DTD 5
ARTICLE IN PRESS

Available online at www.sciencedirect.com
SCIENCE DIRECT®

Medical Engineering & Physics xxx (2005) xxx–xxx

Medical Engineering & Physics
www.elsevier.com/locate/medengphy

First-order system least-squares (FOSLS) for modeling blood flow

J.J. Heys [a,∗], C.G. DeGroff [c], T.A. Manteuffel [b], S.F. McCormick [b]

[a] Chemical and Materials Engineering Department, Arizona State University, Tempe, AZ 85287-6006, USA
[b] Department of Applied Mathematics, University of Colorado at Boulder, Boulder, CO 80309, USA
[c] University of Colorado HSC, The Children's Hospital, Denver, CO, USA

Received 19 May 2005; received in revised form 28 September 2005; accepted 4 October 2005

Abstract

The modeling of blood flow through a compliant vessel requires solving a system of coupled nonlinear partial differential equations (PDEs). Traditional methods for solving the system of PDEs do not scale optimally, i.e., doubling the discrete problem size results in a computational time increase of more than a factor of 2. However, the development of multigrid algorithms and, more recently, the first-order system least-squares (FOSLS) finite-element formulation has enabled optimal computational scalability for an ever increasing set of problems. Previous work has demonstrated, and in some cases proved, optimal computational scalability in solving Stokes, Navier–Stokes, elasticity, and elliptic grid generation problems separately. Additionally, coupled fluid–elastic systems have been solved in an optimal manner in 2D for some geometries. This paper presents a FOSLS approach for solving a 3D model of blood flow in a compliant vessel. Blood is modeled as a Newtonian fluid, and the vessel wall is modeled as a linear elastic material of finite thickness. The approach is demonstrated on three different geometries, and optimal scalability is shown to occur over a range of problem sizes. The FOSLS formulation has other benefits, including that the functional is a sharp, a posteriori error measure.

Keywords: Navier–Stokes; Blood flow; Coupled; Finite elements; Least-squares; Multigrid

1. Introduction

Over the past century, many mathematical models for blood flow have been developed [10]. From the earliest 1D models, which were solved analytically, to the present 3D unsteady models, for which only an approximate numerical solution can be obtained, the goal has always been to obtain more accurate models. Further, to avoid the error associated with the introduction of artificial boundary conditions, there is a constant desire to model larger regions of the circulatory system. Unfortunately, present methods for solving the large linear systems of equations associated with the numerical approximation generally do not scale optimally, i.e., the computation costs are not proportional to the number of unknowns but proportional to the number of unknowns to some power greater than 1. However, we have recently shown that the use of a first-order system least-squares (FOSLS) finite-element formulation in conjunction with an algebraic multigrid (AMG) solver is capable of achieving optimal scalability of the computational costs for some 2D fluid–structure problems [8]. The goal of this paper is not to present a new model of blood flow, but to extend this new method and show that it achieves optimal computational scalability on a 3D transient model of blood flow through a compliant vessel.

Modeling blood flow within a compliant vessel wall requires consideration of both the vessel wall domain and the flowing blood domain. As an added complication, the shape of the blood domain is not known a priori to solving the equations and is continually evolving with the current displacement of the vessel wall. An example of a no-flow domain is shown in the upper half of Fig. 1, and this domain is separated into a blood region (Ω_β) and a vessel wall region (Ω_V). The equations for the vessel wall are normally defined from the rest position, so they are based on this no-flow domain. The deformed, flowing blood domain with

∗ Corresponding author.
E-mail addresses: jheys@asu.edu (J.J. Heys), degroff.curt@tchden.org (C.G. DeGroff), tmanteuf@colorado.edu (T.A. Manteuffel), stevem@colorado.edu (S.F. McCormick).

1350-4533/$ – see front matter © 2005 Published by Elsevier Ltd on behalf of IPEM.
doi:10.1016/j.medengphy.2005.10.002

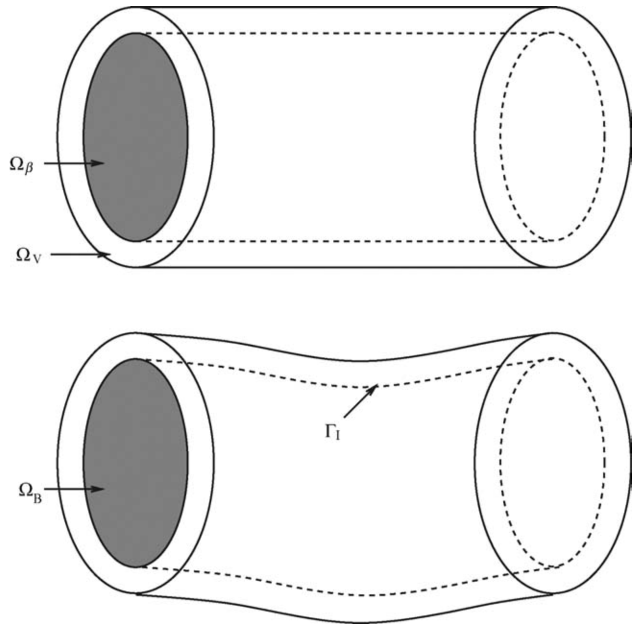

Fig. 1. The no-flow blood (Ω_β) and vessel wall (Ω_V) domains (above) and the deformed blood domain (Ω_B) (below) for a coupled blood vessel system with the initial interface and a deformed interface Γ_I.

displaced vessel wall is shown in the lower half of Fig. 1, and the blood flow region is denoted as Ω_B. The fluid equations are typically defined on this deformed domain, and the interface between the two domains is shown as Γ_I in the deformed case. Mechanical coupling between the domains requires the traction to be continuous along the interface between the blood and vessel wall regions. In the typical case of a non-steady problem, the velocity must also be continuous.

