Published in final edited form as:
Comput Aided Des. 2012 January ; 44(1): 3–14. doi:10.1016/j.cad.2011.01.004.

Virtual Interventions for Image-based Blood Flow Computation

Guanglei Xiong,
Biomedical Informatics Program, Stanford University, Stanford, CA 94305, USA

Gilwoo Choi, and
Department of Mechanical Engineering, Stanford University, Stanford, CA 94305, USA

Charles A. Taylor
Departments of Bioengineering and Surgery, Stanford University, Stanford, CA 94305

Guanglei Xiong: glxiong@stanford.edu; Gilwoo Choi: giroo@stanford.edu; Charles A. Taylor: taylorca@stanford.edu

Abstract

Image-based blood flow computation provides great promise for evaluation of vascular devices and assessment of surgical procedures. However, many previous studies employ idealized arterial and device models or only patient-specific models from the image data after device deployment, since the tools for model construction are unavailable or limited and tedious to use. Moreover, in contrast to retrospective studies from existing data, there is a pressing need for prospective analysis with the goal of surgical planning. Therefore, it is necessary to construct models with deployed devices in a fast, virtual and interactive fashion. The goal of this paper is to develop new geometric methods to deploy stents or stent grafts virtually to patient-specific geometric models constructed from a 3D segmentation of medical images. A triangular surface representing the vessel lumen boundary is extracted from the segmentation. The diseased portion is either clipped and replaced by the surface of a deployed device or rerouted in the case of a bypass graft. For diseased arteries close to bifurcations, bifurcated device models are generated. A method to map a 2D strut pattern on the surface of a device is also presented. We demonstrate three applications of our methods in personalized surgical planning for aortic aneurysms, aortic coarctation, and coronary artery stenosis using blood flow computation. Our approach enables prospective model construction and may help to expand the throughput required by routine clinical uses in the future.

Keywords

model construction; geometric processing; surgical planning; stent; stent graft; bypass graft; aortic aneurysm; aortic coarctation; coronary artery stenosis; blood flow computation

1 Introduction

Image-based blood flow computation, once emerged, is widely used as a valuable tool for investigating the role of local hemodynamics and the development of vascular diseases, such as atherosclerosis and aneurysms [1,2]. Nowadays, increasing attention has been focused on applications of this tool to evaluate the safety and efficacy of vascular devices [3] and assess or optimize the outcomes of surgical procedures for acquired [4,5] and congenital heart

diseases [6,7]. Considerable insights were gained by studying blood flow through stents in occlusive [8,9] and aneurysmal diseases [10,11] and computing fluid forces on stent grafts [12]. However, many early studies employ idealized arterial and device models, which may be inadequate for replicating realistic hemodynamic conditions. More recent studies have shifted to construct and analyze patient-specific models [13,14], which provide great promises for inferring realistic conditions on an individual basis. Nevertheless, a majority of previous studies employed a limited number of cases stemming from the fact that the model construction process from 3D medical images is still tedious. Furthermore, retrospective studies of the states from existing image data are typically performed to analyze the post-operative effects of physically deployed devices [15,16]. In order for image-based blood flow computation to be used in clinical practice, there is an obvious requirement of prospective analysis to predict the outcomes and compare the benefits and harms of alternative treatment plans [17].

