Efficient Simulation of Blood Flow Past Complex Endovascular Devices Using an Adaptive Embedding Technique

Juan R. Cebral* and Rainald Löhner

***Abstract*—The simulation of blood flow past endovascular devices such as coils and stents is a challenging problem due to the complex geometry of the devices. Traditional unstructured grid computational fluid dynamics relies on the generation of finite element grids that conform to the boundary of the computational domain. However, the generation of such grids for patient-specific modeling of cerebral aneurysm treatment with coils or stents is extremely difficult and time consuming. This paper describes the application of an adaptive grid embedding technique previously developed for complex fluid structure interaction problems to the simulation of endovascular devices. A hybrid approach is used: the vessel walls are treated with body conforming grids and the endovascular devices with an adaptive mesh embedding technique. This methodology fits naturally in the framework of image-based computational fluid dynamics and opens the door for exploration of different therapeutic options and personalization of endovascular procedures.**

***Index Terms*—Adaptive embedded unstructured grids, cerebral aneurysms, coiling, computational fluid dynamics, endovascular devices, stenting.**

I. Introduction

INTRACRANIAL aneurysms are pathological dilatations of the cerebral arteries, which occur mostly in curved arteries and at arterial bifurcations in the circle of Willis [1]–[3]. The most serious complication is the rupture of intracranial aneurysms and the associated subarachnoid hemorrhage which is often fatal [4]–[6]. The most common methods for treatment of cerebral aneurysms can be divided into surgical and endovascular procedures [7]. Surgical clipping consists in placing a metallic clip across the neck of the aneurysm in order to block the blood flow into the aneurysm. There are high risks associated with these surgical procedures, and in many instances the clipping is very difficult and sometimes impossible [8]. Endovascular procedures consist in the implantation of intravascular devices to limit the flow of blood into the aneurysm and promote thrombus formation in the sac, while preventing distal thromboembolism into the parent artery. The most used method to treat aneurysms is coiling, which consists in packing platinum coils in the aneurysm sac to stop the flow. The second alternative used mainly in cases that are difficult to coil due to the shape of the aneurysm is stenting, which consists in placing a stent in the parent vessel to limit the flow into the sac. Sometimes a combination of coiling and stenting to hold the coils inside the aneurysm sac are used.

In vitro and numerical models have shown that several parameters such as the size and amount of coils packed into an aneurysm or the porosity and wire dimension of stents used to alter the flow into an aneurysm are very important [9]–[12]. Unfortunately, these idealized models cannot be used in a straightforward manner for patient evaluation and planning of the endovascular procedures. In contrast, personalized or patient-specific computational models of cerebral aneurysms can be constructed from medical image data [8], [13]–[16]. However, the main difficulty lies in the construction of appropriate computational grids around the endovascular devices. Two types of grids are most commonly used for computational fluid dynamics (CFD) simulations: body-conforming and embedded. For body-conforming grids the external mesh faces match up with the surface (vessel walls, surfaces of the endovascular devices) of the domain. This is not the case for the embedded approach (also known as ficticious domain, immersed boundary or Cartesian method), where the surface is placed inside a large mesh (typically a regular parallelepiped), with special treatment of the elements close to the surfaces. Considering the general case of moving or deforming surfaces, both approaches have complementary strengths and weaknesses:

1) *Body-Conforming Moving Meshes*: the partial differential equations (PDEs) describing the flow need to be cast in an arbitrary Lagrangian-Eulerian (ALE) frame of reference and the mesh is moved at every timestep in such a way as to minimize distortion. If required, a complete or partial mesh regeneration is carried out and the solution is re-interpolated. All of these steps have been optimized over the course of the last decade, and this approach has been used extensively [17]. The body-conforming solution strategy exhibits the following shortcomings: the description of the surface of the computational domain must be given by a watertight assembly of analytical or discrete patches. For complex geometries and complex endovascular devices (e.g., coils), it can be a tedious process to construct such a surface description. The interpolation required between grids invariably leads to some loss of information. Additionally, there is an extra cost associated with the recalculation of geometry, wall-distances and mesh velocities as the mesh deforms.
2) *Embedded Fixed Meshes*: the mesh is not body-conforming and does not move. Hence, the PDE's describing

Manuscript received April 27, 2004; revised December 31, 2004. The work of J. R. Cebral was supported in part by the Whitaker Foundation and in part by Philips Medical Systems. *Asterisk indicates corresponding author.*

*J. R. Cebral is with the School of Computational Sciences, George Mason University, 4400 University Drive, MSN 4C7, Fairfax, VA 22030 USA (e-mail: jcebral@gmu.edu).

R. Löhner is with the School of Computational Sciences, George Mason University, Fairfax, VA 22030 USA.

Digital Object Identifier 10.1109/TMI.2005.844172

0278-0062/$20.00 © 2005 IEEE

the flow can be left in the simpler Eulerian frame of reference. At every timestep, the edges of the tetrahedral elements of the CFD grid (CFD edges) that are crossed by embedded objects are identified and proper boundary conditions are applied in their vicinity. While used extensively [18], [19] the embedding solution strategy also exhibits some shortcomings: the boundary, which has the most profound influence on the ensuing physics, is also the place where the worst elements are found; at the same time, a coarse representation of the boundary is used; no stretched elements can be introduced to resolve boundary layers; adaptivity is essential for most cases; and there is an extra cost associated with the recalculation of geometry (when adapting) and the crossed edge information.

Experience over the last three years [20]–[22] has shown that while allowing a higher degree of automation, the embedded approach requires higher grid resolution. For this reason, in the present work for the first time a compromise solution is advocated: the external walls of the vessels are treated using body-conforming methods, while the complex endovascular devices are treated using the embedded approach.

The basic idea of our approach is schematically illustrated in Fig. 1. Briefly, a model of the vasculature is constructed (e.g., from anatomical images) and meshed [Fig. 1(a)]. The edges of the elements of this grid that are cut by the surface of the endovascular device are then removed from the flow calculation and appropriate boundary conditions are applied [Fig. 1(b)]. In order to increase the mesh resolution and accuracy, the grid is adaptively refined in the vicinity of the surface of the endovascular device [Fig. 1(c)].

However, as "image to body-conforming mesh" has matured in recent years, a procedure of this kind allows for a high degree of automation, making it accessible for optimization schemes and nonexpert users. A second, less obvious but equally important advantage of the embedded approach is that the description of the embedded endovascular devices does not have to be exact. If we consider coils in particular, it is very difficult to describe analytically the twists and turns. A much simpler way to describe a coil is via beads of spheres, allowing for automatic coil generation. The same applies to stents and other embedded endovascular devices.

This paper describes the application of an adaptive grid embedding technique for the efficient simulation of blood flows past geometrically complex endovascular devices. Since these techniques do not require analytical representations of the device geometry, they fit naturally in the framework of patient-specific modeling of hemodynamics from medical images. The methodology, which is at a preliminary stage, is demonstrated in the context of coiling and stenting of cerebral aneurysms.

II. Methods

A. Vascular Models

Personalized models of blood vessels are constructed from three–dimensional (3-D) rotational angiography (3DRA) images [15], [16]. Briefly, the images are first smoothed using a combination of blurring and sharpening operations followed by a region growing segmentation and iso-surface extraction. This surface is then used to initialize a deformable model under the action of internal smoothing forces and external forces from the gradients of the original un-processed images. The model is then smoothed, the vessels are interactively truncated and the boundaries are extruded in order to minimize the effects of boundary conditions in the region of the aneurysm. The anatomical model is then used as a support surface to generate a finite element grid using an advancing front method that first re-triangulates the surface and then marches into the domain generating tetrahedral elements. An element size distribution appropriate for CFD calculations is prescribed using source functions and adaptive background grids [23], [24].