Three important choices must be made when modeling blood flow in a compliant vessel. The first choice is the mathematical model of the vessel wall—both shell models and finite thickness models have been used by others. The second choice is in the iteration approach used to handle the changing blood domain shape—one iteration per time step or multiple iterations. The third choice is the coupling between the three sets of equations, i.e., the coupling between the blood equations (Navier–Stokes), the vessel wall equations, and the remapping or remeshing equations, which handle the changing blood domain shape. We look at each of these choices, beginning with the vessel wall model.

The simplest method is to model the vessel wall as a shell [13,20]. The viscous shear stress is typically ignored [14], resulting in displacement only in the radial direction. An important method related to modeling the vessel wall as a shell is the immersed boundary method developed by Peskin [16], which uses a regular structured grid over the fluid domain, with the elastic boundary expressed in terms of a localized force distribution (Dirac delta functions) within the regular grid. The advantages of the immersed boundary method include the ability to use straightforward finite difference approximations and the computational savings from not having to move the mesh over the fluid domain. The immersed boundary method can have problems with numerical stability if explicit time stepping is used [21], and the use of discrete delta functions prevents the method from achieving more that first-order accuracy [11]. Lee and LeVeque [12] derived a similar method, called the immersed interface method, that overcomes some of the traditional difficulties with the immersed boundary method. The other option is to model the vessel wall as a structure of finite thickness. The choice between these two options depends upon the ratio of vessel wall thickness to vessel diameter. The smaller this ratio, the smaller the error introduced by the shell approximation. For purposes of generality, all models presented in this paper are based on finite vessel wall thickness.

Because the position of the interface and the final shape of the deformed fluid domain are not known a priori, a number of different iterative methods have been developed to handle this moving domain problem. They can loosely be divided into two categories: (1) one iteration per time step approaches or (2) multiple iterations per time step. Unfortunately, both approaches have potential pitfalls because performing multiple iterations may result in slow convergence [7] and higher computational costs, and the single iteration approach may require very small time steps to maintain a stable solution [21]. All simulations in this paper used multiple iterations to ensure that the domain shape was nearly correct for the time step. However, in many cases, the second iteration was not necessary if the time step was sufficiently small. The choice here is highly dependent upon the choice of equation coupling, which is described next.

The third choice that must be made concerns the coupling of the three different sets of equations—blood flow equations, vessel wall equations, and the equations that handle the changing shape of the blood domain. The first option is to solve the equations in series, beginning with the equations describing the blood flow on the current (fixed) domain. The wall stress along the interface from the blood flow solution is then used in the solution of the vessel wall equations to find the new wall displacement. At this point, the shape of the blood flow domain has changed, and additional equations are typically solved to account for the new shape. Depending on the solution approach, the nodes associated with the blood flow discretization are moved, or the new blood flow domain may be mapped back to the original domain. The second option is to couple just the blood flow and vessel wall equations and solve them simultaneously, with the mapping equations solved separately. Finally, it is also possible to solve all of the equations coupled together so that the remapping or remeshing equations are solved simultaneously with the blood flow and vessel wall equations.

The advantage to solving the three parts in series is that this method requires the smallest amount of computer memory, but it often requires multiple iterations per time step to achieve acceptable continuity of traction due to the oscillating convergence sometimes observed [7]. The advantages of coupling the blood flow and vessel wall equations include the assurance of continuity of interface tractions [6] and typically fewer iterations. The latter advantage may stem from the Jacobian matrix containing terms coupling the blood flow and vessel wall domains, thus helping prevent oscillations. However, this coupled system of equations increases memory usage and potentially increases computational costs over solving the three parts in series. For the fully coupled method, the disadvantages include a large memory requirement to store the complex Jacobian matrix [17], while its advantages include potential quadratic convergence near the solution [17]. A comparison between the three options is presented in [8]. In this paper, all simulations were performed using the second approach, that is, by solving the coupled blood flow and vessel wall equations followed by the solving of the remapping equations separately.

The FOSLS methodology has previously been applied to the individual pieces of the coupled model—Navier–Stokes flow, elasticity equations for the vessel wall, and elliptic grid generation (EGG) for remapping the fluid domain. Theoretical results for the Stokes and linear elasticity equations yield optimal discretization error estimates in the H^1 product norm and optimal algebraic convergence [2]. In addition, the FOSLS formulation of EGG, used to map the deforming fluid domain to a reference domain, has been shown to be H^1-elliptic, providing optimal multigrid convergence [4]. The optimality of FOSLS for solving coupled fluid–elastic equations in 2D has also been demonstrated numerically [8]. In summary, FOSLS provides optimal overall convergence properties for each of the three parts of the compliant blood flow system separately, and it has been numerically demonstrated in 2D for fluid–elastic problems. Our aim now is to demonstrate the scalability of the approach on 3D compliant blood flow problems.

2. Model equations and formulation

The blood vessel wall is modeled as a compressible linear elastic solid:

$$-\mu_{\mathrm{V}}\Delta\mathbf{u} - (\lambda + \mu_{\mathrm{V}})\nabla\nabla\cdot\mathbf{u} = 0 \;\text{ in }\; \Omega_{\mathrm{V}}, \tag{1}$$

where μ_{V} and λ are Lamé constants and $\mathbf{u} = (u_1, u_2, u_3)$ is the displacement. Eq. (1) is defined on the original, undeformed domain using a Lagrangian reference frame and can be rewritten in dimensionless form by defining the following the dimensionless variables (indicated by hat symbol (ˆ)):

$$\hat{\boldsymbol{x}} = \frac{\boldsymbol{x}}{L}, \qquad \hat{\mathbf{u}} = \frac{\mathbf{u}}{L},$$

where L is a characteristic length scaling. Multiplying by $\frac{L}{\mu_{\mathrm{V}}}$ yields

$$-\hat{\Delta}\hat{\mathbf{u}} - \left(\frac{\lambda}{\mu_{\mathrm{V}}} + 1\right)\hat{\nabla}\hat{\nabla}\cdot\hat{\mathbf{u}} = 0 \;\text{ in }\; \Omega_{\mathrm{V}}. \tag{2}$$

