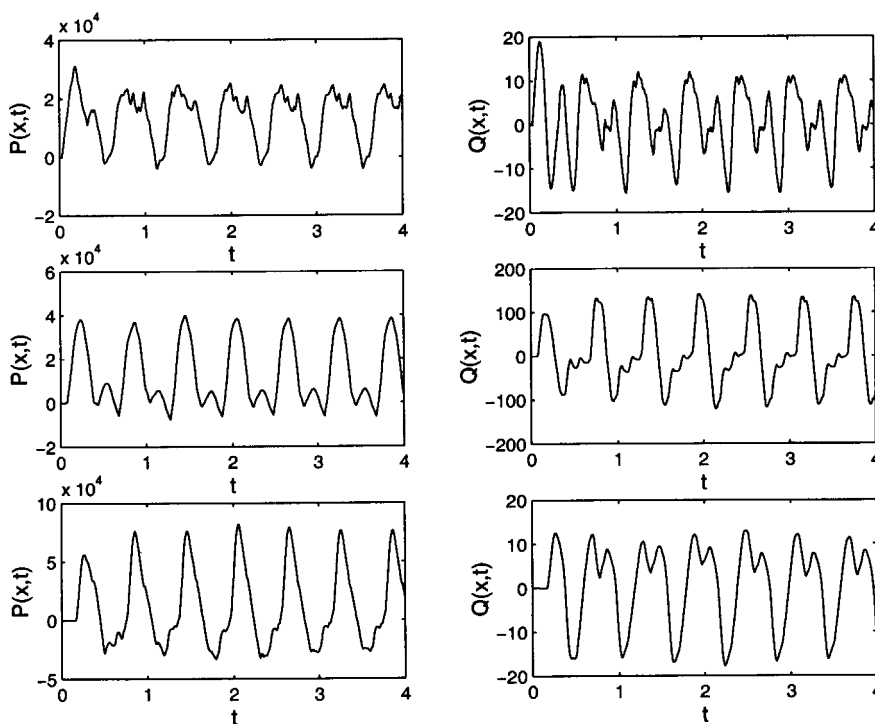
will be referred to as sinusoidal inflow. The second inflow condition given by

$$A(0, t) = 1.01 - 0.02\sin(400\pi(t - t_0)) - 0.02\sin(800\pi(t - t_0)) - 0.01 \cos(800\pi(t - t_0)) \tag{20}$$

shows a more realistic waveform and will be referred to as “realistic inflow”.

In view of periodic nature of the inflow waves all the computations have been carried out for a period of four cycles i.e.  $4T$ ,  $T$  being the period of one cycle which in this case happens to be unit value. Results to be discussed clearly indicate that the simulations for four cycle time stretch is adequate to ensure the periodicity in solution.

To begin with, the parallel code has been validated by solving the system of equations (1–2) representing the flow in our human arterial tree model and comparing the results obtained by the validated sequential code [6]. For this



**Fig. 13.** Time history plots of  $p(x, t)$  corresponding to (a) Right Carotid Artery, (c) Abdominal Artery, (e) Left Femoral Artery with sinusoidal inflow waveform in the presence of terminal resistance. Corresponding  $Q(x, t)$  plots are in Figures 13(b, d, f)

comparison we have chosen four different arteries namely, Ascending Aorta, External Carotid Artery, Abdominal Artery and Anterior Tibial Artery representing the top, middle and the bottom portions of the arterial tree model. In Figures 6(a-d) the values of  $A$  and  $Q$  corresponding to these four arteries obtained with the parallel computations are in 100% agreement with those from the sequential computations. For these numerical computations a sinusoidal pressure pulse (i.e. for  $A$ ) as inflow condition from heart to proximal end of Ascending Aorta has been considered. One may notice that despite the initial data is  $A = A_0$  and  $Q = 0$  everywhere, the periodicity of the solution established in the second cycle itself.