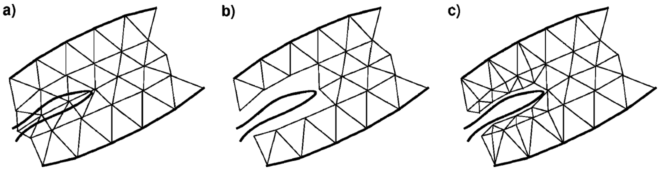

Fig. 1. Modeling of vessels using body conforming grids and endovascular devices using mesh embedding approach. (a) Body-fitted grid for the vessel and superposed endovascular device, (b) first-order approximation of endovascular device using embedding technique, (c) first-order embedding with mesh refinement close to the device.

B. Device Models

1) Description of Devices: The description of the medical devices embedded into arterial blood flows can be performed in a variety of ways. The highest level of precision would be obtained by an exact description of the surfaces via non-uniform rational B-Spline surfaces (NURBS). While desirable, such a description would place a high demand on user input time. A far simpler description is obtained by assembling spheres into tightly packed beads that describe the medical devices. As the number of spheres grows, the approximation to the exact surface of the medical device improves, and can be made arbitrarily accurate. Coils, whose final placement is driven by complex interactions between the coils and the vessel wall, and whose function is mainly the blocking and adhesion of blood, are described very well by this method. The same applies to stents.

2) Generation of Coils: The generation of coils is performed using an advancing front method for the generation of arbitrary objects [25], [26]. Objects can be simply points, spheres, assemblies of spheres (spheres forming tetrahedra, ellipsoids, tubes or coils), or a watertight, closed assembly of triangles. As in traditional grid generators for unstructured grids, the mean distance between objects is specified by means of background grids and sources [27]. In order not to generate objects outside the computational domain, a triangulation of the surface that is to be filled with objects is generated. The size of this triangulation is compatible with the desired mean distance between objects specified by the user. Starting from this initial 'front' of objects, new objects are added, until no further objects can be introduced. Whereas the advancing front technique for the generation of volume grids removes one face at a time to generate elements, the advancing front technique for filling up space with arbitrary objects removes one object at a time, attempting to introduce as many objects as possible in its immediate neighborhood.

The 'core forming block' for coils is given by an assembly of spheres that are packed together along a circular arc. Typically, a 60° segment is taken. We will denote this 'core forming block' as coillet. For every possible location where a new coillet may be introduced the following operations are required:

- generate a coillet whose origin starts at the present location;
- if the coillet crosses any other coillets: reject it;
- if the coillet crosses any boundaries: reject it;
- introduce the coillet into the list of generated coillets;
- Add the coillet end-points to the list of possible locations for further coillet introduction.

Up to 20 attempts with varying, randomly generated orientations are made to introduce a new coillet at a given location. If no coillet could be introduced, the next available end-point of coillets is queried. The procedure stops when either:

- no available end-point of coillets are present;
- the desired total length of the coil has been reached; or
- The desired volume fill ratio has been reached.

In order to obtain continuous coils, the end-point of the last generated coillet is used as the next location to introduce a new coillet. Moreover, during the first 5 attempts at introducing the next coillet, an orientation that is somewhat aligned with the end of the last coillet is chosen. Fig. 2 illustrates the procedure for a small two-dimensional example.

The use of optimal data structures, such as heap-lists, octrees and linked lists [27] makes it possible to fill aneurysms with coils described by $10^5 - 10^6$ spheres in less than a minute on a typical PC.

C. Blood Flow Models

Blood flow is mathematically modeled by the unsteady Navier-Stokes equations for an incompressible fluid [28]

$$\nabla \cdot \mathbf{v} = 0 \tag{1}$$

$$\rho\left(\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v}\cdot\nabla\mathbf{v}\right) = -\nabla p + \nabla\mu\cdot\nabla\mathbf{v} \tag{2}$$

where ρ is the density, $\mathbf{v}$ is the velocity field, p the pressure, and μ the viscosity. For simplicity, a Newtonian blood viscosity model is used and the vessel walls are assumed rigid. Physiologic pulsatile velocity boundary conditions are imposed using a superposition of Womersley solutions for each Fourier mode of the flow rate curves [29].

D. Numerical Schemes

Over the years, several numerical schemes have been used to solve the incompressible Navier-Stokes equations (1), (2) [30]–[32]. What sets incompressible flow solvers apart from compressible flow solvers is the fact that the pressure is not obtained from an equation of state but from the divergence constraint. In other words, the pressure establishes itself instantaneously and must therefore be integrated implicitly [27]. The equation for the pressure can be found by taking the divergence of the momentum equation (2)

$$\nabla^2 p = -\nabla\cdot(\mathbf{v}\cdot\nabla\mathbf{v}). \tag{3}$$

The hyperbolic character of the advection operator and the elliptic character of the pressure-Poisson equation have led to a number of so-called projection schemes. Our methodology is based on this approach and will be described in detail below. The basic idea of fractional step or pressure projection methods is to predict first a velocity field from the current flow variables without taking into account the divergence constraint. In a second step, the divergence constraint is enforced by solving a Poisson equation for the pressure (3). An explicit scheme is composed of the following stages [33].

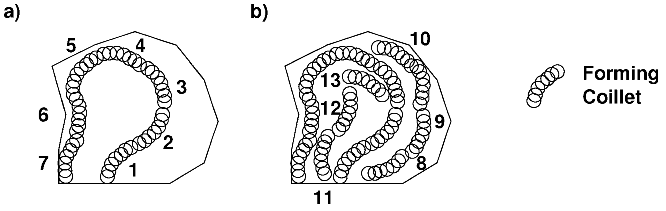

Fig. 2. Schematics of the advancing front algorithm for filling space with ellipses. (a) Coil-generated after insertion of 7 segments and (b) coils after 13 segments.

Advective-diffusive prediction: $\mathbf{v}^n \rightarrow \mathbf{v}^*$

$$\left(\frac{1}{\Delta t} - \nabla\mu\cdot\nabla\right)(\mathbf{v}^* - \mathbf{v}^n) + \mathbf{v}^n\cdot\nabla\mathbf{v}^n + \nabla p^n = \nabla\mu\cdot\nabla\mathbf{v}^n. \tag{4}$$

Pressure correction: $p^n \rightarrow p^{n+1}$

$$\nabla\cdot\mathbf{v}^{n+1} = 0 \tag{5}$$

$$\frac{\mathbf{v}^{n+1} - \mathbf{v}^*}{\Delta t} + \nabla(p^{n+1} - p^n) = 0 \tag{6}$$

which results in a Poisson equation for the pressure:

$$\nabla^2(p^{n+1} - p^n) = \frac{\nabla\cdot\mathbf{v}^*}{\Delta t}. \tag{7}$$

Velocity correction: $\mathbf{v}^* \rightarrow \mathbf{v}^{n+1}$

$$\mathbf{v}^{n+1} = \mathbf{v}^* - \Delta t\nabla(p^{n+1} - p^n). \tag{8}$$

It is well known that a straightforward Galerkin finite element approximation of the advection terms leads to unstable numerical schemes. Therefore, the equations are discretized in space using an edge-based upwind finite element approximation [33], [34]. The discretized momentum equation is solved using a generalized minimal residual (GMRES) solver, while a diagonal preconditioned conjugate gradient solver is used for the pressure Poisson equation [35].

