Journal of Engineering Mathematics **47:** 405–418, 2003.
© 2003 *Kluwer Academic Publishers. Printed in the Netherlands.*

A mesh-updating scheme for hemodynamic simulations in vessels undergoing large deformations

DEHONG ZENG[1] and C.R. ETHIER[1,2]
[1]*Department of Mechanical and Industrial Engineering, 5 King's College Road, University of Toronto, Toronto, Ontario M5S 3G8, Canada, email: ethier@mie.utoronto.ca;* [2]*Institute for Biomaterials and Biomedical Engineering, U. of Toronto, Toronto, Canada*

Received 12 September 2002; accepted in revised form 4 August 2003

Abstract. Several arteries, notably the coronary arteries, experience large motions and deformations believed to affect their hemodynamic environment. These arterial motions are 3D and complex, and *in vivo* data sets usually do not completely describe the resulting arterial wall motions. Here we present a mesh-updating scheme for such situations, thus allowing numerical simulation of arterial hemodynamics in arteries whose motion is described by (necessarily) incomplete *in vivo* data sets. The scheme works by first coupling information about arterial cross-sectional shape to motion of the artery axis. Motion of a subset of surface nodes ("control nodes") is then specified, and this motion is used to drive all nodes in the mesh through an extension of a semi-torsional spring-analogy model. Numerical experiments were carried out on unstructured tetrahedral meshes generated in model geometries and in a right coronary artery model undergoing physiologically accurate motion. Results show that the quality of the dynamic mesh, as evaluated through elemental aspect ratios and Jacobian values, is extremely well preserved by this scheme, even during large deformations. Testing indicates the number of control nodes necessary to attain a high level of geometric fidelity of the mesh. We conclude that the algorithm is valuable for computing hemodynamic patterns in moving arteries using fully unstructured meshes.

Key words: cineangiography, finite-element flow simulation, hemodynamics, physiologic motion, right coronary artery, spring-analogy model

1. Introduction

All large arteries experience some wall motion. Causes include the time-varying pressure within the vessel, joint motion (for arteries crossing articulating joints) and myocardial movement (for the coronary arteries). Depending on its magnitude, such motion can have potentially significant hemodynamic effects. This is particularly true of the coronary arteries, which undergo relatively large motions and deformations that have been hypothesized to influence coronary artery atherogenesis [1–4].

Computational modeling is an attractive tool for studying the hemodynamic environment in moving arteries. This requires an algorithm capable of solving the Navier-Stokes equations in moving domains, as well as a scheme for deforming the computational mesh/grid in response to movement of the computational domain. The mesh/grid-deformation problem is non-trivial when considering physiologically realistic deformations in anatomically realistic arterial geometries. The goal of this work is to describe a scheme for mesh updating in situations where incomplete information about the motion of the artery wall is available, which is often the case when the arterial motion is determined experimentally. The specific application of this work is the modeling of hemodynamic patterns in moving coronary arteries.

2. Methods

In this work we consider a fully unstructured computational mesh formed from tetrahedra, although the generalization to other element types is straightforward. We adopt an approach in which an existing mesh is deformed to follow the motion of the computational domain, rather than generating a new mesh at each time step in the computation. If the motions of the nodes on the boundaries of the computational domain are known, then there are techniques available for moving the interior nodes of a pre-existing mesh; see *e.g.* [5–10]. The problem is therefore reduced to one of specifying the time-varying motion of the mesh's surface nodes.

In certain calculations (*e.g.* coupled fluid-wall calculation), wall-motion information arises naturally as part of the calculation. Here we consider a different situation, in which the motion of wall nodes is incompletely specified. This is typical of situations in which the arterial motion is determined experimentally. For example, in a recent study of flow patterns in a moving right-coronary-artery (RCA) model, an experimental data set describing the *in vivo*, 3D, time-varying positions of markers on the RCA axis was available [11]. No information about the position of the artery wall *per se* was available. Clearly, in such cases, an algorithm is needed to link nodal displacements to the motions of the RCA axis. Furthermore, this algorithm must be fairly robust, in the sense that the motions of a large number of nodes are to be driven by the movement of a relatively small number of points along the vessel axis. Below we describe such an algorithm.

2.1. Overview

To implement the algorithm, four major issues have to be addressed:

2.1.1. *Dynamic geometry representation.*

In the case of the RCA, the time-varying positions of the artery axis were specified based on bi-plane cineangiogram data sets. Cross-sectional contours were then assembled onto the dynamic RCA axis, and the assembled contour set was displaced to follow the dynamic RCA axis. The moving contour set represented the artery wall of the RCA. The details of this step are specific to the experimental data set that is available.

2.1.2. *Selection of control nodes on the initial mesh.*

A fully unstructured computational mesh is generated on an initial geometry, and a small subset of surface nodes on this mesh are identified as "control nodes". In the case of the RCA described above, we selected groups of control nodes so that each group formed a ring approximately centered on a marker on the RCA axis whose position was known. We refer to such rings as "control rings".

2.1.3. *Passing geometrical variations to control nodes.*

By specifying the relationship between control rings and artery wall location, control rings are moved along with the artery wall. In this way dynamic geometrical information is passed to a subset of mesh nodes on the wall.

2.1.4. *Displacing the entire mesh.*

Nodes that are not on control rings are updated through a semi-torsional spring-analogy model. In this approach, all element edges are treated as fictitious springs and an equilibrium condition governs the displacement of the nodes connected to these springs.

We now describe in more detail how this algorithm was implemented in the case of a dynamically deforming right coronary artery, where only information about the position of the RCA axis was available.