Now the performance of the parallel code in terms of speedup and efficiency factors will be discussed in detail. For this purpose, four different cases consisting of (1) 5636, (2) 9800, (3) 14700 and (4) 29400 degrees of freedom (dof) have been considered. These four cases correspond to the grid size  $\Delta x$  or  $h = 0.2(0.5), 0.15, 0.1, 0.05$ . The corresponding  $\Delta t$  values are  $3e-05, 5e-05, 25e-06$  and  $3e-06$ . These values have been chosen in view of the stability criteria. Primarily numerical simulations have been carried out on ' $n$ ' number of PEs ( $n = 1, 2, 3, 4$ ) of MASG68. The speedup factor is measured as the ratio of CPU time taken to run the code on one processor to the time taken to run the code on

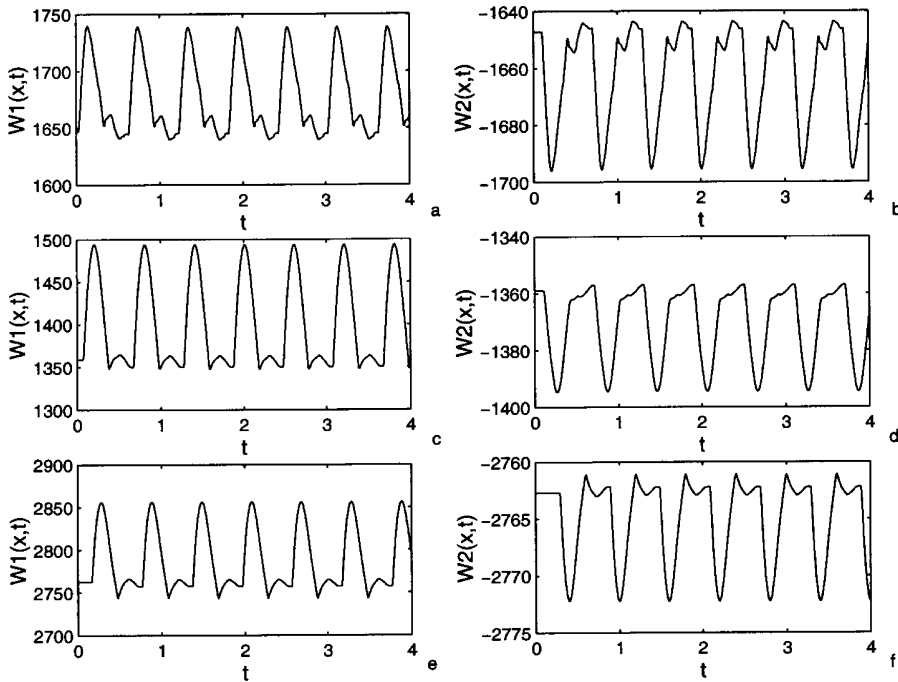


Fig. 14. Time history plots of  $W_1(x,t)$  corresponding to (a) Right Carotid Artery, (c) Abdominal Artery, (e) Left Femoral Artery with sinusoidal inflow waveform in the absence of terminal resistance. Corresponding  $W_2(x,t)$  plots are in Figures 14(b, d, f)



' $n$ ' processors. The efficiency factor is measured by the ratio of speedup factor to the number of PEs. The speedup and efficiency factors corresponding to the parallel implementations with sinusoidal inflow condition are presented in Figures 7(a–b). In Figures 8(a–b) the speedup and efficiency factors corresponding to realistic wave inflow condition are presented. From Figures 7(a) and 8(a) it may be noted that as the data size is increased from 5636 dof to 29400 dof there is a definite increase in the speedup factor. With the increase in data size i.e. as we move from case (1) to case (4) the speedup factor, on four PEs, in Sine wave inflow case increases from 2.865 to 3.49 and in realistic wave case it increases from 2.89 to 3.482. In Figures 7(b) and 8(b) the corresponding efficiency factors are presented. Clearly with the increase in data size parallel code gets more efficient. As we move from case (1) with 5636 dof to case (4) with 29400 dof on four PEs, with both the inflow conditions more than 87% efficiency has been achieved. With 29400 dof on eight PEs the efficiency drops down to 68%. With the further increase in data size an improvement in efficiency level is noticed. Since there is only a marginal variation in the solution as one moves from 9800 dofs to higher dofs the idea of further increasing the data size is not followed up.