Explicit integration of the advective terms implies that information can travel at most one element per timestep. In order to allow for faster transfer of information and larger timesteps, the advective terms have to be integrated implicitly. In this case, (4) becomes [33]

$$\left(\frac{1}{\Delta t} + \mathbf{v}^*\cdot\nabla - \nabla\mu\cdot\nabla\right)(\mathbf{v}^* - \mathbf{v}^n) + \mathbf{v}^n\cdot\nabla\mathbf{v}^n + \nabla p^n = \nabla\mu\cdot\nabla\mathbf{v}^n \tag{9}$$

which results in a nonsymmetric system of equations of the form

$$\mathbf{A}\cdot\Delta\mathbf{v} = \mathbf{r}. \tag{10}$$

This system can be written as

$$\mathbf{A}\cdot\Delta\mathbf{v} = (\mathbf{L} + \mathbf{D} + \mathbf{U})\cdot\Delta\mathbf{v} = \mathbf{r} \tag{11}$$

where $\mathbf{L}$, $\mathbf{D}$, $\mathbf{U}$ denote the lower, diagonal and upper diagonal entries of $\mathbf{A}$. This system is solved using a lower-upper symmetric Gauss–Seidel (LU-SGS) relaxation scheme

$$(\mathbf{L} + \mathbf{D})\cdot\mathbf{D}^{-1}\cdot(\mathbf{D} + \mathbf{U})\cdot\Delta\mathbf{v} = \mathbf{r}. \tag{12}$$

This relaxation scheme has been optimized over the years, resulting in very efficient matrix free solvers [36], [37]. Although this scheme can substantially accelerate the convergence of steady state problems, the timestep is still constrained for unsteady problems. The maximum Courant number that can be used with this scheme is approximately 5. The result is that for typical hemodynamics simulations several tens or hundreds of thousands of timesteps are required per cardiac cycle. The alternative is to use fully implicit time integration schemes. These schemes allow arbitrary timestep sizes, and Courant numbers of the order of 100 are typically used for hemodynamic calculations. The simplest of such schemes can be written in the following form [33]:

$$\frac{\mathbf{v}^\theta - \mathbf{v}^n}{\theta \Delta t} + \mathbf{v}^\theta \cdot \nabla \mathbf{v}^\theta + \nabla p^\theta = \nabla \mu \cdot \nabla \mathbf{v}^\theta \tag{13}$$

$$\nabla \cdot \mathbf{v}^\theta = 0. \tag{14}$$

When $\theta = 1$ the first-order backward Euler scheme is recovered, while for $\theta = 0.5$ the second-order Cranck–Nicholson scheme is obtained. Moving the first term to the right-hand side, this scheme can be interpreted as the steady-state solution of the pseudotime system

$$\frac{\partial \mathbf{v}^\theta}{\partial \tau} + \mathbf{v}^\theta \cdot \nabla \mathbf{v}^\theta + \nabla p^\theta = \nabla \mu \cdot \nabla \mathbf{v}^\theta - \frac{\mathbf{v}^\theta - \mathbf{v}^n}{\theta \Delta t}. \tag{15}$$

This equation has the same form as the original Navier-Stokes equation (2) but with a source term on the right-hand side. The solution is then advanced by solving a steady-state problem in pseudotime τ at each timestep. These steady state problems were solved using the implicit advection scheme of (9) with a local Courant number of 5 to compute the pseudotimesteps, and the LU-SGS relaxation scheme of (12). The calculations presented here were obtained with $\theta = 1$.

E. Embedded Endovascular Devices

1) Treatment of Embedded Objects/Surfaces: Two basic approaches have been proposed to modify field solvers in order to accommodate embedded surfaces: force-based and kinematics-based. The first type applies an *equivalent balancing force* to the flowfield in order to achieve the kinematic boundary required at the embedded surface [19], [38]. The second approach, followed here, is to apply *kinematic boundary conditions* at the nodes close to or inside the embedded objects/surfaces.

Given the embedded objects/surfaces and the CFD mesh, the edges of the tetrahedral elements of the CFD grid (CFD edges) that are either cut or inside embedded objects/surfaces are found and deactivated. Considering an arbitrary field point i, the time-advancement of the unknowns $\mathbf{u}^i$ for an explicit edge-based time integration scheme [27] is given by

$$M^i \Delta \mathbf{u}^i = \Delta t \sum_{ij\Omega} C^{ij}(F_i + F_j). \tag{16}$$

Here, C, F, and M denote, respectively, the edge-coefficients, fluxes, and mass-matrix. For any edge ij that has been eliminated, the coefficients C^{ij} are set to zero. This implies that for a uniform state $\mathbf{u} = const.$ the balance of fluxes for interior points with cut edges will not vanish. This is remedied by defining a new boundary point to impose total/normal velocities, as well as adding a "boundary contribution," resulting in

$$M^i \Delta \mathbf{u}^i = \Delta t \left[\sum_{ij\Omega} C^{ij}(F_i + F_j) + C^i_\Gamma F_i \right]. \tag{17}$$

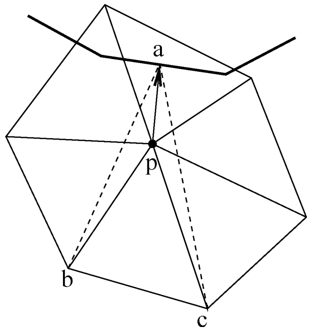

Fig. 3. Boundary conditions for end points of cut edges: velocities at a boundary point p are interpolated from neighboring points b and c and the closest point on the device surface a.

The point-coefficients C^i_Γ are obtained from the condition that $\Delta \mathbf{u} = 0$ for $\mathbf{u} = const.$ Given that the gradients $(\mathbf{g})$ of the unknowns (e.g., for limiting) are also constructed using a loop of the form given by (16) as:

$$M^i \mathbf{g}^i = \sum_{ij\Omega} C^{ij}(u_i + u_j) \tag{18}$$

it would be desirable to build the C^i_Γ coefficients in such a way that the constant gradient of a linear function u can be obtained exactly. However, this is not possible, as the number of coefficients is too small. Therefore, the gradients at the boundary are either set to zero or extrapolated from the interior of the domain.

The mass-matrix M^i of points surrounded by cut edges must be modified to reflect the reduced volume due to cut elements. The simplest possible modification of M^i is used. In a pass over the edges, the smallest 'cut edge fraction' ξ for all the edges surrounding a point is found. The modified mass-matrix is then given by

$$M^i_* = \frac{1 + \xi_{min}}{2} M^i. \tag{19}$$

Note that the value of the modified mass-matrix can never fall below half its original value, implying that timestep sizes will always be acceptable.

For the end-points of edges that have been eliminated, the usual Navier-Stokes boundary conditions are applied, i.e., the velocity is set to the velocity of the embedded object/surface (zero if the object is at rest). Alternatively, for each of the boundary points p, the closest point a on the surface of the device is found. Then a tetrahedral element is formed by connecting a with three neighbor points of p that contains p. The velocity at the location of p is then linearly interpolated from the nodes of this tetrahedron and setting $v = v_{wall}$ at a. The imposition of boundary conditions at points of crossed edges is schematically illustrated in Fig. 3 in two dimensions. In order to obtain the wall shear stress at the surface of the device, the gradient of the velocity field is computed everywhere in the fluid domain. Then normal gradients are extrapolated to the points belonging to cut edges. The full shear tensor is then constructed from this information, and using the surface normals, this tensor is then transformed to a shear force on the wall. Finally, in order to increase the efficiency of the algorithm, points that are completely surrounded by edges that have been eliminated are automatically deactivated. Further details on these procedures may be found in [22].