2.2. Dynamic RCA geometry

The *in vivo* positions of the RCA axis at ten selected instants during a cardiac cycle were determined from analysis of biplane cineangiograms [12]. Briefly, biplane coronary cineangiograms were obtained from a patient undergoing routine catheterization for diagnosis of coronary heart disease. Spatial resolution in the image acquisition was about 3 pixels/mm. Two orthogonal projections of the selected section of the RCA were obtained at the rate of 12.5 frames/second. A template matching technique in conjunction with a thinning algorithm was applied to each series of the RCA projections, to derive two sets of two-dimensional vessel axes representing the frame-by-frame motion of the axis of the RCA within each projection plane. Using these two sets of axes, a 3D curve representing the centroid of the RCA lumen (RCA axis) was reconstructed. The *in vivo* positions of the RCA axis at ten selected instants during a cardiac cycle were determined, with 374 markers on the RCA axis tracked in time. The original RCA axis was not spatially smooth and the “noise” level was less than the spatial resolution of 3 pixels/mm. Cubic B-spline interpolation was used to smooth the RCA axis and 126 markers out of the 374 markers were identified along one RCA axis to be tracked in time. To derive the location of the RCA axis at an arbitrary instant during the cardiac cycle, cubic-spline interpolation in time was applied to the smoothed RCA axis based on its known 10 frames of configurations.

Figure 1 shows the RCA axis at one instant and the trajectories of two markers on the axis. On average, the distance that a marker covered in one cardiac cycle was around 6 cm, with a pulse curvature[*] near $0{\cdot}33\ \mathrm{cm}^{-1}$ and pulse torsion of around $4{\cdot}45\ \mathrm{cm}^{-1}$. Considering that the average length of a RCA is about 11 cm, these numbers indicate that the RCA axis experiences large variations in both curvature and torsion during its complex 3D motion.

Clearly, a moving 3D curve as shown in Figure 1 is an incomplete representation of the dynamic arterial geometry, since cross-sectional information is missing. For the case of the RCA considered in this work, the spatial resolution of the cineangiograms was insufficient to specify cross-sectional information about the artery, and it was necessary to supplement the axial motion with cross-sectional information. There are several possible sources for this supplemental information: one can assume a cross-sectional shape [13], impose cross-sectional shapes determined from a CT scan of a cast of a static RCA, or use cross-sectional shapes measured by other *in vivo* techniques (*e.g.* ANGUS [14]). Below we describe the procedure used for mapping static (non-changing) but otherwise arbitrary arterial cross-sections onto the moving RCA axis.

The static geometry contours can be seen in Figure 2, where it is observed that the artery has a significant amount of curvature in the anterior-posterior (AP) view. We denote the curvature seen in this view as the “primary curvature”. An algorithm was developed to assemble these static contours onto the dynamic RCA axis at end diastole (ED). We chose end diastole, since this configuration of the RCA was felt to correspond most closely to the post mortem state from which the static RCA contours were obtained; however, this choice was somewhat arbitrary. Static contours were mapped to axial locations on the dynamic axis based

[*] Pulse curvature (torsion) is the difference between the maximum and minimum curvature (torsion) at one marker, over the cardiac cycle.

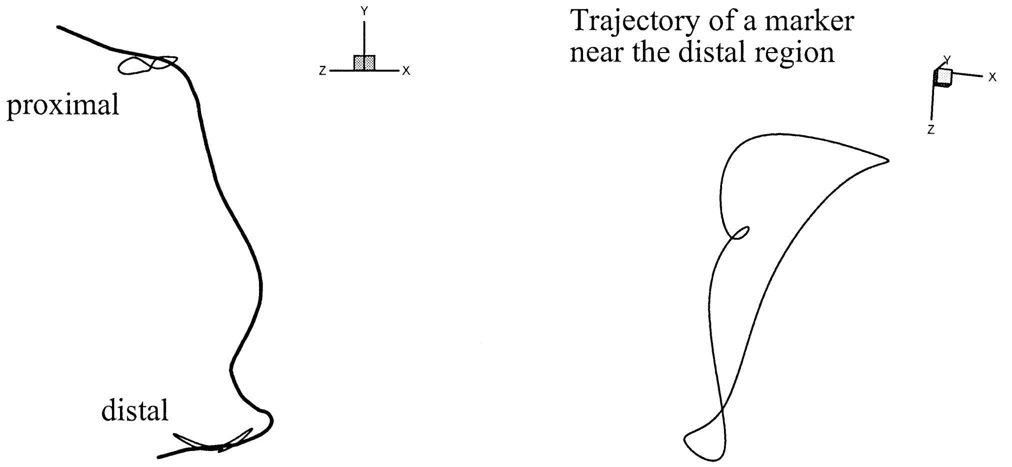

Figure 1. Motion of the RCA axis as determined by cineangiograms. The long open curve represents the RCA axis at one instant in a cardiac cycle (left panel). Each marker on the axis traces out a closed curve over a complete cardiac cycle. The two closed curves in the plot are path lines of two markers on the axis. A magnified view of one trajectory is shown in the right panel.