Blood is modeled using the Navier–Stokes equations:

$$-\rho(\mathbf{v}\cdot\nabla\mathbf{v}) - \nabla p + \mu_{\mathrm{B}}\Delta\mathbf{v} = \rho\frac{\partial\mathbf{v}}{\partial t} \;\text{ in }\; \Omega_{\mathrm{B}}, \tag{3}$$

$$\nabla\cdot\mathbf{v} = 0 \;\text{ in }\; \Omega_{\mathrm{B}}, \tag{4}$$

where ρ is the density, p the pressure, μ_{B} the viscosity, and $\mathbf{v} = (v_1, v_2, v_3)$ is the velocity. Eqs. (3) and (4) are defined for an Eulerian reference frame, which moves as the vessel wall is displaced. The deformed fluid domain is also referred to as the physical domain. The traction matching condition between the regions is

$$\mathbf{n}\cdot\sigma_{\mathrm{V}}(\mathbf{u}) = \mathbf{n}\cdot\sigma_{\mathrm{B}}(\mathbf{v}) \;\text{ on }\; \Gamma_{\mathrm{I}}, \tag{5}$$

where σ_{V} and σ_{B} are the total stress tensors for the vessel wall and flowing blood, respectively, and $\mathbf{n}$ is the outward unit normal vector on the deformed or physical domain interface.

Eqs. (3) and (4) can be rewritten in dimensionless form by defining the following the dimensionless variables (indicated by hat symbol (ˆ)) and number:

$$\hat{\mathbf{v}} = \frac{\mathbf{v}}{\mathcal{V}}, \qquad \hat{p} = \frac{p}{\rho\mathcal{V}^2}, \qquad \hat{t} = \frac{\mathcal{V}t}{L}, \qquad Re = \frac{L\mathcal{V}_\rho}{\mu_{\mathrm{B}}},$$

where $\mathcal{V}$ is a velocity scaling, and Re is the Reynolds number. Using the new variables, Eqs. (3) and (4) can be rewritten as

$$-(\hat{\mathbf{v}}\cdot\hat{\nabla}\hat{\mathbf{v}}) - \hat{\nabla}\hat{p} + Re^{-1}\,\hat{\Delta}\hat{\mathbf{v}} = \frac{\partial\hat{\mathbf{v}}}{\partial\hat{t}} \;\text{ in }\; \Omega_{\mathrm{B}}, \tag{6}$$

$$\hat{\nabla}\cdot\hat{\mathbf{v}} = 0 \;\text{ in }\; \Omega_{\mathrm{B}}, \tag{7}$$

It is important to ensure that the *dimensional* stresses are matched between the fluid and elastic solid. However, the dimensionless variables require the definition of the following dimensionless stresses:

$$\hat{\sigma}_{\mathrm{V}} = (\nabla\hat{\mathbf{u}} + (\nabla\hat{\mathbf{u}})^{\mathrm{T}}) - \left(\frac{\lambda}{\mu_{\mathrm{V}}}\right)(\nabla\cdot\hat{\mathbf{u}})\delta_{ij} = \frac{\sigma_{\mathrm{V}}}{\mu_{\mathrm{V}}}, \tag{8}$$

and

$$\hat{\sigma}_{\mathrm{B}} = Re^{-1}(\nabla\hat{\mathbf{v}} + (\nabla\hat{\mathbf{v}})^{\mathrm{T}}) - \hat{p}\delta_{ij} = \frac{\sigma_{\mathrm{B}}}{\rho\mathcal{V}^2}, \tag{9}$$

where δ_{ij} is the Kronecker delta symbol. Therefore, Eq. (5) can be replaced by

$$\mathbf{n}\cdot\hat{\sigma}_{\mathrm{V}}(\hat{\mathbf{u}}) = \mathbf{n}\cdot\left(\frac{\rho\mathcal{V}^2}{\mu_{\mathrm{V}}}\right)\hat{\sigma}_{\mathrm{B}}(\hat{\mathbf{v}}) \;\text{ on }\; \Gamma_{\mathrm{I}}, \tag{10}$$

thus allowing the use of dimensionless variables with the stress matching conditions. The use of a consistent length scaling, L, between the domains allows the use of a dimensionless position matching condition along the interface,

$$\hat{\mathbf{x}} = \hat{\boldsymbol{\xi}} + \hat{\mathbf{u}} \ \text{ on } \ \Gamma_{\mathrm{I}}, \tag{11}$$

where $\hat{\boldsymbol{\xi}}$ corresponds to the no flow or undeformed coordinates. We drop hat symbol ($\hat{}$) in what follows since only dimensionless variables are considered.

Elliptic grid generation (EGG) is used to map the deformed blood flow region (the physical domain, Ω_{B}) back to the original, undeformed computational region (Ω_β). The EGG equations are derived by requiring that the map be bijective and satisfy

$$\Delta_{\mathbf{x}}\boldsymbol{\xi} = 0 \ \text{ in } \ \Omega_{\mathrm{B}}, \tag{12}$$

where $\boldsymbol{\xi} = (\xi(x, y, z), \eta(x, y, z), \zeta(x, y, z))$ are the undeformed computational coordinates. Eq. (12) is defined on the unknown physical domain, Ω_β, but it can be inverted so that the equation is defined on the computational domain [9]. The solution to the EGG equations allows Eqs. (6) and (7) to be rewritten so that they are defined on the original computational domain instead of the physical domain.