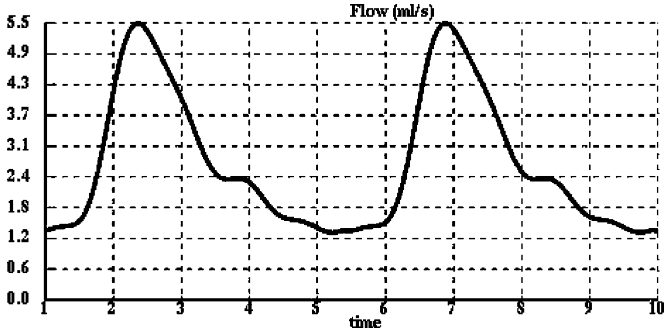

Fig. 4. Flow waveform (in milliliters/second) used to impose boundary conditions for the idealized stented aneurysm model. The time axis is shown in arbitrary units, the heart rate was 60 bpm.

2) Adaptive Mesh Refinement: Adaptive mesh refinement is very often used to reduce CPU and memory requirements without compromising the accuracy of the numerical solution. For transient problems with moving discontinuities, adaptive mesh refinement has been shown to be an essential ingredient of production codes [20], [21], [39]. For embedded objects/surfaces, the mesh in their vicinity can be automatically refined. This has been done in the present case by including two additional refinement indicators (in addition to the usual ones based on the flow variables). The first one looks at the edges cut by embedded objects/surfaces, and refines the mesh to a certain element size or refinement level. The second, more sophisticated indicator, looks at the surface curvature of the embedded objects/surfaces, and refines the mesh only in regions where the element size is deemed insufficient.

III. Results

A. Stenting of Idealized Aneurysm Model

The blood flow in an idealized aneurysm model was computed. The geometry and flow conditions were taken from [40]. The flow waveform is shown in Fig. 4. The geometric model was constructed by intersecting the triangulations of a straight circular cylinder representing the parent vessel, and the triangulation of a sphere representing the aneurysm. The merging was done with an adaptive volumetric technique [41]. The cylinder was 0.35 cm in diameter and 10 cm in length, while the sphere was 0.5 cm in diameter and was displaced 0.26 cm from the axis of the cylinder.

The stent was modeled as 12 helical structures evenly placed around the vessel cross section, covering the neck of the aneurysm. The thickness of the stent was 0.01 cm and its length was 0.5 cm. These cylindrical structures were discretized as a series of overlapping spheres. The radius of the spheres was equal to the stent thickness, and they were separated by half the radius, i.e., the packing factor was $0.5R$ where R is the radius.

The different model components are shown in Fig. 5: verlapping triangulations of the aneurysm and vessel [Fig. 5(a)], vascular model after merging [Fig. 5(b)], model of the stent composed of overlapping spheres [Fig. 5(c), and model of the vessel with the embedded stent [Fig. 5(d)].

A finite element grid with a uniform size distribution of about 0.02-cm resolution was generated. A second finer grid was generated by increasing the resolution to 0.007 cm in the region of the stent. A third grid was generated by adaptively refining the elements of the coarser mesh in the vicinity of the stent. Two levels of h-refinement were used. These meshes, which we denote grids 1, 2, and 3, contained roughly 0.9, 4.1, and 1.5 million elements, respectively.

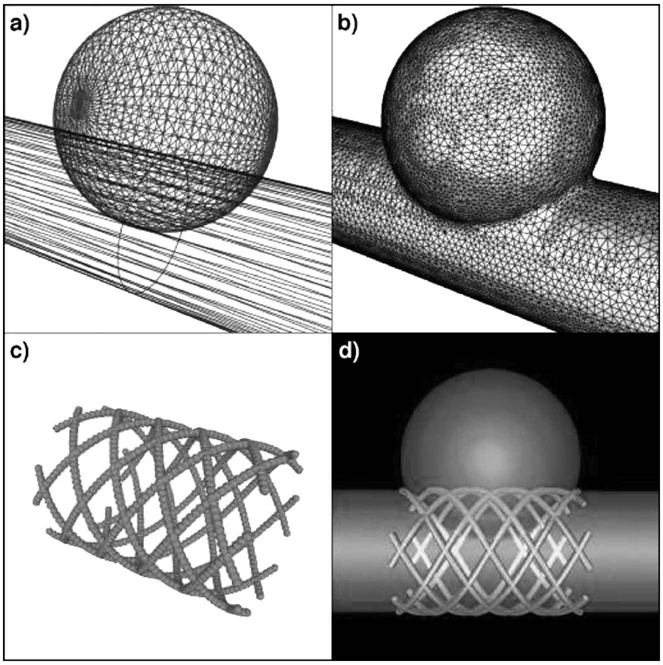

Fig. 5. Components of an idealized model of a stented aneurysm. (a) Aneurysm and vessel, (b) vascular model after merging of aneurysm and vessel, (c) stent model, and (d) complete model of the stented aneurysm.

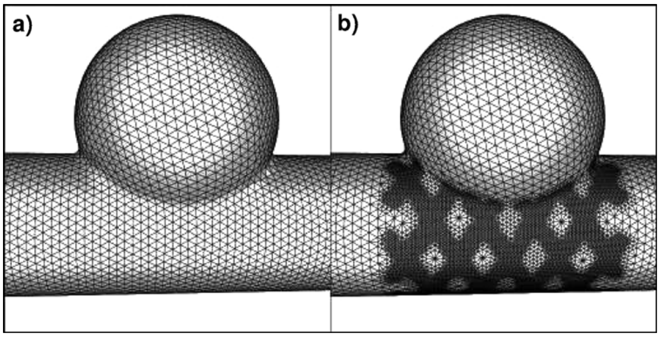

Fig. 6. Surface of the finite element grids. (a) Coarse mesh and (b) mesh adaptively refined close to the embedded stent.

The surfaces of the coarse and adaptively refined finite element grids are shown in Fig. 6(a) and (b), respectively.

Pulsatile flow simulations were performed on both the prestenting and poststenting configurations using grids 1, 2, and 3. Each of these calculations was carried out in shared-memory parallel mode on 6 processors of a Silicon Graphics Altix with 14 1.4-GHz Itanium processors and 14 GB of RAM. They required a total of 6, 54, and 12 h to run two complete cardiac cycles, respectively. The computed flow patterns using the fine mesh (2) and the adaptively refined mesh (3) closely match, however the adaptively refined mesh contains much less elements and is therefore more efficient.

Visualizations of the computed blood flow patterns are shown in Fig. 7. Contours of velocity magnitude in the middle plane are shown at peak systole for the aneurysm before stenting [Fig. 7(a)] and after stenting [Fig. 7(b)]. The stopping effects of the stent are evident in these visualizations.