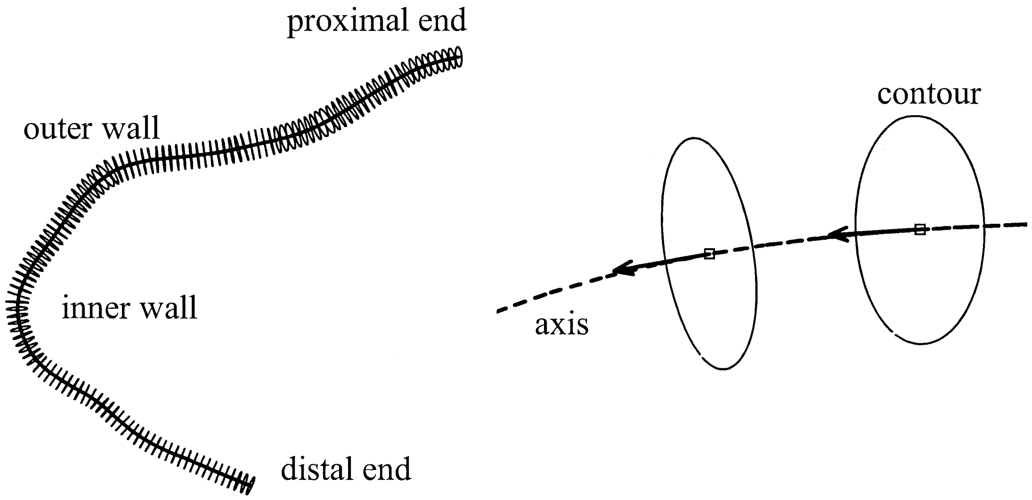

Figure 2. Cross-sectional contours of a static RCA model are used to define the arterial wall geometry (left panel). Contours are equally spaced along the RCA axis, which runs through all contours in the center of the RCA lumen. The myocardial surface (inner wall) and pericardial surface (outer wall) are the inner side and outer side of the primary curvature, respectively, in this anterior-posterior (AP) view. Right panel: a magnified view of two of the contours. Each contour is associated with one RCA axis marker, and is perpendicular to its local axis tangent direction, as shown by the arrows.

on their relative distance from the ostium, as well as the location of landmarks (*e.g.* major branches and well-defined bending/turning points). The orientations of the contours on the dynamic RCA axis were based on two requirements: (i) each contour plane was required to lie normal to the local tangent direction of the RCA axis; and (ii) the inner and outer wall locations of each contour were required to match the inner and outer wall locations on the dynamic geometry. The application of the second condition required that the “inner” and “outer” walls of the RCA be defined. This was somewhat arbitrary, and was done based on

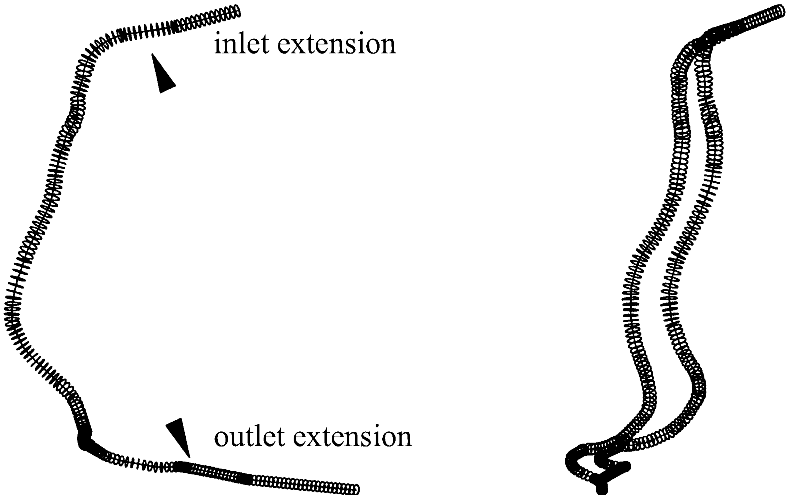

Figure 3. Complete contour set representing the RCA geometry with inlet/outlet extensions shown in AP view (left panel). The arrows indicate the boundaries between actual RCA lumen and the extensions. In the right panel, contour sets at two instants are shown in a view close to the mediolateral view.

visual inspection of models of the artery as seen in the AP view. The “inner” wall corresponded to the myocardial side of the artery, while the “outer” wall corresponded to the epicardial side.

Under the assumption that the location and orientation of each contour was invariant with respect to the local axis marker and axis tangent direction, the assembled contours were displaced and rotated to follow the dynamic RCA axis. Since no twisting of the contours along the RCA axis was imposed, and since the shape of the contours was invariant, only global wall motions* were represented. A more complete data set would be required to specify both global and local changes.

In order to facilitate hemodynamic simulations, inlet and outlet extensions were added to the dynamic RCA model. Each extension consisted of a straight, stationary section and a flexible section that connected the moving vessel with the fixed straight section (Figure 3). The axis of the curved extension was a B-spline that insured a smooth connection between the stationary section and the moving RCA axis. The stationary portions of the inlet and outlet extensions were both 1·5 cm long, the flexible inlet section was ∼1·2 cm long, and the flexible outlet section was about 3 cm long. The cineangiogram data represented the section of RCA starting approximately 1∼1·5 cm downstream of the ostium. Therefore the boundary between the stationary section and the flexible section of the inlet extension corresponded approximately to the *in vivo* location of the ostium.

2.3. Control nodes and control rings

A fully unstructured (tetrahedral) finite-element mesh was generated on the initial RCA geometry** using ICEMCFD TETRA (Version 4.02; ICEM CFD, Berkeley CA). Each tetrahedral

*Artery wall motion can be viewed as a superposition of “global” and “local” changes. We define global changes to refer to movement of the artery wall resulting from variation in curvature and torsion of the RCA axis, while local changes refer to distensible wall motion induced by the pulse pressure, as well as twisting of the artery wall around the RCA axis.

**The RCA lumen configuration at any instant in the cardiac cycle can be taken as the initial geometry on which the initial FEM mesh is generated.

element contained 10 velocity nodes (4 vertex nodes and 6 mid-side nodes). Nodes were categorized as surface vertices, surface mid-side nodes, internal vertices and internal mid-side nodes.

The known motion of the arterial contours was then coupled to the motion of control nodes on the surface of the mesh. We selected control nodes from surface vertices so that the control nodes formed closed loops that were approximately coincident with arterial contours. These closed loops of control nodes were denoted as "control rings". For each contour, nodes making up the associated control ring were chosen by first selecting a seed node that was close to the contour on the surface. Surface nodes adjacent to this seed node were then evaluated, and the node that was closest to the contour while traveling in a circumferential direction around the artery was selected. This process was repeated until the nodes formed a closed loop. The average deviation of a control ring from its associated contour was about half of an inter-nodal spacing. Control nodes constituted approximately 3%–5% of surface nodes.