Next the influence of CFL number on the speedup factor is analysed. On keeping the grid-size ' $h$ ' constant and on decreasing ' $\Delta t$ ' the stability criteria is satisfied to

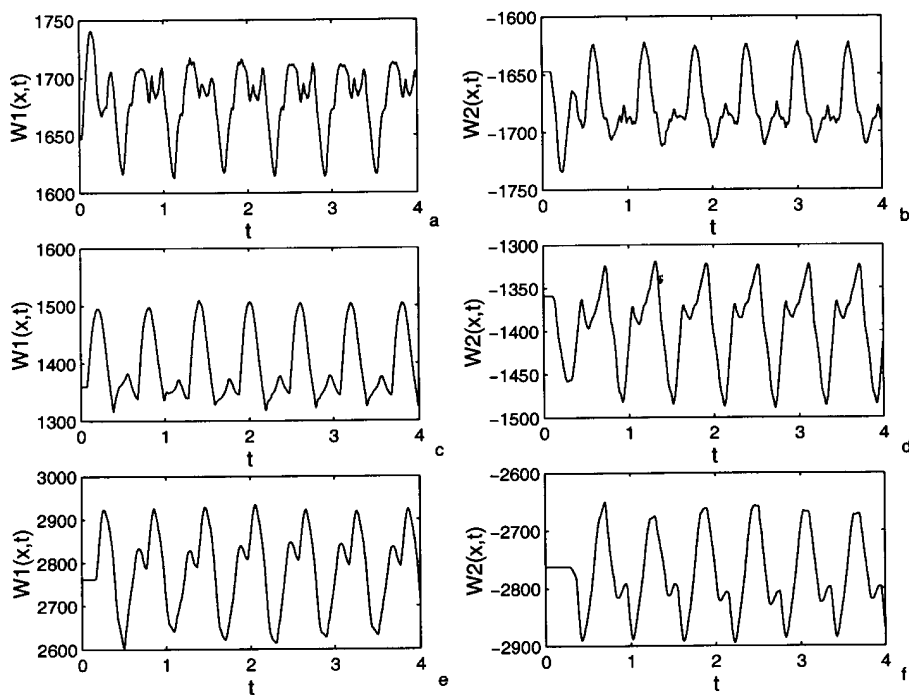


Fig. 15. Time history plots of  $W_1(x,t)$  corresponding to (a) Right Carotid Artery, (c) Abdominal Artery, (e) Left Femoral Artery with sinusoidal inflow waveform in the presence of terminal resistance. Corresponding  $W_2(x,t)$  plots are in Figures 15(b, d, f)

an increased extent. However this will lead to an increase in the bulk of the computational work. This increase in computational load is due to the increase in number of time steps and not due to the increase in dof. To assess its influence on the performance of the parallel code settings based on case (b) i.e.  $\Delta x = 0.15, \Delta t = 0.00005$  have been considered. The speedup factors corresponding to this case, as  $\Delta t$  is decreased from 0.00005 to 0.000025 both for sinusoidal inflow and realistic inflow conditions, are presented in Figure 9(a) and 9(b). From these two figures it is clear that as the CFL number is decreased by decreasing  $\Delta t$  keeping  $\Delta x$  (i.e.  $h$ ) fixed, there is a gain in the speedup factor due to the increase in computational load. Though actually the ratio of computation to communication is not changed there is a change in the ratio of computation time to communication time. This is due to faster communication enabled by the tightly coupled shared memory architecture of SGI 2000 series of machines with high bandwidth inter connectors called NUMalink<sup>TM</sup>.