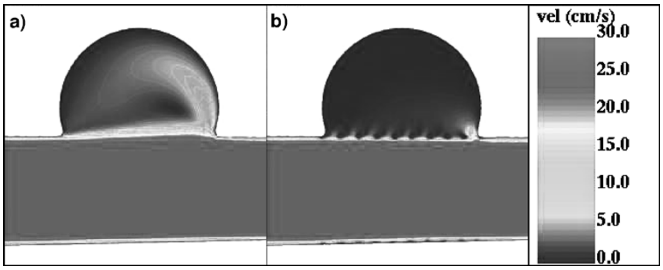

Fig. 7. Visualization of velocity maps at peak systole on a planar cut through the symmetry axis of the vascular model (a) before stenting and (b) after stenting using adaptively refined grid.

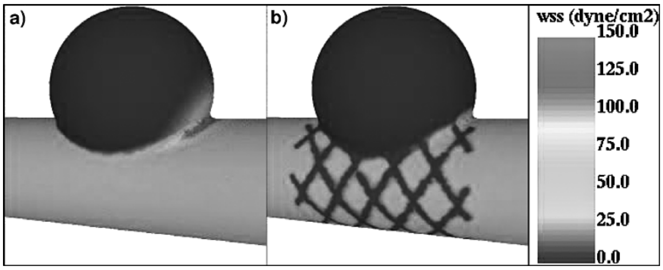

Fig. 9. Visualization of the wall shear stress magnitude at peak systole: (a) before stenting, (b) after stenting.

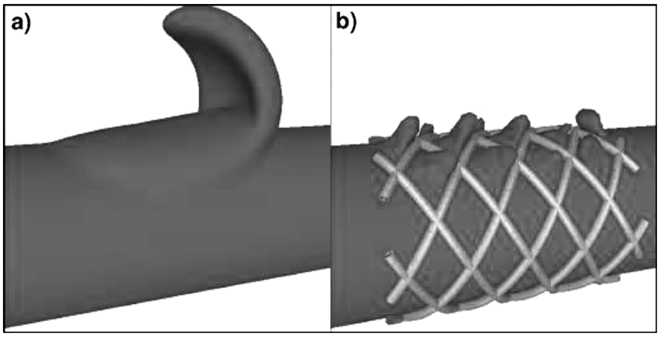

Fig. 8. Visualization of blood flowing into the aneurysm. Iso-velocity surfaces of 10 cm/s are presented for (a) aneurysm before stenting and (b) after stenting.

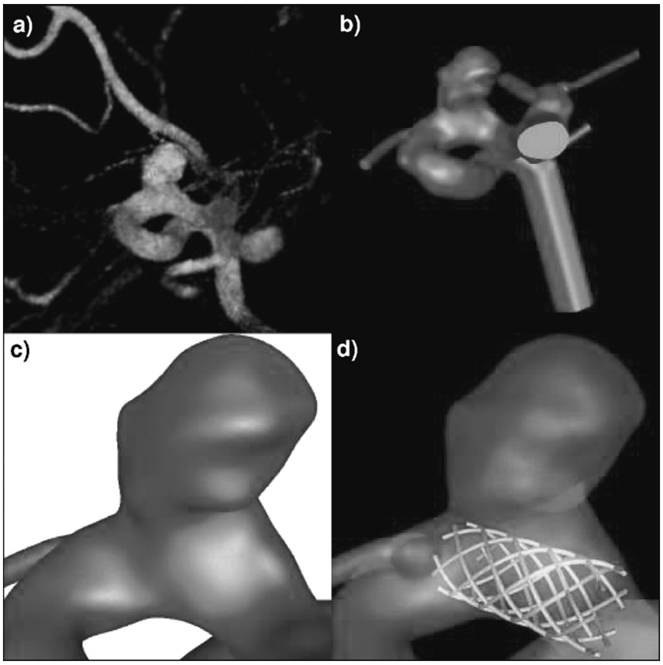

Fig. 10. Construction of a personalized model of stented aneurysm: (a) volume rendering of the 3DRA images, (b) reconstructed vascular model, (c) detail of the aneurysm model, and (d) aneurysm model after stenting.

Visualizations of the blood flowing into the aneurysm are presented in Fig. 8. Iso-surfaces of velocity magnitude are plotted for $v = 10\ \mathrm{cm/s}$ for the aneurysm before [Fig. 8(a)] and after [Fig. 8(b)] stenting. Again, these images clearly show the inflow regions and how they are affected by the placement of the stent.

Visualizations of the distribution of wall shear stress magnitude at peak systole are shown in Fig. 9 before [Fig. 9(a)] and after [Fig. 9(b)] stenting. These results are compatible with those obtained by Steinman *et al.* [40] using body fitted grids. Note that the regions of the wall that are inside the stent wires are inaccessible to the flow so that zero wall shear stress is returned by the computation.

B. Stenting of Personalized Aneurysm Model

The next example demonstrates the ability of the current method to model blood flows past endovascular devices in personalized anatomical models reconstructed from medical images. In this case, an idealized stent was placed in the parent vessel of a cerebral aneurysm model constructed from 3DRA images. The geometry of the stent was similar to the one in the previous section.

As in the previous example, the finite element grid was adaptively refined close to the stent surface using two levels of h-refinement. Roughly, the prestenting and poststenting grids contained 1.6 and 2.9 million elements, respectively. Fig. 10 shows a volume rendering of the 3DRA images [Fig. 10(a)], the reconstructed vascular model [Fig. 10(b)], a detail of the aneurysm model [Fig. 10(c)], and the model with the stent in place [Fig. 10(d)].

Comparisons of the prestenting and poststenting flow patterns are presented in Fig. 11. This figure shows visualizations of velocity contours at peak systole on a cut plane through the aneurysm before [Fig. 11(a)] and after [Fig. 11(b)] stenting. Iso-velocity surfaces of 40 cm/s are also shown before [Fig. 11(c)] and after [Fig. 11(d)] stenting. Again, these visualizations clearly display the flow changes induced by the implantation of the stent. In this case, since the stent did not fully cover the neck of the aneurysm, the flow into the aneurysm was not efficiently blocked. Furthermore, this stent did not perfectly conform to the vessel walls and produced undesirable flow disturbances in the parent vessel. This example shows how image-based CFD and mesh-embedding techniques can be used to evaluate endovascular treatment options. In this case, a curved stent that conforms to the parent vessel would be a better alternative.

C. Coiling of Personalized Aneurysm Model

This example illustrates the application of the embedding technique to the simulation of blood flows in coiled aneurysms. In this case, a model of a cerebral aneurysm was constructed from 3DRA images and filled with endovascular coils. Two coil models were generated inside the sac of the aneurysm. The coil thickness was 0.01 cm, the spiral diameter was 0.2 cm and the length was 10 cm and 20 cm, respectively. With these sizes, the coils covered 14% and 17% of the volume, respectively. Fig. 12

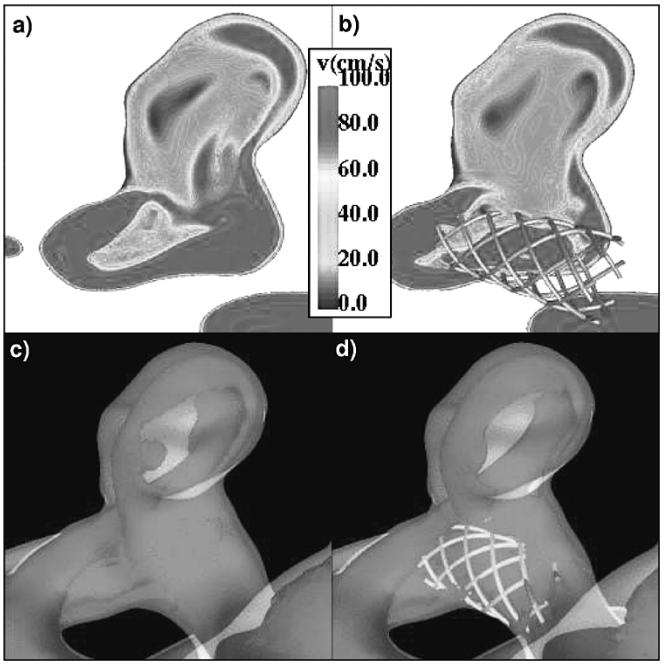

Fig. 11. Visualizations of blood flow in patient-specific model of stented aneurysm: velocity maps at peak systole on cut through the aneurysm (a) before stenting and (a) after stenting. Iso-velocity surfaces of 40 cm/s (c) before and (d) after stenting.