To date, there are two major approaches to construct vascular models from medical images with the ultimate goal of blood flow computation. One is the NURBS (Non-uniform Rational Basis Spline)-based modeling [18,19]. Usually, one parametric surface is reconstructed from sweeping a series of contours by segmenting in 2D over cross sectional images along the path of a vessel branch. Surfaces for multiple branches are then lofted together to represent the entire vascular tree of interest. The other is triangle-based (sometimes polygon-based) modeling, where 3D segmentation is directly applied and triangulated into a single surface [20,21]. The former approach preceded the latter and is quite useful for small vessels in images of poor contrast. The NURBS-based approach shows clear advantages in the case of surgical planning since not only arteries but also deployed devices can be simultaneously modeled and ultimately combined [4]. The benefit is due to the fact that NURBS, a common representation for medical devices, offers a wealth of inherent features, such as interactive editing and robust Boolean operations, of which the triangle-based approach has fewer such tools. However, unless a NURBS representation is necessary in computing blood flow (e.g. isogeometric analysis [22]), it has been considered to be less favorable to the latter, which is more efficient to obtain through fast 3D segmentation, less subjective due to reduced human factors, and of more fidelity to anatomical shapes, especially for large arteries in decent-quality images. Although there are techniques to convert triangle-based surfaces to NURBS-based surfaces [23], they are not widely adopted since the NURBS representation has to be discretized in the subsequent flow computation. On the other hand, the triangle-based representation is more difficult to edit, a critical feature for geometric modification for surgical planning. In order to fill the gap between triangle-based modeling and surgical planning, there is a pressing need for developments of new geometry processing techniques that are efficient, specific to vascular structure, and ideally usable in clinical settings.

In this paper, we present novel geometric methods to deploy stents or stent grafts virtually in triangle-based patient-specific models constructed from direct 3D segmentation of medical images. Two common scenarios are considered. One example is when the deployed device serves as a replacement of the original diseased vessel segment. For example, an endovascular stent graft is deployed in the case of a thoracic aortic aneurysm to support the blood flow and prevent the aneurysm from rupture. Another example is a stent is inserted to hold the occluded coronary artery open and sustain the blood supply to the heart. We propose a novel mesh editing method based on computational geometry that clips the surface of a diseased vessel segment and replaces it with the surface of a deployed device. Techniques to handle bifurcated branches for deployed devices are also developed and further divided into two cases: bifurcated with a small angle and with an arbitrary angle. The other scenario is the deployed device serves as a bypass of the narrowed vessel segment. For example, a bypass graft reroutes the blood around the clogged artery in coronary artery

bypass surgery. In this case, a technique that combines the original surface and the surface of the graft is developed. Our approach allows the user to modify the pre-operative model interactively to incorporate the geometric effects of a virtual stent or stent graft. We also describe a method to map a strut pattern defined in 2D to the stented model. It should be noted that we are here interested in the geometric effects of a deployed state instead of the deployment process itself, which is a challenging and active research topic [24].

The remainder of this paper is organized as follows: Section 2 surveys the related work. Section 3 briefly introduces the methods of model construction and centerline extraction that our approach is based on. Sections 4–9 describe the detailed algorithms of our method, which comprises virtual stenting, handling bifurcations, mapping strut patterns, and combining bypass surfaces. Section 10 presents three clinically-relevant applications of our method. Finally, we conclude and suggest future work in Section 11.

2 Related Work

Image-based geometric modeling is the first and challenging step in personalized blood flow computation, since vascular anatomy varies largely from one subject to another, from healthy to diseased state, and from diseased to surgically altered state. Construction of vascular geometry has shifted from idealized to subject-specific, from individual to group-wise. A number of efficient and automated techniques have been developed to address the increasing demand of computation cost and manual interaction of this shift. However, we note that there has been little work directed towards modification of constructed geometries for the goal of predicting the outcomes of surgical procedures involving deployment of medical devices, particularly stents and stent grafts.

2.1 Construction of vascular geometry

In the following review, we focus on triangle-based modeling since it is the representation we use in this paper. Construction of vascular geometry generally includes two major steps: image segmentation and surface modeling. A thorough review of methods for segmentation of vascular images is beyond the scope of this paper. Instead, we survey some recent work used for blood flow computation. There are two approaches in image segmentation that have been employed: deformable models (also known as active surfaces) and the level set method. In the former approach [25,26], a surface mesh is deformed by internal forces computed from surface features, e.g. curvature, as well as external forces computed from the image, e.g. gradient. This approach has limitations when segmenting large portions of the vasculature because of its inflexibility to handle complex shape and topology changes, although merging of smaller segmentations is possible [27]. The level set method overcomes this difficulty by implicitly representing the surface as the zero level of a higher dimensional function [28] and has been successfully applied to segment vascular images [20,23,29] using variants of forcing terms, e.g. [30]. Since the initialization of the level set function, ideally close to the target segmentation, is crucial to robustness and convergence, it is usually achieved by means of a fast marching method [28] due to its computational efficiency. An improved initialization strategy, known as “colliding fronts”, is also proposed in [31].