In Figures 10(a-d) and 11(a-d) time histories of  $p(x, t), Q(x, t), W_1(x, t), W_2(x, t)$  over a period of  $4T$  for Aortic artery with sinusoidal inflow pressure wave pattern in the absence and presence of terminal resistance are presented respectively. The pattern of  $W_1(x, t)$  in Figures 10(c) and 11(c) match very well with those presented by Sherwin et al. [7]. From Figures 10(a) and 11(a) corresponding to the pressure

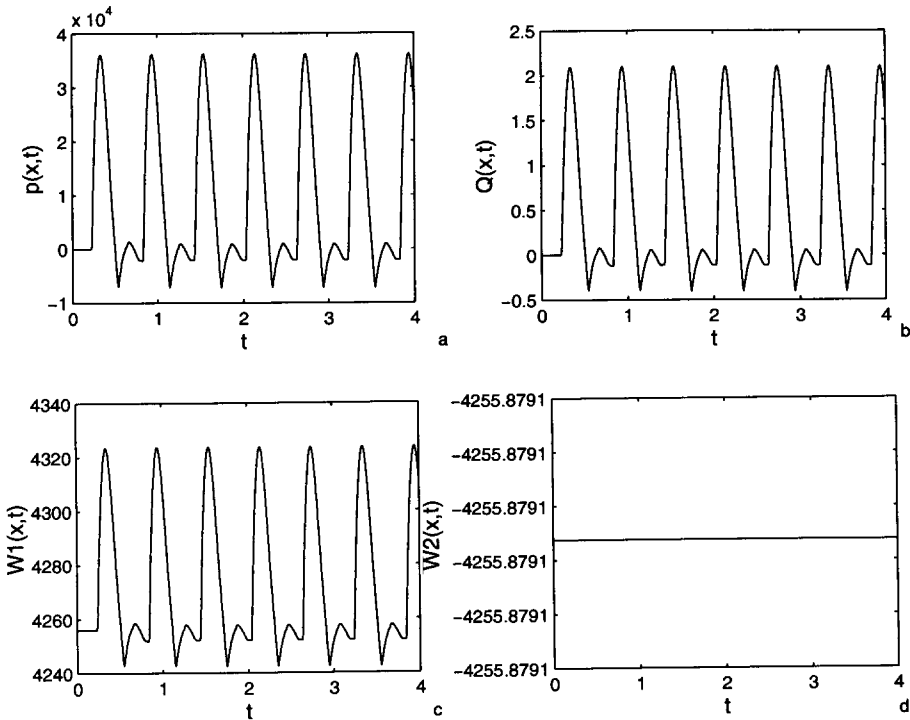


Fig. 16. Time history plots of (a)  $p(x, t)$ , (b)  $Q(x, t)$ , (c)  $W_1(x, t)$  and (d)  $W_2(x, t)$  corresponding to the Left Anterior Tibial Artery in the absence of terminal resistance

waves in Aortic artery one may clearly find that introduction of a terminal resistance into the arterial tree model leads to more oscillations specially in the vicinity of peak pressure locations. One may also notice that in the presence of terminal resistance the minimum values reached by the pressure are positive and the manner in which the pressure waves reach their minimum values is also different. From Figures 10(d) and 11(d) corresponding to the characteristic waves  $W_2$ , one may notice that the spikes get flattened (with small oscillations) when terminal resistance is introduced. Also the minimum value attained by the characteristic waves is enhanced.

Now in Figure 12(a-f) and 13(a-f) time histories of  $p(x,t)$  and  $Q(x,t)$  over a period of four cycles for Right Carotid artery, Abdominal Artery and Left Femoral Artery with sinusoidal inflow pressure wave pattern in the absence and presence of terminal resistance are presented respectively. The corresponding characteristic waves  $W_1(x,t)$  and  $W_2(x,t)$  both in the absence and presence of terminal resistance are presented in Figures 14(a-f) and 15(a-f) respectively. In Figures 16(a-d) and 17(a-d) time histories of  $p(x,t)$ ,  $Q(x,t)$ ,  $W_1(x,t)$  and  $W_2(x,t)$  corresponding to the Left Anterior Tibial Artery are presented. Clearly the time history plots depict that the pressure, flow and characteristic wave patterns in the entire arterial network are very sensitive to the pressure of terminal resistance.