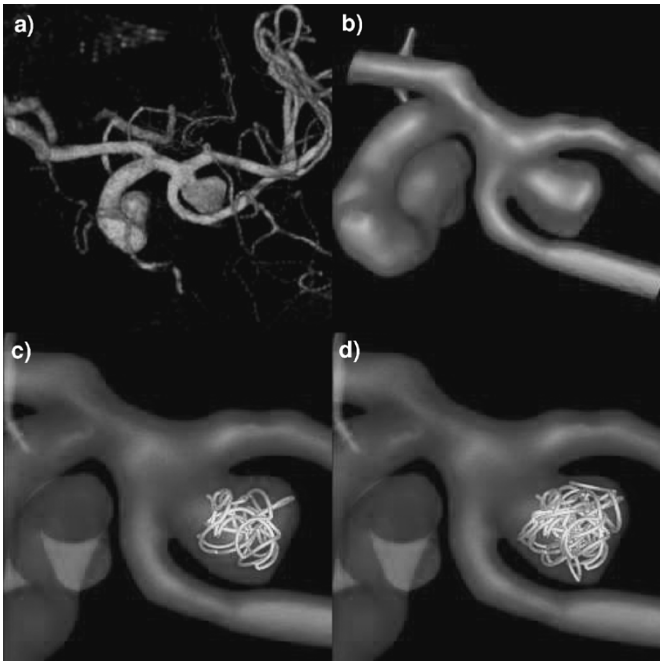

Fig. 12. Personalized model of cerebral aneurysm with coils: (a) volume rendering of the 3DRA images, (b) reconstructed vascular model, (c) aneurysm with 10-cm coil, and (d) aneurysm with 20-cm coil.

shows: a volume rendering of the 3DRA images [Fig. 12(a)], the reconstructed anatomical model [Fig. 12(b)], the aneurysm model with a 10-cm coil [Fig. 12(c)], and with a 20-cm coil [Fig. 12(d)].

As before, two levels of refinement were used to adapt the original finite element grid close to the coils. The meshes contained 2.0, 3.9, and 5.6 million elements for the precoil case, the 10-cm coil model and the 20-cm coil model, respectively.

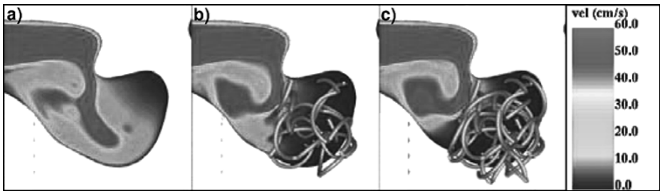

Fig. 13. Visualization of velocity patterns in coiled cerebral aneurysm: velocity maps at peak systole on cut through the aneurysm with (a) no coils, (b) 10-cm coil, and (c) 20-cm coil.

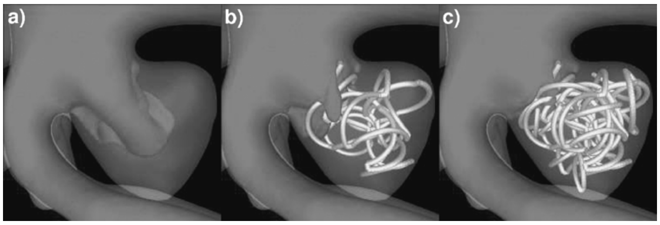

Fig. 14. Visualization of the flow into coiled aneurysms using iso-velocity surfaces of 50 cm/s for the aneurysm with (a) no coils, (b) 10-cm coil, and (c) 20-cm coil.

Pulsatile flow simulations were carried out on these three models. The calculations were performed on 16 processors of a Silicon Graphics Altix with 128 1.3-GHz Itanium processors and 240 GB of RAM. The total computing times for each mesh were approximately 9, 27 and 37 hours, respectively.

Visualizations of the blocking effects of the coils are presented in Figs. 13 and 14. Fig. 13 shows velocity contours at peak systole on a cut through the aneurysm for the original precoiled aneurysm [Fig. 13(a)] and the models with 10-cm [Fig. 13(b)] and 20-cm [Fig. 13(c)] coils.

Fig. 14 shows iso-velocity surfaces of 50 cm/s for each of the models. These pictures demonstrate how the coils stop the flow into the aneurysm and that, as expected, blood flows into the aneurysm more slowly with increasing coil packing.

Finally, "virtual angiograms" were produced in order to visualize the complex filling pattern of the aneurysm. For this purpose, the transport equation for a simulated dye was solved using a finite element formulation and the precomputed flow fields [42]. The injection of a bolus of contrast agent was simulated by prescribing a Gaussian time concentration function at the inflow boundary. The evolution of the concentration was then calculated over several cardiac cycles assuming periodic flows and visualizations of the dye concentration were produced using volume rendering techniques.

Fig. 15 shows the time evolution of the concentration of contrast agent for the precoiled aneurysm (left column) and for the aneurysm filled with a 10-cm coil (right column). It can clearly be seen that with this coil there is still a significant amount of dye entering the aneurysm, and that the aneurysm with the coil fills and empties more slowly than the un-treated aneurysm.

IV. Discussion

The application of unstructured grid embedding techniques for the efficient simulation of blood flow past complex endovascular devices has been described. These techniques have been previously developed in the context of complex fluid-structure interaction problems with changing topologies [20]–[22]. Endovascular devices such as the coils and stents used to treat cere-