Eqs. (2), (6) and (7) and the inverse of (12) can be recast as a first-order systems of equations by defining new variables. For example, the vessel wall equation (i.e., linear elasticity) requires defining a new 3×3 matrix of variables $U = U_{ij}$ that represent derivatives of the primary variables. Then, rewriting Eq. (2) as a first-order system gives

$$U - \nabla \mathbf{u} = 0 \ \text{ in } \ \Omega_{\mathrm{V}}, \tag{13}$$

$$-(\nabla \cdot U)^{\mathrm{T}} - \left(\frac{\lambda}{\mu_{\mathrm{V}}}\right) \nabla\, \mathrm{tr}(U) = 0 \ \text{ in } \ \Omega_{\mathrm{V}}, \tag{14}$$

$$\nabla \times U = 0 \ \text{ in } \ \Omega_{\mathrm{V}}, \tag{15}$$

where $\mathrm{tr}(U) = U_{11} + U_{22} + U_{33}$. For the first-order system, bold letters indicate a vector, capital letters indicate a second-order tensor, and the shape of zero is implied by the left side. Eq. (15) is added to expose divergence-free errors and to establish H^1-ellipticity [3]. It is important to note that Dirichlet boundary conditions, given by

$$\mathbf{u} = g \ \text{ on } \ \Gamma_{\mathrm{V}}, \tag{16}$$

are now supplemented with the additional, but consistent, tangential boundary condition:

$$\tau \cdot U = 0 \ \text{ on } \ \Gamma_{\mathrm{V}}, \tag{17}$$

where τ is the unit vectors tangential to the surface. Neumann boundary conditions can be rewritten as $\mathbf{n} \cdot U = b$, where $\mathbf{n}$ is the vector normal to the surface and b is the specified flux [3]. The Neumann and Dirichlet conditions for the fluid and EGG equations are also modified in a consistent manner.

The first-order system for the EGG equations can be written as [4]

$$J - \nabla \mathbf{x} = 0 \ \text{ in } \ \Omega_\beta, \tag{18}$$

$$(J^{-\mathrm{T}} J^{-1} \nabla) \cdot J = 0 \ \text{ in } \ \Omega_\beta, \tag{19}$$

$$\nabla \times J = 0 \ \text{ in } \ \Omega_\beta, \tag{20}$$

where $\mathbf{x} = (x(\xi, \eta, \zeta), y(\xi, \eta, \zeta), z(\xi, \eta, \zeta))$ is the mapping from the undeformed computational domain to the deformed physical domain (see Fig. 1), J the Jacobian of the mapping, and $J^{-\mathrm{T}}$ is the inverse of the transpose of the Jacobian. Eq. (19) is nonlinear and illustrates that compliant blood flow problems are always nonlinear in character, either implicitly or explicitly.

Finally, the first-order system for the Navier–Stokes equations (6) and (7) is, after mapping,

$$V - \nabla \mathbf{v} = 0 \ \text{ in } \ \Omega_\beta, \tag{21}$$

$$Re^{-1}(J^{-\mathrm{T}} J^{-1} \nabla) \cdot V - (J^{-1} \nabla p_{\mathrm{s}})^{\mathrm{T}} - (\mathbf{v} \cdot J^{-1} V) = \frac{\partial \mathbf{v}}{\partial t} \ \text{ in } \ \Omega_\beta, \tag{22}$$

$$\nabla\, \mathrm{tr}(J^{-1} V) = 0 \ \text{ in } \ \Omega_\beta, \tag{23}$$

$$(J^{-1} \nabla) \cdot \mathbf{v} = 0 \ \text{ in } \ \Omega_\beta, \tag{24}$$

$$Re^{-1}\, \nabla \times V = 0 \ \text{ in } \ \Omega_\beta. \tag{25}$$

In approximating the solution to this system, Eqs. (23) and (24) can be strictly enforced or weighted more heavily to achieve a result with less error in mass conservation and more error in momentum conservation, which may or may not be desirable.

The dimensionless stress matching condition (10) between the two regions can now be rewritten in terms of first-order variables:

$$J^{-1}\bar{\mathbf{n}} \cdot \left(U + U^{\mathrm{T}} + \frac{\lambda}{\mu_{\mathrm{V}}}\, \mathrm{tr}(U)\delta_{ij}\right) - J^{-1}\bar{\mathbf{n}} \cdot \left(\frac{\rho \mathcal{V}^2}{\mu_{\mathrm{V}}}\right) \times (Re^{-1} J^{-1} V + (Re^{-1} J^{-1} V)^{\mathrm{T}} - p\delta_{ij}) = 0 \ \text{ on } \ \Gamma_{\mathrm{I}}, \tag{26}$$

where $\bar{\mathbf{n}}$ is the outward unit normal vector on the undeformed computational interface. The J^{-1} operator maps $\bar{\mathbf{n}}$ to $\mathbf{n}$, the vector normal to the deformed or physical interface. In many cases, it is possible to compute $\mathbf{n}$ directly, which may be desirable to prevent inaccuracies in J^{-1} from contaminating the traction matching condition.

The construction of the least-squares functional(s) from the system of first-order equations (13)–(26) depend on the method chosen to solve the coupled fluid–elastic problem. For the approach used in this paper, coupling the blood flow and vessel wall equations while leaving the remapping equations as a separate problem, the first functional is given by

$$G(\mathbf{u}, U, \mathbf{v}, V, p) := ||U - \nabla\mathbf{u}||^2_{0,\Omega_V} + \left\| (-\nabla\cdot U)^{\mathrm{T}} - \left(\frac{\lambda}{\mu_V}\right)\nabla\,\mathrm{tr}(U)\right\|^2_{0,\Omega_V} + ||\nabla\times U||^2_{0,\Omega_V} + ||V - \nabla\mathbf{v}||^2_{0,\Omega_\beta}$$
$$+ \left\| Re^{-1}(J^{-\mathrm{T}}J^{-1}\nabla)\cdot V - (J^{-1}\nabla p_s)^{\mathrm{T}} - (\mathbf{v}\cdot J^{-1}V)\frac{\partial\mathbf{v}}{\partial t}\right\|^2_{0,\Omega_\beta} + ||\nabla\,\mathrm{tr}(J^{-1}V)||^2_{0,\Omega_\beta} + ||(J^{-1}\nabla)\cdot\mathbf{v}||^2_{0,\Omega_\beta} + ||Re^{-1}\nabla$$
$$\times V||^2_{0,\Omega_\beta} + \left\| J^{-1}\bar{\mathbf{n}}\cdot\left(U + U^{\mathrm{T}} + \frac{\lambda}{\mu_V}\mathrm{tr}(U)\delta_{ij}\right) - J^{-1}\bar{\mathbf{n}}\cdot\left(\frac{\rho\mathcal{V}^2}{\mu_V}\right)(Re^{-1}J^{-1}V + (Re^{-1}J^{-1}V)^{\mathrm{T}} - p\delta_{ij})\right\|^2_{(1/2),\Gamma_I}, \quad (27)$$

where $||\cdot||^2_{0,\Omega}$ denotes the L_2 norm of the enclosed quantity over the region Ω. J is initially the identity matrix, but calculated for later iterations by first minimizing the following functional:

$$G_{\mathrm{EGG}}(\mathbf{x}, J) := ||J - \nabla\mathbf{x}||^2_{0,\Omega_\beta} + ||(J^{-\mathrm{T}}J^{-1}\nabla)\cdot J||^2_{0,\Omega_\beta}$$
$$+ ||\nabla\times J||^2_{0,\Omega_\beta}. \quad (28)$$