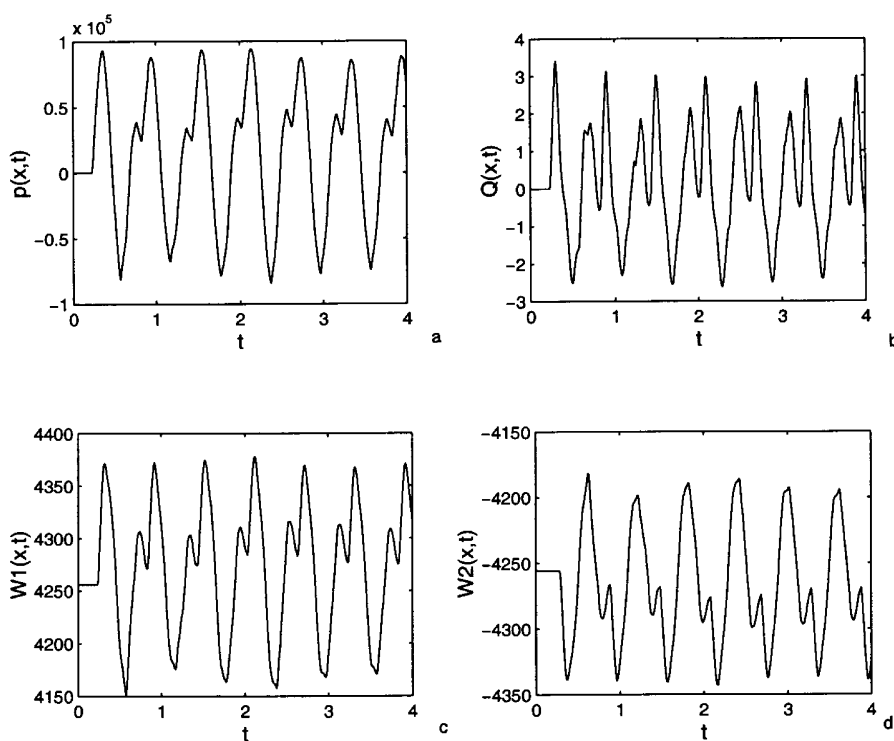


Fig. 17. Time history plots of (a)  $p(x,t)$ , (b)  $Q(x,t)$ , (c)  $W_1(x,t)$  and (d)  $W_2(x,t)$  corresponding to the Left Anterior Tibial Artery in the presence of terminal resistance

The inclusion of resistance to the terminal arteries increases the number of waves in the system due to the forward traveling waves being reflected at the terminal vessels and introduces backward traveling waves,  $W_2$ , which are re-reflected at the bifurcations, hence a complex pattern of waves occur in the network. Introducing resistance has changed the shapes of all the waves throughout arterial network. The shape of the waves varies significantly from vessel to vessel whereas the shapes of the waves in the network with no terminal resistance were all very similar. For perfectly matched arteries there should be no backward traveling wave,  $W_2$ , if there is no terminal resistance because there should be no reflections at the bifurcations and there are no reflections at the terminal vessels. Figure 16(d) shows that there is no  $W_2$  wave at the Left Anterior Tibial Artery. But from Figures 10(d), 14(b), 14(d), 14(f) and 16(d) one may find small  $W_2$  wave in some of the arteries because they are not perfectly well-matched forward waves and consequently small reflections occur at the bifurcations. Since the reflection coefficients are close in terminal vessels,  $W_2$  is similar in magnitude to  $W_1$  and will have a large effect on the waveforms, particularly in the end vessels (Figure 17(d)). All these observations agree well with those reported by Sherwin et al. [7].

Next in Figures 18(a-d), 19(a-f) and 20(a-d) time histories of  $p(x, t)$ ,  $Q(x, t)$ ,  $W_1(x, t)$ ,  $W_2(x, t)$ , in Ascending Aorta, Left Carotid Artery, Abdominal Artery, Femoral Artery and the Tibial Artery when the inflow pressure wave is a realistic