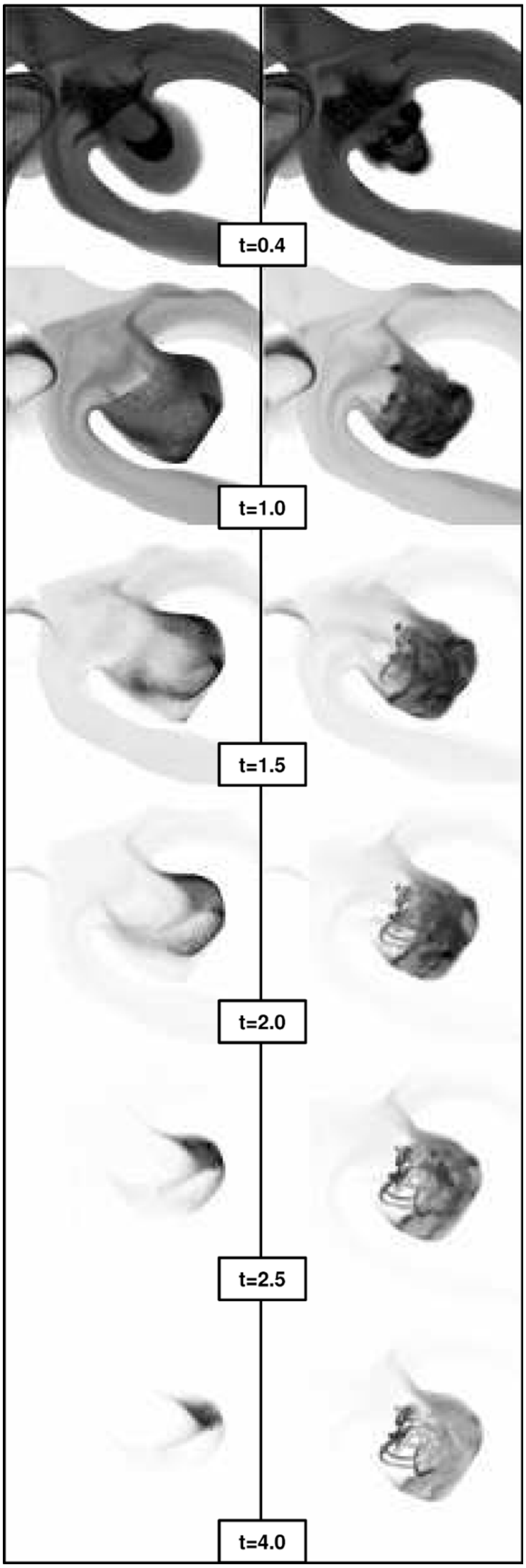

Fig. 15. Virtual angiograms of aneurysm before coiling (left column) and aneurysm with 10-cm coil (right column). Times given are relative to the period of the cardiac cycle.

bral aneurysms are of such geometric complexity that the construction of body fitted grids is extremely difficult and tedious. Moreover, combining an analytical representation of the device surface with personalized vascular models defined via a surface triangulation is also very problematic.

On the other hand, the use of the current embedding methodology is very robust, extremely simple, fast, and fits very naturally in the context of endovascular device simulation. The main limitation of the technique is the low resolution close to the device surface. This problem can be minimized by using adaptive grid refinement techniques to increase the mesh resolution around the embedded devices. This is a fully automated procedure that needs to be performed only once if the devices are stationary inside the vessels. Our simulations indicate that increasing the mesh resolution only close to the devices yields similar results to a global refinement of the mesh, with a substantial reduction of the computation time. However, the question of how much mesh refinement is necessary in order to achieve an accuracy comparable to body fitted grids remains to be investigated.

The second limitation of our approach is the geometrical modeling of the endovascular devices, which is actually independent of the embedding method used to calculate the blood flows. The simplest way of modeling complex cylindrical structures such as coils and stents is to use overlapping spheres along the axis of the filaments. Since these representations are used to find the CFD edges cut, sufficient packing of spheres along the axis is required in order to accurately model the geometry of the device. The number of spheres needed and its relation to the mesh resolution requires further investigation. The geometry of the stents and coils used in the numerical examples presented in this paper, although realistic, were not patient-specific. More realistic models of the coil packing can be constructed using finite element beam models and contact algorithms for the interaction of the coils with the vessel walls. In addition, the dimensions and amount of coils used to treat a given aneurysm are usually recorded and can be used to create more personalized models of the procedures.

Validation of personalized models of cerebral aneurysms with embedded endovascular devices is very difficult since there is no gold standard for measuring blood flows *in vivo*. However, visualization techniques such as our "virtual angiograms" offer the possibility of direct comparison with real angiograms obtained at a high frame rate during a rapid bolus injection of contrast agent. These imaging techniques are similar to the ones routinely used to evaluate the filling of an aneurysm during an interventional procedure. Further studies are required to validate the methodology with *in vitro*, animal and *in vivo* data.

Our future efforts will focus on the following.

- study of mesh resolution requirements and accurate device modeling:
- development of more refined and possibly image-based stent and coil placement/deployment:
- estimation of forces on coil mass:
- development of models of coil compaction:
- development of models of blood clotting around coils:
- study of the amounts of coils necessary to occlude the flow into a given aneurysm and their optimal placement:
- link to optimization algorithms to enhance endovascular procedures for individual patients.

In conclusion, this paper demonstrated that the geometric flexibility offered by the unstructured grid embedding techniques provides a natural framework for efficient simulation of blood flow past complex endovascular devices on a patient-specific basis.

It is thought that hemodynamic forces, i.e., wall shear stress and pressure at the flow impingement region, are involved in the growth and rupture of cerebral aneurysms. Eventually, the determination of growth and failure modes of the aneurysm that leads to rupture and bleeding must account for blood-vessel interactions that also involve other factors such as wall internal stresses and structural properties of the vessel wall. Current endovascular procedures are based on the hypothesis that reducing the blood flow into an aneurysm causes flow stasis and thrombosis which reduces the risk of bleeding [43]. The significance of the present work is that performing patient-specific CFD simulations of cerebral aneurysms after endovascular treatment is now possible with very little increase in effort. This opens the door to the optimal choice of therapy and personalization of the procedures. Virtual coiling and stenting procedures can now be performed in order to explore different therapeutic options such as the determination of the optimal size, placement, and amounts of coils for a given individual.

References

[1] W. E. Stehbens, Intracranial aneurysms, in Pathology of the Cerebral Blood Vessels, Mosby, St. Louis, MO, pp. 351–470, 1972.

[2] G. N. Foutrakis, H. Yonas, and R. J. Sclabassi, "Saccular aneurysm formation in curved and bifurcation arteries," *AJNR*, vol. 20, pp. 1309–1317, 1999.

[3] B. Weir, "Unruptured intracranial aneurysms: a review," *J. Neurosurg.*, vol. 96, pp. 3–42, 2002.

[4] F. H. Linn, G. J. Rinkel, A. Algra, and J. van Gjin, "Incidence of subarachnoid hemorrhage: role of region, year, and rate of computed tomography: a meta-analysis," *Stroke*, vol. 27, no. 4, pp. 625–629, 1996.

[5] F. Tomasello, D. D'Avella, F. M. Salpietro, and M. Longo, "Asymptomatic aneurysms. Literature meta-analysis and indications for treatment," *J. Neurosurg. Sci.*, vol. 42, no. 1, pp. 47–51, 1998.

[6] H. R. Winn, J. A. Jane, J. Taylor, D. Kaiser, and G. W. Britz, "Detection of asymptomatic incidental aneurysm: review of 4568 arteriograms," *J. Neurosurg.*, vol. 96, no. 1, pp. 43–49, 2002.

[7] A. J. Ringer, D. K. Lopes, A. L. Boulos, L. R. Guterman, and L. N. Hopkins, "Current techniques for endovascular treatment of intracranial aneurysms," in *Sem. Cerebrovasc. Diseases Stroke*, vol. 1, 2001.

[8] L. D. Jou, C. M. Quick, W. L. Young, M. T. Lawton, R. Higashida, A. Martin, and D. Saloner, "Computational approach to quantifying hemodynamic forces in giant cerebral aneurysms," *AJNR*, vol. 24, pp. 1804–1810, 2003.

[9] C. Groden, J. Laudan, S. Gatchell, and H. Zeumer, "Three-dimensional pulsatile flow simulation before and after endovascular coil embolization of a terminal cerebral aneurysm," *J. Cereb. Blood Flow Metab.*, vol. 21, no. 12, pp. 1464–1471, 2001.

[10] B. B. Lieber, V. Livescu, L. N. Hopkins, and A. K. Wakhloo, "Particle image velocimetry assessment of stent design influence on intra-aneurysmal flow," *Ann. Biomed. Eng.*, vol. 30, pp. 768–777, 2002.