Once the final segmentation is obtained, it is converted into a triangulated surface representation, generally using the Marching Cubes [32] or Marching Tetrahedra [33] algorithms. Another algorithm which outputs triangulation of a proven better quality can also be utilized [34]. However, this initial surface mesh cannot be directly used for flow computation for multiple reasons. Several mesh processing operations based on computational geometry, such as smoothing, pruning, elongation, trimming, circularization, and remeshing, have been developed [20,23,31] to adjust the surface mesh to yield the final vascular domain, which will be eventually volumetrically meshed for computing blood flow.

Thanks to the availability of commercial (e.g. Mimics) and academic (e.g. vmtk) software packages, many, if not all, of the above techniques can be readily accessed and greatly facilitate the construction of vascular geometry. Meanwhile, we should emphasize that the geometric construction is not yet fast enough to be used in clinical practice, there is still plenty of room for improvement for easier interaction and more automation.

2.2 Modification of vascular geometry

Compared to the construction of vascular geometry, there have been only a few studies devoted to the modification of subject-specific vascular geometry, especially towards the goal of predicting the outcomes of surgical procedures. We note two recent studies that are most relevant to this paper. The first study proposed an objective approach to obtain plausible pre-aneurysmal states from digital removal of cerebral aneurysms of saccular shape [35]. The approach relied on computing the Voronoi diagram of a vascular surface. The aneurysmal portion is removed by clipping the corresponding Voronoi points. Guided by a centerline, a new set of Voronoi points are generated by interpolations from remaining points within a certain distance. Finally, all Voronoi points are transferred back to a new surface by evaluating on a fine orthogonal grid and then performing the Marching Cubes algorithm. This method is clearly very effective to remove the aneurysm. However, the need to regenerate the whole surface may be cumbersome for a large geometry since the non-clipped portion may have already been smoothed or trimmed.

Another study described a new technique to map different stent designs to a patient-specific geometry in endovascular treatment of cerebral aneurysms [36]. To avoid the difficulty to create body-conforming meshes for patient-specific models with endovascular devices (immersed in the blood stream), a hybrid approach of body-conforming meshes for the vessel and adaptive-refined mesh embedding for the stent is proposed. The geometry of the stent is implicitly represented using a series of overlapping spheres and then virtually mapped to a deformed cylindrical surface, which is obtained by solving the elasticity equations. The approach provides an important way to compute the blood flow in patient-specific geometry with embedded devices. However, the need to modify the fluid solver for incorporating mesh embedding may not be necessary if the device is only partially embedded in the flow stream and purely body-confirming meshing is possible, as will be considered in this paper.

3 Vascular Model Construction and Centerline Extraction

Subject-specific vascular models are constructed from 3D Computed Tomography (CT) or Magnetic Resonance (MR) images using a previously reported method [37]. Briefly, the image is first preprocessed to reduce noises and enhance contrasts. The vessel lumen is then segmented using region-based and edge-based level set methods [13]. A triangular surface mesh is generated from the segmentation using the method proposed in [34], which produces provably good sampling and meshing. To obtain faithful geometry and ensure stability for further computation, several geometric processing techniques, such as smoothing, trimming, pruning, circularization and elongation, are performed to yield the final model when applicable. This geometry serves as the pre-operative model considered in this paper.

A method to extract the multi-branch centerline from the constructed model is then utilized. Two Euclidean distance transforms of the model: distance to inlet/outlet and distance to vessel wall, are computed discretely using the fast marching method on the volumetric grid of the model. Based on the former distance, the model is divided into regions. Centerline nodes with the maximum of the latter distance in every region were selected and connected to yield a raw centerline using adjacency relations among regions. In this way, the multi-

branch nature of the centerline is attained and the connectivity of the centerline is equivalent to the connectivity of the model. The centerline was then refined using cubic B-splines and re-sampled to reach a higher quality. We note the endpoints of the centerline exactly correspond to the inlets/outlets of the model. Figure 1 demonstrates a 3D image of the abdominal aorta with an aneurysm, its geometric model, and centerline.