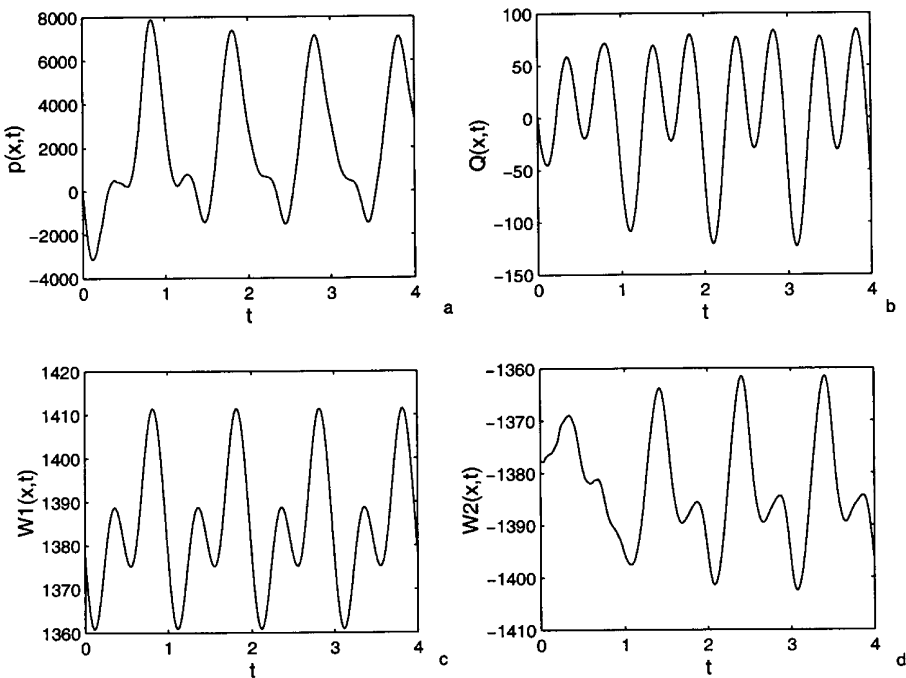


Fig. 18. Time history plots of (a)  $p(x, t)$ , (b)  $Q(x, t)$ , (c)  $W_1(x, t)$  and (d)  $W_2(x, t)$  corresponding to the Ascending Aorta in the presence of terminal resistance when the inflow pressure wave is a realistic wave

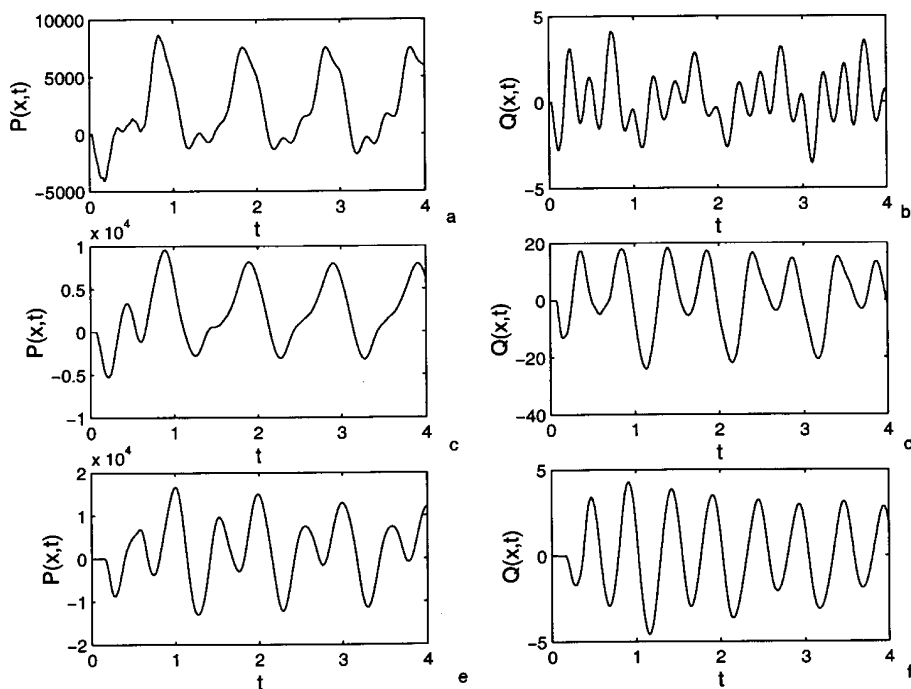


Fig. 19. Time history plots of  $p(x, t)$  corresponding to (a) Right Carotid Artery, (c) Abdominal Artery, (e) Left Femoral Artery with realistic inflow waveform in the presence of terminal resistance. Corresponding  $Q(x, t)$  plots are in Figure 19(b, d, f)