Thus, G is a nonlinear functional that is minimized first, followed by a minimization of G_{EGG}. These minimizations may be repeated to check for convergence. The boundary conditions, other than the traction matching condition, have been omitted from G and G_{EGG} because they can be imposed directly on the finite-element (approximation) space. Alternatively, these boundary conditions can be enforced weakly in a least-squares sense by adding additional terms to the functional. This choice has been shown to have little effect on the final solution, especially as the mesh is refined, but it can affect the convergence rate of the linear solver. The simulations in this paper used strictly enforced boundary conditions unless noted otherwise. In G, L^2 norms are used for the domain and $H^{1/2}$ norms are used for the boundary. In the numerical implementation, L^2 norms scaled by $1/h$ are used for the weak boundary terms.

The equations that are used in the functional (G and G_{EGG}) are first linearized so that the solution can be found using a Gauss–Newton approach. The value of the nonlinear functional is calculated after each Gauss–Newton step to ensure that the nonlinear functional is decreasing to a minimum. The functional for the linearized equations is minimized over the finite-element spaces by setting the derivative to zero in the weak sense for each linearized step. A finite-element basis is then chosen so that the weak form generates a matrix problem. All of the simulations presented in Section 3 use a trilinear finite-element basis for all of the variables. The FOSLS formulation allows the solution spaces for the variables to be chosen independently, with no restrictive stability condition to satisfy. As a result, both the pressure and velocity in the Navier–Stokes equations can be approximated with a trilinear basis. Functionals G and G_{EGG} measure the first derivative of the error in the primary variables (i.e., velocity, pressure, and displacement), unlike the error in the L_2 sense. Therefore, error characterized by 'wiggles' in the solution, which may be small and hidden in the L^2 norm, are large and thus controllable in the functional norm.

Fig. 2 summarizes the many levels of iteration that take place in solving compliant blood flow problems.

- The outermost level consists of cycling between the functional for blood flow and vessel wall, G, and the EGG functional, G_{EGG}, for remapping the blood domain. Typically, only a single outer iteration is required for each time step, but a second iteration can be performed to check convergence.
- Since both functionals in the outer iteration are nonlinear, each individual functional is linearized and at least one iteration is performed. Typically, one iteration is sufficient for small time steps (<0.05 s) and two iterations are sufficient for larger time steps.
- The inner most iterations solve the linear system using an algebraic multigrid (AMG) preconditioner [1,18] for a conjugate gradient (CG) iteration. Under this AMG/CG method, a single V-cycle is used to calculate a preconditioner for a single CG iteration. Most of the computational cost is associated with the V-cycle. Typically, 20–40 AMG/CG iterations provide a sufficiently accurate approximation to the solution of each linear system.

3. Results

The scalings and dimensionless numbers used for all simulations, unless noted otherwise, are summarized in Table 1. These values are calculated based on the assumptions of a kinematic viscosity, υ, for blood of 4×10^{-6} m^2/s, a Young's

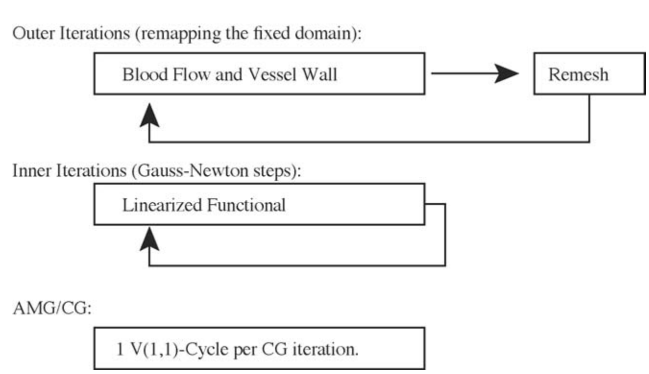

Fig. 2. Summary of the different levels of iteration for modeling compliant blood flow.

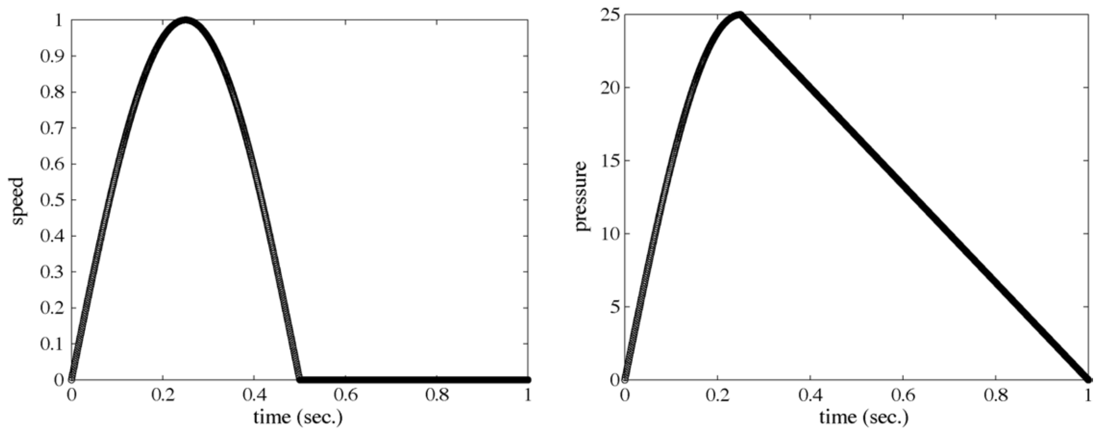

Fig. 3. Maximum dimensionless inlet speed (left) and dimensionless outlet pressure (right) over a single, 1 s cardiac cycle.