2.4. Node updating using control rings

In order to propagate contour motion to control node motion, a connection between each control ring and its associated contour had to be specified. A completely rigid connection seemed unrealistic, while an extremely flexible connection would not transfer geometrical changes effectively. We took a compromise approach consisting of the following four steps that together constituted one mesh-updating step.

2.4.1. *Motion of control nodes.*

It was assumed that there was a rigid connection between each contour and its associated control ring, so that the control nodes were moved in exactly the same way as the contours. This implicitly assumed that surface nodes in the mesh represented material points on the artery wall.

2.4.2. *Vertex updating.*

Control-node displacements were propagated to other nodes in the mesh, using a semi-torsional spring analogy model. This model is described in detail elsewhere [10], but in brief, each element edge is replaced by a spring whose stiffness depends on the edge length and the magnitude of the intraelement angle opposite the edge. Changes in the locations of the control nodes are propagated to all other nodes by the "springs" through a requirement that all spring forces sum to zero at each node. A coarse spring-connection scheme was used, in which only vertex nodes were directly moved by the fictitious springs; mid-side node locations were then obtained by averaging the locations of the two vertices that shared the edge with a mid-side node.

2.4.3. *Control-node relaxation.*

We then allowed all control nodes to adjust their locations locally by holding all other nodes fixed and applying the equilibrium spring force condition to the control nodes themselves. This prevented elements that were attached to the control nodes from becoming excessively deformed.

2.4.4. *Surface mid-side node correction.*

Mid-side nodes on the surface of the initial mesh lay on a curved 3D surface. The averaging used in Step 2 to update mid-side nodal locations caused mid-side nodes on the surface to leave the curved surface, thus creating artefactual mid-side surface-node displacements, *i.e.*, displacements that were not due to contour motion *per se.* These displacements in turn caused artefactual wall velocities that were potentially very large when the time-step size was small, since the magnitude of the artefactual displacement did not diminish with time-step size. In order to avoid such artefactual wall velocities, a surface spring analogy model was developed to move surface mid-side nodes only in response to contour motions. Essentially, the mid-side nodal positions obtained in vertex updating were just starting guesses for the following algorithm.

A surface mesh was formed by first identifying all the faces lying on the boundary of a mesh domain, based on the connectivity and boundary node tables. Each surface face was a 6-node triangular element, and was subdivided into 4 triangular elements with 3 nodes each. The surface mesh was then formed from all these 3node triangular elements. Extension of the spring analogy to the surface mesh was straightforward: the edges of the surface mesh were modeled as semi-torsional springs. Data structures required for implementing the surface-spring analogy model were similar to those in the volume mesh model.

Some nodes in the surface mesh were vertices in the original volume mesh, and they were treated as control nodes in the surface mesh, at which nodal displacements were known through vertex updating. The other nodes in the surface mesh were surface mid-side nodes in the original volume mesh, and these nodes were updated through the surface mesh spring analogy model. Since vertex movement only responded to contour motion, nodal displacements at surface mid-side nodes were then functions of contour motion, thus avoiding the fictitious wall motion from averaging coordinates of vertices.

2.5. SOR ITERATION

The nodal-spring-force equilibrium equations were solved using the successive overrelaxation (SOR) method, through which nodal displacements at uncontrolled nodes were derived. It turned out that deformations on the RCA mesh were spatially non-uniform. Specifically, there were higher degrees of deformations in the distal region than in other parts of the mesh. Because of these differences, the solution converged at different rates in different regions within the mesh. We made use of this locality of deformations to speed up the SOR solver, as follows: The stopping criterion for the SOR iteration was $\max\{|\delta_{1,i}^{k+1}-\delta_{1,i}^{k+}|, |\delta_{2,i}^{k+1}-\delta_{2,i}^{k+}|, |\delta_{3,i}^{k+1}-\delta_{3,i}^{k+}|\} \leq \varepsilon$ where i ranged over all uncontrolled nodes, k denoted iteration, and δ_1, δ_2, δ_3 were the three components of nodal displacement. The value of ε was between 5×10^{-6} and 1×10^{-5}. In the beginning of a SOR iteration, each vertex was first checked for convergence of the solution at the vertex and all of its neighbours based on the information from the previous iteration. If the solution was locally converged in the previous iteration, this vertex was not updated in the current iteration; otherwise, it was updated as usual. Using this approach, fewer and fewer vertices were updated in successive iterations, thus saving significant amounts of CPU time. The same technique was applied to the iterative solver in the surface-spring-analogy model.

3. Results

We first evaluated the vertex update and surface-spring analogy algorithms by using a mesh with 15,669 elements generated in a curved tube of circular cross-section. The ratio between the initial radius of curvature and the tube radius was 8. The centre of curvature of the initial mesh was at the origin of the Cartesian coordinates $(x, y, z) = (0, 0, 0)$. The two ends of the tube were designated as the inlet and outlet, respectively. Two vertex update tests and one surface-spring analogy test are described here.

3.1. Vertex update

In the first two tests, we evaluated how control-node motion drove the movement of other nodes in the mesh through vertex updating. Specifically, different mesh motions were obtained by specifying the movement of only two control rings, one on the inlet and the other on the outlet. In Test 1, the outlet control ring moved cyclically according to the following step-wise nodal displacement specification: $\delta x = 0$, $\Delta y = 0$, $\Delta z = 5a \cos\left(\frac{2\pi t}{T}\right) \frac{\Delta t}{T}$, where a is the radius of the tube, T is the period of the motion, t is time, and Δt is the time-step size. In Test 2 the outlet control ring's motion was given by $\delta x = 5a \cos\left(\frac{2\pi t}{T}\right) \frac{\Delta t}{T}$, $\Delta y = 0$, $\Delta z = 5a \cos\left(\frac{2\pi t}{T}\right) \frac{\Delta t}{T}$. In both tests, the inlet nodes were kept fixed in space. One cycle of deformation was achieved through 250 time steps. At each time step we evaluated the minimum elemental Jacobian within the mesh (which must be positive to avoid element inversion) and the maximum elemental aspect ratio. Elemental aspect ratio was defined as $\frac{i_{\max}}{h_{\min}}$, where $i_{\max}$ and $h_{\min}$ were the largest edge length and the smallest distance between a face and its opposite vertex, respectively.