wave, as described in equation (21) are presented in the presence of terminal resistance. On comparing the peak values in pressure and flux plots corresponding to the case of sinusoidal inflow pressure with terminal resistance (i.e. Figures 11(a–d), 13(a–f), 17(a–d)) with those in Figures 18(a–d), 19(a–f) and 20(a–d) one may notice that the influence of sinusoidal wave inflow condition is more pronounced than that of realistic wave inflow condition even as far as Tibial Artery.

## 6 Conclusions

A MIMD parallel flow computation of blood flow in a simplified arterial network consisting of 55 main arteries has been successfully carried out by Taylor Galerkin Finite Element Method. Data flow graphs in conjunction with time profiles of sequential algorithm have been found to be effective in developing the parallel algorithm. The Greedy approach in load distribution has lead to a nearly equal automatic static load distribution on all PEs. Presence of terminal resistance is found to influence pressure and velocity waveforms in the entire human arterial network. The parallelization of the 1D code is an important step towards multi-scale model based cardio-vascular flow simulations.

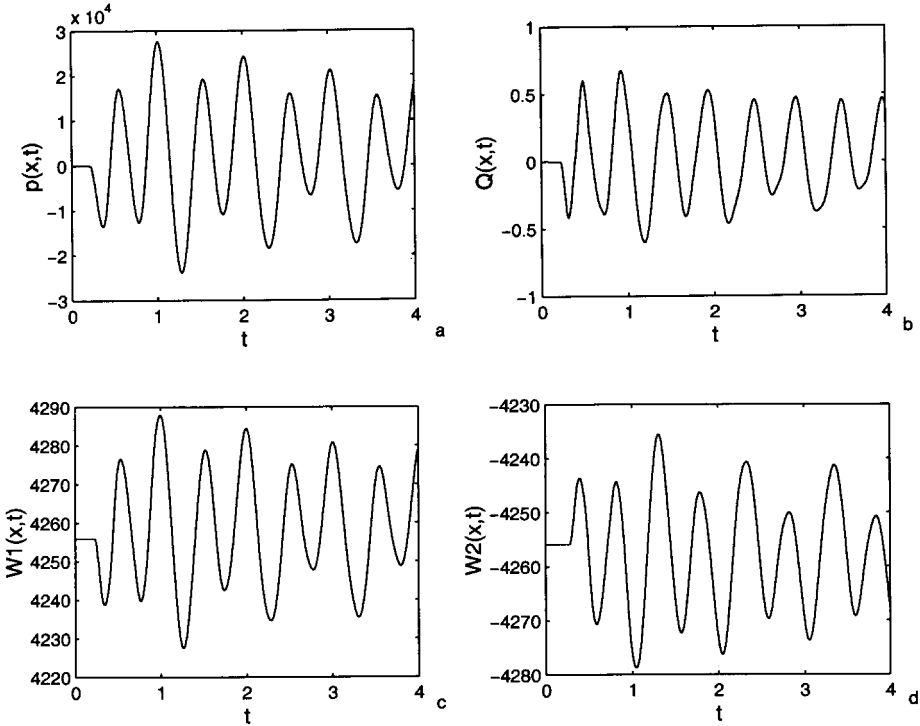


Fig. 20. Time history plots of (a)  $p(x,t)$ , (b)  $Q(x,t)$ , (c)  $W_1(x,t)$  and (d)  $W_2(x,t)$  corresponding to the Tibial in the presence of terminal resistance when the inflow pressure wave is a realistic wave