Table 1
Scalings and dimensionless parameters used in the model for compliant blood flow. Any changes to these values are explicitly noted

Length scaling, L (m)	0.01
Velocity scaling, $\mathcal{V}$ (m/s)	0.5
Reynolds number, Re	1250
Lamé ratio, $\frac{\lambda}{\mu_{\mathrm{V}}}$	24
Traction matching scaling, $\frac{\rho \mathcal{V}^2}{\mu_{\mathrm{V}}}$	7.4×10^{-3}
Time step scaling, $\frac{\mathcal{V}}{L}$	50

modulus of 1×10^5 Pa for the vessel wall, and a Poisson's ratio of 0.48 for the vessel wall. These parameters are chosen to represent typical conditions found in the major vessels [5]. Our goal in this paper is not to present a new, more accurate model of blood flow in one particular location, but to present a methodology for modeling compliant blood flow in a computationally scalable manner. Therefore, simplicity and generality are the objective in choosing parameters, boundary conditions, and geometries in this section. All calculations were performed on a modest 700 MHz Itanium processor using up to 8 GB memory.

The first test problem is flow through a simple straight tube of length 5.0 and internal diameter 1.0 along the entire axis at rest (no flow). The vessel wall has a thickness of 0.1 (1 mm in dimensional terms), and the ends of the vessel wall are assumed to be fixed. A no-stress condition is imposed on the outer normal surface of the vessel, based on the assumption that the surrounding tissue applies negligible force. The imposing of other, more complicated boundary conditions on the outer tube surface is trivial and does not impact the numerical performance. A parabolic velocity profile is used for the blood flow at the inlet, and the maximum velocity along the inlet is varied using a half-sine wave (Fig. 3). In this way, the velocity is initially zero, increases in a sine wave profile to a maximum at 0.25 s, decreases back to zero at 0.5 s, and remains zero until 1.0 s, at which point a new pulse is begun. Other smooth inlet velocity profiles that we tried exhibited numerical performance similar to what we report below. A no-slip condition is imposed along the vessel wall, and the tangential velocity is also set to zero at both the inlet and outlet. The pressure at the outlet is based on a dimensionless pressure equal to zero at the beginning of each pulse. The pressure rises to 25 in a quarter sine wave pattern at the beginning of the pulse, and then it linearly returns to zero at the end of the pulse (Fig. 3). Both the inlet velocity profile and the outlet pressure profile are qualitatively based on the profile data in Perktold and Rappitsch [14].

The numerical performance of the FOSLS formulation and the AMG/CG solver is summarized in Table 2. In this table, the problem size is varied over an order of magnitude, yet the CPU time is nearly proportional to the number of unknowns. This is clearly seen in the bottom two lines that show a doubling of the number of unknowns and a doubling of the CPU time, i.e., optimal scalability. The convergence factor is the ratio of the value of the residual after the AMG/CG cycle to the value before the cycle. As Table 2 shows, the residual for the coupled functional, blood flow and vessel wall, is decreased by a factor of 0.90 every AMG/CG iteration. The convergence factors for the EGG functional were much lower (approximately 0.4).

The final value of the functional is a sharp measure of the error in the solution as measured in the H^1-norm. Lines 2

Table 2
Numerical performance of the FOSLS finite-element formulation using a AMG/CG solver for the straight tube problem

Average mesh spacing (cm)	Number of unknowns	CPU time per step (min)	Convergence factor	Total functional
0.17	1.07×10^5	17	0.89	7.63×10^{-3}
0.13	2.08×10^5	35	0.87	7.61×10^{-3}
0.085	7.04×10^5	126	0.90	2.50×10^{-3}
0.067	1.39×10^6	258	0.90	2.11×10^{-3}

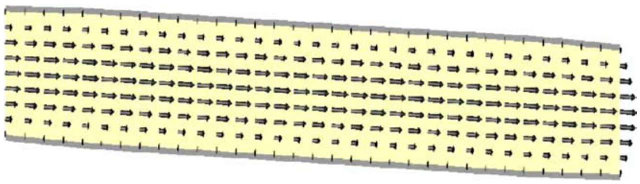

Fig. 4. Velocity profile and vessel wall for the central cross-section of the straight tube test problem at peak velocity.

and 4 represent refinements of lines 1 and 3, respectively, in the axial direction only. Thus, line 3 represents a full refinement, i.e., halving the mesh spacing, of line 1, and line 4 represents full refinement of line 2. Refinement in the axial direction does not significantly reduce the error compared to refinement in the radial and circumferential directions. For the coarsest mesh, axial refinement reduced the error by less than 1% (compare lines 1 and 2), but refinement of that mesh only in the radial and circumferential directions reduced the error by a factor of 3 (compare lines 2 and 3). It is common in blood flow modeling to take advantage of this fact by using less refinement in the axial direction [19]. However, a full refinement of the mesh, i.e., halving the mesh spacing, results in a functional decrease of a factor of more than 3.

The velocity profile for this test problem is shown for the central cross-section in Fig. 4. The flow is laminar, and we did not observe any transitions to turbulence through the flow cycle. The vessel wall is only shown on a single plane along the axis for clarity. The displacement of the vessel wall at the center of the tube is shown in Fig. 5, where the light gray is the at rest position and the overlying dark gray represents the displacement at peak velocity. The wall displacement (compliance) was small (approximately 8–10%) for this particular test, consistent with the results of Perktold et al. [15].