The results showed that the entire mesh deformed cyclically in response to the movement of the outlet, without element inversion (Figures 4 and 5). The maximum elemental aspect ratio remained less than 7, indicating that elements in the mesh were well-formed throughout the deformation cycles. More than two control rings would be used in an actual arterial flow simulation, but these experiments showed the effectiveness of the vertex-updating scheme in deforming a mesh using only a small number of control nodes. The same vertex-update tests on finer meshes with up to 200,000 elements gave similar results.

3.2. Surface-mesh-spring analogy and displacement error

Test 3 used the same mesh as in the previous tests, but with a variable number of control rings evenly distributed along the tube. The surface-spring analogy was invoked to further update the surface mid-side nodes after vertex updating. The displacement of a control node i at each time step was specified as:

$$\Delta x_i = c \frac{x_i}{\sqrt{x_i^2 + y_i^2}}, \tag{1}$$

$$\Delta y_i = c \frac{y_i}{\sqrt{x_i^2 + y_i^2}}, \tag{2}$$

$$\Delta z_i = 0{\cdot}0, \tag{3}$$

where x_i, y_i are the x- and y-coordinates of the control node i; Δx_i, Δy_i, Δz_i are the components of nodal displacements, and c is a constant, taken to be $0{\cdot}1$. Since the center of curvature

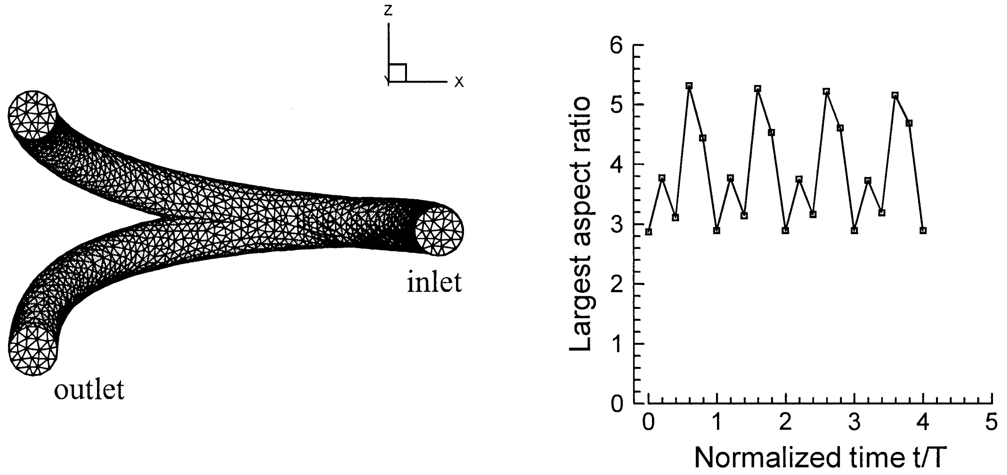

Figure 4. Summary of results of Test 1, where a curved tube mesh containing 15,669 elements was deformed cyclically between the two extreme configurations shown in the left panel. For this test, only two control rings were used, one on the inlet, and one the outlet. The inlet control ring was held fixed, and the outlet ring was displaced periodically in the z-direction. Right panel: the largest elemental aspect ratio of the mesh through several cycles of deformation. The time was normalized by the period of the deformation T. The mesh was deformed for 4 cycles.

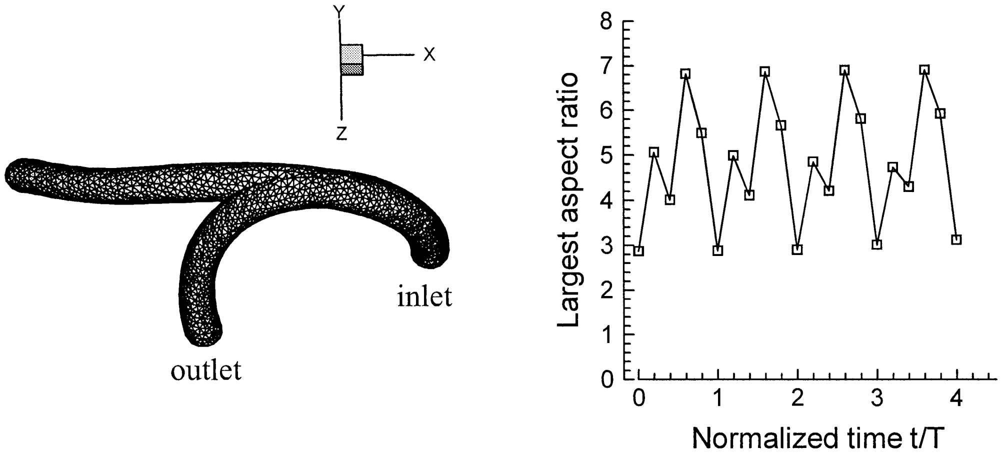

Figure 5. Summary of results of Test 2, which was similar to Test 1, except that the outlet control ring had displacements imposed in both the x- and z-directions. The largest elemental aspect ratio *vs.* time is shown in the right panel.

of the tube was at the origin $(x, y, z) = (0, 0, 0)$, the above nodal displacement specifications imposed a radial motion of the control nodes away from the center of curvature, as shown in the left panel of Figure 6.