## References

- [1] Hughes, T. J. R., Taylor, C., Zarins, C.: Finite element modeling of blood flow in arteries. *Comp. Meth. Appl. Mech. Engng.* 158, 155–196 (1998).
- [2] Perktold, K., Resch, M., Peter, R.: Three-dimensional numerical analysis of pulsatile flow and wall shear stress in the carotid artery bifurcation. *J. Biomech.* 24(6), 1, 409–420 (1991).
- [3] Rathish Kumar, B. V., Yamaguchi, T., Liu, H., Himeno, R.: A numerical study of 3D unsteady flow dynamics in a doubly constricted vessel. *Int. J. Numerical Methods in Fluids* 38(2), 1159–1172 (2002).
- [4] Rathish Kumar, B. V., Yamaguchi, T., Liu, H., Himeno, R.: Parallel computation of LV hemodynamics. *Riken Review* 40 (2001).
- [5] Formaggia, L., Nobile, F., Quarteroni, A.: A one dimensional model of blood flow: application to vascular prosthesis. In: Babuska, I., Miyoshi, T., Ciarlet, P. G. (eds.) *Mathematical Modeling and Numerical Simulation in Continuum Mechanics*. Vol. 19 of *Lecture Notes in Computational Science and Engineering*, pp. 137–153, Berlin: Springer-Verlag 2002.
- [6] Formaggia, L., Lamponi, D., Quarteroni, A.: One dimensional models for blood flow in arteries. IMA-MOCS Technical Report 03/2002, EPFL, Lausanne, 2002, (submitted to journal of engineering mathematics).
- [7] Sherwin, S. J., Formaggia, L., Peiro, J., Franke, V.: Computational modeling of 1D blood flow with variable mechanical properties and its applications to the simulation of wave propagation in the human arterial system (to appear in *Int. J. Num. Meth. in Fluids*, 2002).
- [8] Formaggia, L., Nobile, F., Quarteroni, A., Veneziani, A.: Multiscale modeling of the circulatory system: a preliminary analysis. *Computing and Visualization in Science* 2, 75–83 (1999).
- [9] Formaggia, L., Gerbeau, J. F., Nobile, F., Quarteroni, A.: On the coupling of 3D and 1D NS equations for flow problems in compliant vessels. *Comp. Methods in Appl. Mech. Engng.* 191, 561–582 (2001).

- [10] Quarteroni, A., Formaggia, L.: Modeling of living system. Chapter on Mathematical Modeling and Numerical Simulation of the Cardio-Vascular System, Handbook of Numerical Analysis, Amsterdam: Elsevier Science 2002 (to appear).
- [11] Brook, B. S., Falle, S. A. E. G., Pedley, T. J.: Numerical solution for unsteady gravity driven flows in collapsible tubes: evolution and roll-wave instability of a steady state. *J. Fluid Mech.* 396, 223–256 (1999).
- [12] Hedstrom, G. W.: Nonreflecting boundary conditions for nonlinear hyperbolic systems. *J. Comp. Physics* 30, 222–237 (1979).
- [13] Thomson, K. W.: Time dependent boundary conditions for hyperbolic systems. *J. Comp. Physics* 68, 1–24 (1987).
- [14] Boukir, K., Maday, Y., Metivet, B.: A high order characteristic method for the incompressible Navier-Stokes equations. *Comp. Meth. Appl. Engng.* 116, 211–218 (1994).
- [15] Donea, J., Giuliani, S., Laval, H., Quartapelle, L.: Time accurate solutions of advection-diffusion problems by finite elements. *Comp. Meth. Appl. Mech. Engng.* 45, 123–145 (1984).
- [16] Quarteroni, A., Valli, A.: Numerical approximation of partial differential equations. Berlin: Springer-Verlag 1994.
- [17] Quartapelle, L.: Numerical solution of the incompressible Navier-Stokes equations. Basel: Birkhäuser Verlag 1993.
- [18] Quarteroni, A., Tuveri, M., Veneziani, A.: Computational vascular fluid dynamics: problems, models and methods. *Computing and Visualization in Science* 2, 163–197 (2000).

B. V. Rathish Kumar  
Alfio Quarteroni  
Luca Formaggia  
Daniele Lamponi  
Institut de Mathematiques (FSB/IMA),  
EPFL, CH-1015, Lausanne, Switzerland

Alfio Quarteroni  
MOX Dipartimento di Matematica,  
Politecnico di Milano, 20133 Milano, Italy

B. V. Rathish Kumar  
Indian Institute of Technology,  
Kanpur, U.P., 208016, India