The second test problem is similar to the first, a tube of length 5. However, this tube has a sine wave shape obstruction and thickening of the vessel wall at the midpoint in the axial direction. The tube diameter at the center of the obstruction is 0.6 compared to an unobstructed diameter of 1.0, and the obstruction length is one fifth of the tube length. The mesh sizes and outer dimensions were otherwise identical to the first problem. Table 3 summarizes the computational performance of the FOSLS approach on this test problem. The solution times were slightly slower than the previous example, due to the slightly higher convergence factors, but they still demonstrate optimal scalability for the method. The biggest difference compared to the first example is with

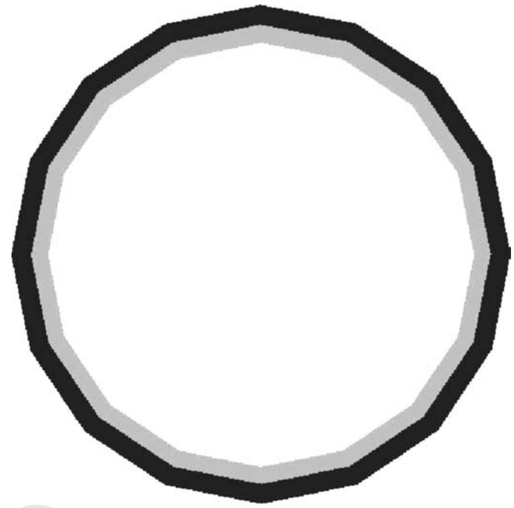

Fig. 5. Displacement of the vessel wall at peak velocity with the at rest position shown in light gray and the overlying dark gray representing the displaced position.

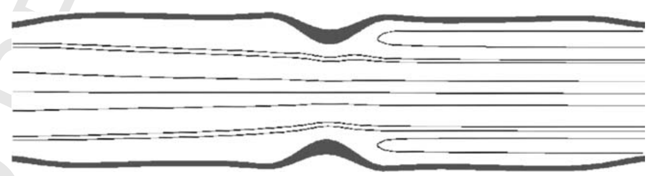

Fig. 6. A cross-section of streamlines for flow down a vessel with a sine shaped obstruction at the peak flow rate. A recirculation forms downstream of the obstruction.

regards to the value of the functional, which is a measure of error in the problem. Clearly, the minimum functional value is not as small using the trilinear basis. The functional value (error) is still going to zero as the mesh is refined, but the use of a higher-order basis could result in less error with little increase in computational costs. Fig. 6 shows the streamlines for flow down the tube with an obstruction. The high Reynolds number results in a recirculation downstream of the obstruction. Fig. 7 shows the displacement of the wall at peak velocity for the tube with obstruction. Again, the vessel wall displacement (approximately 8–12%) is consistent with the results of others.

The final test problem is a single tube, which has a complete semi-circular curve (radius of curvature of 2.667) followed by a straight section. Near the beginning of the straight section is a parabolic shaped obstruction that reduces the tube diameter to 0.5 from 1.0. This represents a diseased state of

Table 3
Numerical performance of the FOSLS finite-element formulation using a AMG/CG solver for the straight tube with obstruction problem

Average mesh spacing (cm)	Number of unknowns	CPU time per step (min)	Convergence factor	Total functional
0.17	1.07×10^5	18	0.90	75.1
0.13	2.08×10^5	36	0.91	66.4
0.085	7.04×10^5	129	0.92	55.3
0.067	1.39×10^6	262	0.92	38.6

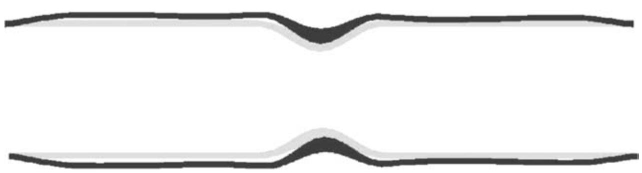

Fig. 7. A cross-section showing vessel wall displacement for the straight tube with obstruction at the peak flow rate. The light gray is the at rest wall position and the darker gray is the deformed vessel wall.

the aorta corresponding to one of the more common types of congenital heart disease termed ‘coarctation of the aorta’ [5]. The tube length is 15 so that changing the length scaling, L, to 1.5 cm results in a geometry that approximates the aorta. The boundary conditions are the same as before, with the maximum inlet velocity being defined by a half sine wave, and the outlet pressure being defined as shown in Fig. 3. For this larger problem, it is not possible to test a large range of mesh sizes on a single processor computer. Basically, the coarsest mesh that accurately represents the geometry results in a linear system with half a million degrees of freedom. As a result, only two different meshes, corresponding to $h \approx 0.1$ and 0.05 were solved. For this most limited of ranges, the solver still seemed to display optimal scaling, and the CPU time per time step was approximately 8 h for the finest grid (4 million degrees of freedom). The results showed a small recirculation downstream of the obstruction only for the inner part of the curve (Fig. 8). Three streamlines in a single plane are also shown in Fig. 8 to illustrate how inertia causes higher flows along the outer curve region of aorta for this particular cross-section. Streamlines in other cross-sections behave differently.