Nodes in the mesh were successfully displaced by the control rings. To quantify the accuracy of this displacement scheme, we defined a displacement error as:

$$\varepsilon = \left\{ \frac{\sqrt{\frac{1}{N}\sum_{j=1}^{N}\left(\Delta\tilde{x}_j - \Delta x_j\right)^2}}{a}, \frac{\sqrt{\frac{1}{N}\sum_{j=1}^{N}\left(\Delta\tilde{y}_j - \Delta y_j\right)^2}}{a}, \frac{\sqrt{\frac{1}{N}\sum_{j=1}^{N}\left(\Delta\tilde{z}_j - \Delta z_j\right)^2}}{a} \right\}, \tag{4}$$

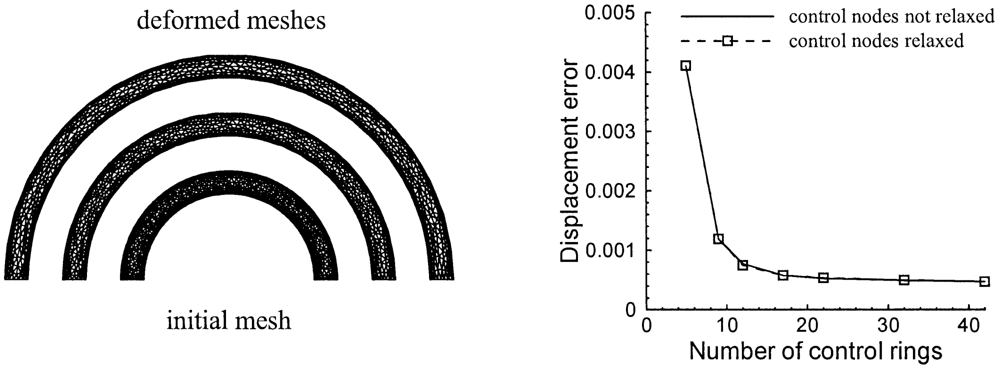

Figure 6. Results of Test 3, in which a variable number of control rings evenly distributed along the tube were moved outward radially from the center of curvature (left panel). The innermost mesh is the initial mesh, and the middle and the outermost one are deformed meshes at the 50th and 100th step of mesh deformation, respectively. Nodal displacement was only specified for control nodes, while the remaining nodes were pnlled outward through fictitious spring forces. Right panel: displacement error, as defined in Equation (4), *vs.* number of control rings for the cases with and without control node relaxation. The difference in error between these two cases is very small, indicating that control node relaxation does not compromise geometric accuracy.

where a is the radius of the tube, $(\Delta\tilde{x}_j, \Delta\tilde{y}_j, \Delta\tilde{z}_j)$ is the nodal displacement for node j computed by the spring analogy, $(\Delta x_j, \Delta y_j, \Delta z_j)$ is the corresponding displacement of node j as determined from the analytical Equations (1) and (2), and N is the total number of surface nodes. Only surface nodes were considered in the error evaluation because surface defects arising from the mesh updating are of primary concern in hemodynamic simulations. Control nodes were included in the summation so that we were able to evaluate the change in the error brought in by relaxing the control nodes (Figure 6, right panel). The quantity in Equation (4) was evaluated after the first step of mesh displacement, which gave values typical for the entire displacement history.

When there were fewer than about 20 control rings, the error was strongly dependent on the number of control rings (Figure 6, right panel). With more than 20 control rings, the displacement error asymptoted to a finite value of around 5×10^{-4}, which was a geometric error arising from the discrete representation of the domain surface. In the same test with a finer mesh containing about 200,000 elements, this asymptotic value was found to be around 6×10^{-5}. Control-node relaxation did not significantly affect the displacement error, while it helped to markedly reduce elemental aspect ratios (see Test 4). These indicated that the number of control rings and the nodal spacing in the mesh were two major factors affecting the displacement error.

A distribution of approximately one control ring per diameter axial distance was found to be adequate to reduce the displacement error to the order of 10^{-4}. This served as guidance in determining the number of control rings needed in updating actual RCA meshes.

3.3. RCA MESH UPDATING

A fully unstructured finite-element mesh was generated on the initial RCA geometry, and the mesh was updated using the above 4-step scheme to follow the dynamic RCA geometry in Test 4. The mesh contained 115,950 elements, and the normalized time-step size was 0.001 T, where T is the cardiac period.

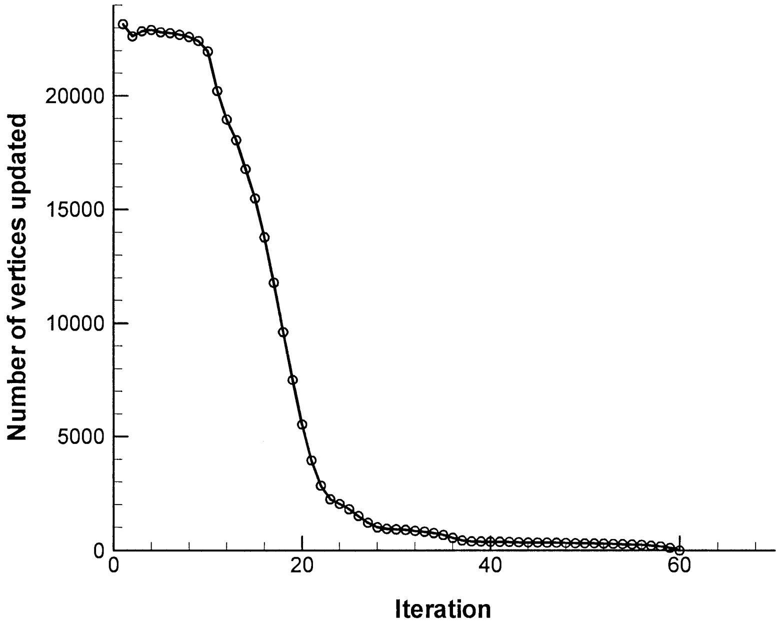

Figure 7. Number of vertices updated at each SOR iteration in Test 4. There were around 25,000 vertices in the mesh. The total number of vertices that needed updating decreased significantly in the first 10 to 20 iterations due to the local convergence check. The resulting CPU-time saving was of the order of 60 %∼70 % compared to that without local convergence check. This figure shows the case for the first time step of vertex updating in a cycle, where there were 60 SOR iterations. At other steps, total number of SOR iterations varied, but the patterns of the curve were similar.