[11] M. Hirabayashi, M. Ohta, D. A. Rufenacht, and B. Chopard. (2003, Aug.) Characterization of flow reduction properties in an aneurysm due to a stent. *Phys. Rev. E Statist. Nonlinear Soft. Matter Phys.* [Online], vol (2; Pt. 1:)021918

[12] C. N. Ionita, Y. Hoi, H. Meng, and S. Rudin, "Particle image velocimetry (PIV) evaluation of flow modification in aneurysm phantoms using asymmetric stents," presented at the SPIE Conf. Med. Imag., San Diego, CA, Feb. 14–19, 2004.

[13] D. A. Steinman, J. S. Milner, C. J. Norley, S. P. Lownie, and D. W. Holdworth, "Image-based computational simulation of flow dynamics in a giant intracranial aneurysm," *AJNR*, vol. 24, no. 4, pp. 559–566, 2003.

[14] J. R. Cebral, M. Hernandez, and A. F. Frangi, "Computational analysis of blood flow dynamics in cerebral aneurysms from CTA and 3D rotational angiography image data," presented at the ICCB 03, Zaragoza, Spain, Sept. 24–26, 2003.

[15] J. R. Cebral, M. Hernandez, A. F. Frangi, C. M. Putman, R. Pergolizzi, and J. E. Burgess, "Subject-specific modeling of intracranial aneurysms," presented at the SPIE Conf. Med. Imag., San Diego, CA, Feb. 14–19, 2004, paper 5369-38.

[16] J. R. Cebral, M. A. Castro, J. E. Burgess, and C. M. Putman, "Cerebral aneurysm hemodynamics modeling from 3D rotational angiography," presented at the ISBI 04, Arlington, VA, April 15–18, 2004.

[17] D. Sharov, H. Luo, J. D. Baum, and R. Löhner, "Time-accurate implicit ALE algorithm for shared-memory parallel computers," presented at the 1st Int1 Conf1 Computational Fluid Dynamics, Kyoto, Japan, July 10–14, 2000.

[18] M. J. Aftosmis, M. J. Berger, and G. Adomavicius, "A Parallel Multilevel Method for Adaptively Refined Cartesian Grids With Embedded Boundaries," American Institute of Aeronautics and Astronautics, Arlington, VA, Tech. Rep. AIAA-00-0808, 2000.

[19] C. S. Peskin, "The immersed boundary method," *Acta Numerica*, vol. 11, pp. 479–517, 2002.

[20] R. Löhner, C. Yang, J. D. Baum, H. Luo, D. Pelessone, and C. Charman, "The numerical simulation of strongly unsteady flows with hundreds of moving bodies," *Int. J. Num. Meth. Fluids*, vol. 31, pp. 113–120, 1999.

[21] R. Löhner, C. Yang, J. R. Cebral, J. D. Baum, H. Luo, E. Mestreau, D. Pelessone, and C. Charman, "Fluid-structure interaction algorithms for rupture and topology change," presented at the JSME Computational Mechanics Division Meeting, Matsuyama, Japan, Nov. 1999.

[22] R. Löhner, J. D. Baum, E. L. Mestreau, D. Sharov, C. Charman, and D. Pelessone, "Adaptive embedded unstructured grid methods," *Int. J. Num. Meth. Eng*, vol. 60, pp. 641–660, 2004.

[23] R. Löhner, "Regridding surface triangulations," *J. Comp. Phys.*, vol. 126, pp. 1–10, 1996.

[24] J. R. Cebral and R. Löhner, "From medical images to anatomically accurate finite element grids," *Int. J. Num. Meth. Eng.*, vol. 51, pp. 985–1008, 2001.

[25] R. Löhner and E. Oñate, "An advancing point grid generation technique," *Comm. Num. Meth. Eng.*, vol. 14, pp. 1097–1108, 1998.

[26] ——, "A general advancing front technique for filling space with arbitrary objects," *Int. J. Num. Meth. Eng.*, vol. 61, no. 12, pp. 1979–1991, 2004.

[27] R. Löhner, *Applied CFD Techniques*. New York: Wiley, 2001.

[28] P. K. Kundu and I. M. Cohen, *Fluid Mechanics*, 3rd ed. New York: Elsevier, 2004.

[29] C. A. Taylor, T. J. R. Hughes, and C. K. Zarins, "Finite element modeling of blood flow in arteries," *Comput. Meth. Appl. Mech. Eng.*, vol. 158, pp. 155–196, 1998.

[30] F. Thomaset, *Implementation of Finite Element Methods for Navier-Stokes Equations*. Berlin, Germany: Springer-Verlag, 1981.

[31] M. D. Gunsburger and R. Nicolaides, Eds., *Incompressible Computational Fluid Dynamics: Trends and Advances*. Cambridge, U.K.: Cambridge Univ. Press, 1993.

[32] M. M. Hafez, Ed., *Numerical Simulation of Incompressible Flows*. Singapore: World Scientific, 2003.

[33] R. Löhner, C. Yang, J. R. Cebral, O. Soto, F. Camelli, and J. Waltz, "Improving the Speed and Accuracy of Projection-Type Incompressible Flow Solvers," American Institute of Aeronautics and Astronautics, Arlington, VA, AIAA-03-3991-CP, 2003.

[34] R. Löhner, C. Yang, E. Oñate, and S. Idelssohn, "An unstructured grid-based, parallel free surface solver," *Appl. Num. Math.*, vol. 31, pp. 271–293, 1999.

[35] Y. Saad, *Iterative Methods for Sparse Linear Systems*. Boston, MA: PWS Pub. Co., 1996.

[36] H. Luo, J. D. Baum, and R. Löhner, "A fast matrix-free implicit method for compressible flows on unstructured grids," *J. Comp. Phys.*, vol. 146, pp. 664–690, 1998.

[37] D. Sharov, H. Luo, J. D. Baum, and R. Löhner, "Implementation of Unstructured Grid GMRES+LU-SGS Method on Shared-Memory, Cache-Based Parallel Computers," American Institute of Aeronautics and Astronautics, Arlington, VA, AIAA-00-0927, 2000.

[38] S. Del Pino and O. Pironneau, "Fictitious domain methods and Freefem3d," presented at the ECCOMAS CFD Conf., Swansea, , U.K., 2001.

[39] R. Löhner and J. D. Baum, "Adaptive H-refinement on 3-D unstructured grids for transient problems," *Int. J. Num. Meth. Fluids*, vol. 14, pp. 1407–1419, 1992.

[40] G. R. Stuhne and D. A. Steinman, "Finite element modeling of the hemodynamics of stented aneurysms," *J. Biomech. Eng.*, vol. 26, no. 3, pp. 382–387, 2004.

[41] J. R. Cebral, R. Löhner, P. L. Choyke, and P. J. Yim, "Merging of intersecting triangulations for finite element modeling," *J Biomech.*, vol. 34, pp. 815–819, 2001.

[42] F. Calamante, P. J. Yim, and J. R. Cebral, "Estimation of bolus dispersion effects in perfusion MRI using image-based computational fluid dynamics," *NeuroImage*, vol. 19, pp. 342–353, 2003.

[43] S. G. Imbesi and C. W. Kerber, "Analysis of slipstream flow in a wide-necked basilar artery aneurysm: evaluation of potential treatment regimes," *AJNR Am. J. Neuroradiol.*, vol. 22, pp. 721–724, 2001.