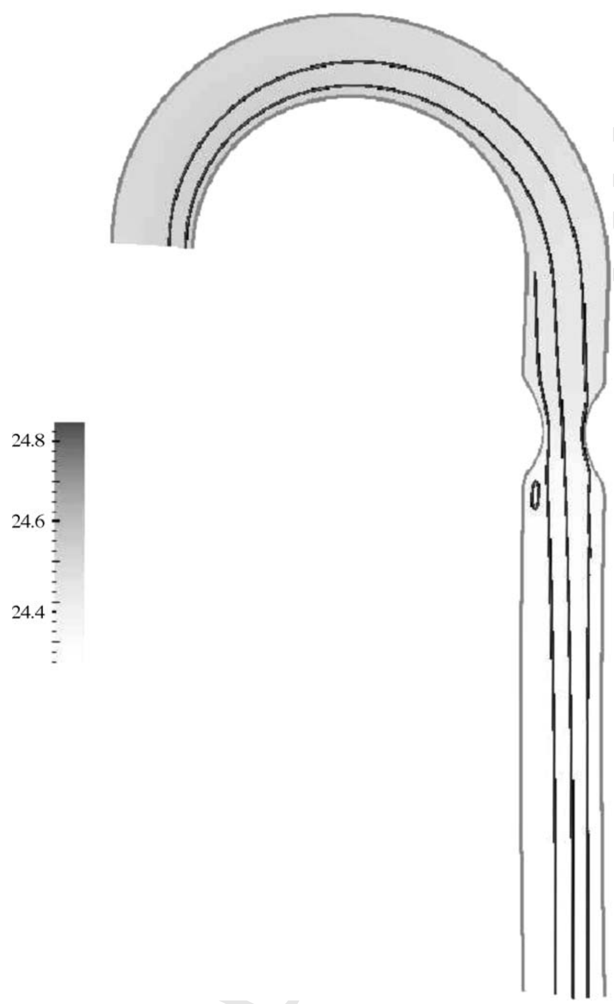

Fig. 8. A central cross-section showing pressure drop and select streamlines for flow through a curved tube with obstruction at peak flow rate. The geometry is approximately that of the aorta.

4. Conclusions

As larger and more complex mathematical models of the vasculature system are developed, the need for scalable algorithms to solve these models will also increase. Even if computational power doubles in 18 months time, only a scalable algorithm can allow a corresponding doubling in the problem size. In this paper, we demonstrated numerically the ability of a FOSLS problem formulation (in conjunction with an AMG/CG) to enable a scalable model of blood flow through a compliant vessel wall. In addition to optimal scalability, the algorithm also provides a sharp error measure in the H^1-norm. The technique provides a great deal of flexibility in that

- the finite-element spaces for each variable may be chosen independently,
- the fluid (blood) and structure (vessel wall) equations may be coupled and solved together or decoupled and solved iteratively, and
- implicit time stepping is probably stable regardless of time step size.

The fact that implicit time stepping is used results in a method that is not well suited for situations in which extremely small time steps must be taken. Further, the use of a lower order basis can result in slow convergence to the actual solution with refinement for some problems. While the focus in this paper was on a very general model and simple geometries, future work will apply the techniques described in this paper to patient specific geometries and more physiologically accurate models.

Acknowledgments

This work was sponsored by the Department of Energy under grant numbers DE-FC02-01ER25479 and DE-FG02-03ER25574, Lawrence Livermore National Laboratory under contract number B533502, Sandia National Laboratory under contract number 15268, and the National Science Foundation under VIGRE grant number DMS-9810751.

References

[1] Brandt A. Multi-level adaptive solutions to boundary value problems. Math Comput 1977;31:333–90.

[2] Cai Z, Manteuffel T, McCormick S. First-order system least squares for the Stokes equations, with application to linear elasticity. SIAM J Numer Anal 1997;34(5):1727–41.

[3] Cai Z, Mantueffel T, McCormick S. First-order system least squares for second-order partial differential equations, Part ii. SIAM J Numer Anal 1997;34(2):425–545.

[4] Codd A, Manteuffel T, McCormick S, Ruge JW. Multilevel first-order system least squares for elliptic grid generation. SIAM J Numer Anal 2003;41(6):2210–32.

[5] DeGroff C, Orlando W, Shandas R. Insights into the effect of aortic compliance on Doppler diastolic flow patterns seen in coarctation of the aorta: a numerical study. J Am Soc Echocardiogr 2003;16:162–9.

[6] Ghattas O, Li X. A variational finite element method for stationary nonlinear fluid–solid interaction. J Comput Phys 1995;121:347–56.

[7] Heil M. Stokes flow in an elastic tube—a large-displacement fluid–structure interaction problem. Int J Numer Meth Fluids 1998;28: 243–65.

[8] Heys J, Manteuffel T, McCormick S, Ruge J. First-order systems least squares (FOSLS) for coupled fluid–elastic problems. J Comput Phys 2004;195(2):560–75.

[9] Knupp P, Steinberg S. Fundamentals of grid generation. Boca Raton, FL: CRC Press; 1993.

[10] Ku D. Blood flow in arteries. Annu Rev Fluid Mech 1997;29: 399–434.

[11] Lai M-C, Peskin C. An immersed boundary method with formal second-order accuracy and reduced numerical viscosity. J Comput Phys 2000;160:705–19.

[12] Lee L, LeVeque R. An immersed interface method for incompressible Navier–Stokes equations. SIAM J Sci Comput 2003;25:832–56.

[13] Perktold K, Rappitsch G. Mathematical modeling of local arterial flow and vessel mechanics. In: Crolet J, Ohayon R, editors. Computational methods for fluid–structure interaction, vol. 306: Pitman research notes in mathematics series. New York: Longman Scientific and Technical/Wiley; 1994. p. 230–45.

[14] Perktold K, Rappitsch G. Computer simulation of local blood flow and vessel mechanics in a compliant carotid artery bifurcation model. J Biomech 1995;28(7):845–56.

[15] Perktold K, Thurner E, Kenner T. Flow and stress characteristics in rigid walled and compliant carotid artery bifurcation models. Med Biol Eng Comput 1994;32:19–26.

[16] Peskin C. Numerical analysis of blood flow in the heart. J Comput Phys 1977;25:220–52.

[17] Rast M. Simultaneous solution of the Navier–Stokes and elastic membrane equations by a finite element method. Int J Numer Meth Fluids 1994;19:1115–35.

[18] Ruge J, Stüben K. Algebraic multigrid. In: McCormick SF, editor. Multigrid methods, vol. 3: Frontiers in applied mathematics. Philadelphia: SIAM; 1987. p. 73–130.

[19] Shahcheraghi N, Dwyer H, Cheer A, Barakat A, Rutaganira T. Unsteady and three-dimensional simulation of blood flow in the human aortic arch. J Biomech Eng 2002;124:378–87.

[20] Tang D, Anderson D, Biz S, Ku D. Steady viscous flow in constricted elastic tubes subjected to a uniform external pressure. Int J Numer Meth Eng 1998;41:1391–415.

[21] Tu C, Peskin C. Stability and instability in the computation of flows with moving immersed boundaries: a comparison of three methods. SIAM J Sci Statist Comput 1992;13(6):1361–76.