Figure 7 shows the number of vertices updated in successive SOR iterations for this mesh. The number of vertices that needed to be updated decreased significantly within the first 10 to 20 SOR iterations because we imposed the local convergence check in view of the locality of deformations.

At each step of mesh updating, we checked elemental Jacobian values and aspect ratios to see if element inversion or badly-shaped elements were produced (Figure 8.) The results indicate that good element quality was maintained with our mesh-updating scheme.

Mesh velocities were also monitored against marker velocities in order to identify “spurious” node movement. Shown in Figure 9 is a comparison between the maximum axis marker velocity and the maximum mesh velocity at each time step in one cardiac cycle. At any time in the cardiac cycle, the maximum components of mesh velocity were quite close to those of axis marker velocity, suggesting that the mesh was faithfully following the RCA axis motion.

Mesh velocities at three randomly selected nodes were checked in more detail. The nodes were selected from the proximal region, from the middle part of the RCA mesh, and from the distal region. Mesh velocities and accelerations varied smoothly in time, which is important for carrying out flow simulations (Figure 10).

All our test results indicated that the mesh-updating scheme was able to reliably update the FEM mesh, both in terms of mesh quality and smoothness of mesh movement. Similar tests on a range of meshes containing from 50,000 to 400,000 elements gave similar results.

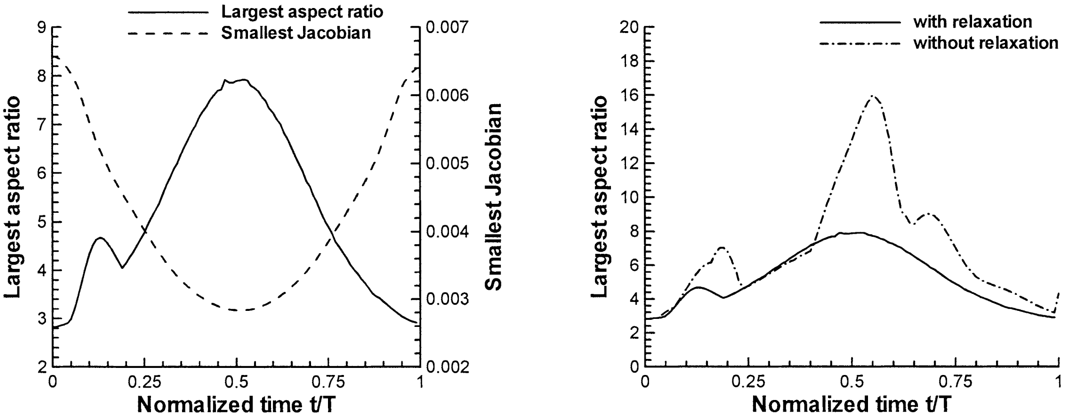

Figure 8. The largest aspect ratio and the smallest elemental Jacobian value of Test 4 as the mesh was updated for one cardiac cycle (left panel). The Jacobian values were normalized by inlet radius cubed (1·91 mm^3), and the time by the cardiac period T. The largest aspect ratio was approximately between 3 and 7, indicating that good mesh quality was maintained through the cardiac cycle. Right panel: largest aspect ratios with and without control node relaxation. The comparison indicates that control node relaxation considerably improves mesh quality. Note different axis scales for left and right panels.

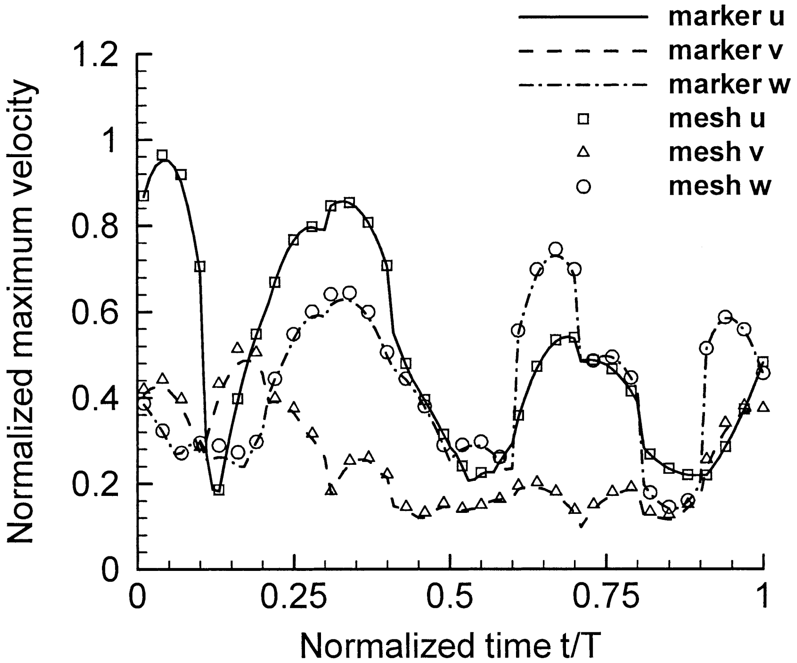

Figure 9. The maximum values of the three components of axis marker velocity and mesh-node velocity for Test Case 4. Differences between the maximum axis marker and mesh-node velocities are not discernable in this plot, indicating that the mesh followed RCA axis closely. Velocity values were normalized by the mean inlet flow velocity, and time by the cardiac period.

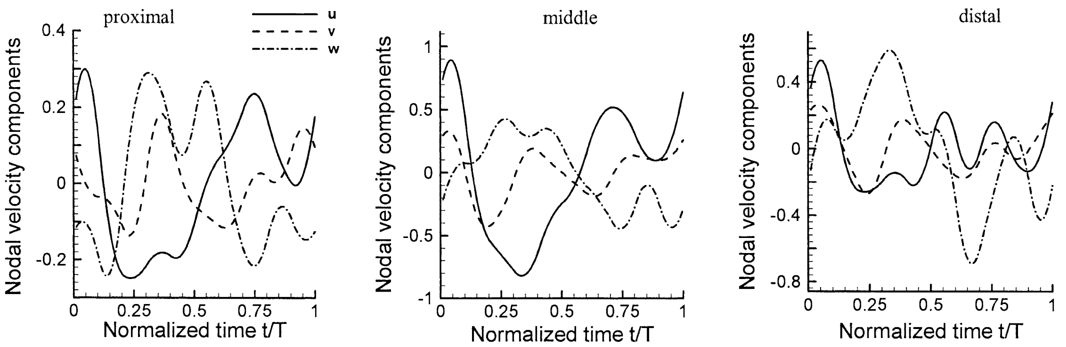

Figure 10. Plot of nodal velocity *vs.* time over one cardiac cycle for three randomly selected nodes in Test 4. Note that the nodal velocities vary smoothly in time. The time is normalized by the cardiac period, and nodal velocity by the mean inlet flow velocity. The scales of the vertical axis are different in three panels.

4. Conclusions

A mesh-updating scheme suitable for finite-element flow simulations in moving arteries with physiologically correct motions is described in this work. The scheme combines a vertex update and a surface spring-analogy process, and was validated through extensive numerical tests. The robustness of the scheme and its capability of handling large deformations suggest its potential for simulating blood flows in human arteries where complete information about wall motion is not available.

Of particular interest for us is to use the mesh-updating scheme for updating FEM meshes generated in a realistic RCA geometry. RCA axis motion and deformation were associated with mesh movement through contours and control nodes, and the control nodes deformed the entire mesh through a four-step scheme that consisted of control-node movement, vertex updating, control-node relaxation, and surface mid-side nodes modification. Good mesh quality was maintained even with large *in vivo* curvature and torsion changes of the RCA. This meshupdating algorithm has been used for subsequent modelling of RCA blood flow [13].

Acknowledgements

We thank Professor M. H. Friedman at Duke University and Dr. Z. Ding at Vanderbilt University for providing us the RCA motion data.

This research has been supported by the Natural Sciences and Engineering Research Council of Canada (CRE); Assistance from an Ontario Graduate Scholarship and Mary I'Anson Gertrude Scholarship (DZ) is gratefully acknowledged.

References

1. Z. Ding and M. H. Friedman, Dynamics of human coronary arterial motion and its potential role in coronary atherogenesis. *J. Biomech. Eng* 122 (2000) 488–492.
2. C. M. Gibson and others, Relation of vessel wall shear stress to atherosclerosis progression in human coronary arteries. *Arterioscler. Thromb.* 13 (1993) 310–315.
3. R. Krams, J. J. Wentzel, J. A. Oomen, J. C. Schuurbiers, I. Andhyiswara, J. Kloet, M. Post, B. de Smet, C. Borst, C. J. Slager, and P. W. Serruys, Shear stress in atherosclerosis, and vascular remodelling. *Semin. Interv. Cardiol.* 3 (1998) 39–44.

4. D. N. Ku, D. P. Giddens, C. K. Zarins, and S. Glagov, Pulsatile flow and atherosclerosis in the human carotid bifurcation. Positive correlation between plaque location and low oscillating shear stress. *Arteriosclerosis* 5 (1985) 293–302.
5. J. T. Batina, Unsteady Euler Airfoil Solutions Using Unstructured Dynamic Meshes. *AIAA J.* 28 (1990) 1381–1388.
6. P. Z. Bar-Yoseph, S. Mereu, S. Chippada, and V. J. Kalro, Automatic monitoring of element shape quality in 2-D and 3-D computational mesh dynamics. *Comput. Mech.* 27 (2001) 378–395.
7. J. Donea, S. Guiliani, and J. P. Halleux, An Arbitrary Lagrangian-Eulerian Finite-Element Method for Transient Dynamic Fluid Structure Interactions. *Comp. Meth. Appl. Mech. Eng.* 33 (1982) 689–723.
8. K. Stein, T. Tezduyar, and R. Benney, Mesh moving techniques for fluid-structure interactions with large displacements. *J. App. Mech.* 70 (2003) 58–63.
9. M. S. Gadala, M. R. Movahhedy, and J. Wang, On the mesh motion for ALE modeling of metal forming processes. *Finite Elem. Anal. Design* 38 (2002) 435–459.
10. D. Zeng, and C. R. Ethier, A semi-torsional spring analogy model for updating unstructured meshes. In: G. E. Schneider (ed.), *Proceedings of the 9th Annual Conference of the CFD Society of Canada.* (5-27-2001). Waterloo: The CFD Society of Canada (2001) pp. 113–118.
11. Z. Ding, H. Zhu, and M. H. Friedman, Coronary artery dynamics in vivo. *Ann. Biomed. Eng* 30 (2002) 419–429.
12. Z. Ding and M. H. Friedman, Quantification of 3-D coronary arterial motion using clinical biplane cineangiograms. *Int. J. Card Imaging* 16 (2000) 331–346.
13. D. Zeng, Z. Ding, M. H. Friedman, and C. R. Ethier, Effects of cardiac motion on right coronary artery hemodynamics. *Ann. Biomed. Eng* 31 (2003) 420–429.
14. R. Krams, J. J. Wentzel, J. A. Oomen, R. Vinke, J. C. Schuurbiers, P. J. de Feyter, P. W. Serruys, and C. J. Slager, Evaluation of endothelial shear stress and 3D geometry as factors determining the development of atherosclerosis and remodelling in human coronary arteries in vivo: Combining 3D reconstruction from angiography and NUS (ANGUS) with computational fluid dynamics. *Arterioscler.Thromb. Vasc. Biol.* 17 (1997) 2061–2065.