4 Problem Statement

Our goal is to replace the diseased portion (e.g. stenosis, aneurysm, etc.) of a pre-operative model with a deployed device (stent or stent graft). The consequent post-operative model will serve as the domain for the purpose of surgical planning using blood flow computation. There are several cases with increasing levels of complexity considered in this paper.

Case 1

As a first and simplest example illustrated in Fig. 2(a), a stent graft (depicted as a dark region) is deployed to constrain the blood flow and strengthen the weakened wall of a thoracic aortic aneurysm. It is noted that there are two fixation zones (depicted in gray) at both ends to attach the graft to the vessel wall. In the geometric perspective, the fixation zone can be treated as a transition region from the subject-specific vascular geometry to the deployed device geometry. The latter can be considered as a tubular shape as a result of self and/or balloon expansion.

Case 2

In Fig. 2(b), a stent graft is deployed in the treatment of an abdominal aortic aneurysm. This is different from the previous case in terms of device topology, i.e. the graft has a bifurcated shape to divide the blood from the aorta and supply it to two iliac arteries. Noticeably, the angle between contralateral limbs is kept as minimal.

Case 3

In Fig. 2(c), a stent is deployed in a segment of coronary arteries to widen the stenosis and restore the blood flow to the heart. This procedure is referred to as percutaneous coronary intervention (PCI). An interesting and challenging situation in this case is that there is a side branch originating from the main vessel just below the stenosis. Deployment of another stent in the side branch is an option to avoid its blockage at the ostium by the first stent. The combination of both stents again leads to a bifurcated shape. However, unlike the previous case, the angle of two downstream branches can be arbitrary and dependent on the state before stenting.

Case 4

In Fig. 2(d), a procedure referred to as a coronary artery bypass graft surgery (CABG) is shown as an alternative to PCI to treat coronary artery stenosis, in which a segment of vessel elsewhere in the patient’s body is grafted to bypass the stenosis to improve the blood supply to the heart. Strictly speaking, the graft in this case is not a device. But we believe it is an important problem and fits well with our purpose.

Finally, both artificial stent and stent graft typically make use of wires to support its physical integrity, which present as strut patterns on the domain of blood flow in the deployed states. They should be considered if near-wall hemodynamics is of interest because the latter is believed to be substantially affected by various designs of strut patterns [38].

5 Stenting with a Tubular Shape

The input pre-operative model is represented as a triangular mesh. We expect the model has been processed (smoothing, trimming, etc.) and ready for flow computation. Virtual stenting is locally applied to the diseased portion while keeping the rest of the model intact. In this way, repetitive processing on the post-operative models is avoided. More importantly, it makes further interpretations of the results from flow computation more straightforward because the variation from pre-operative to post-operative or between multiple alternative operations is only due to local geometric differences introduced by deployed devices.

5.1 Splitting algorithm

In order to remove the diseased portion, it is necessary to split the triangular mesh at the borders between the diseased and healthy portions (denoted as locations 1 and 2 in Fig. 2(a)). The mesh is cut through using a plane Π, defined with a point $\boldsymbol{p}_\Pi$on the mesh and a normal direction $\boldsymbol{n}_\Pi$. Without loss of generality, we assume $\boldsymbol{p}_\Pi$ is one of the vertices of the mesh. Otherwise, it can be made to be one by dividing its incident triangle(s). The normal direction $\boldsymbol{n}_\Pi$ is automatically selected by minimizing the cross-sectional area S of the plane cutting the mesh, i.e. $\boldsymbol{n}_\Pi = \text{argmin}_{\boldsymbol{n}}\, \boldsymbol{S}\,(\boldsymbol{n})$ using a conjugate gradient method with guaranteed descent [39]. To cut the mesh, we start at $\boldsymbol{p}_\Pi$, and then traverse circumferentially on the mesh within the plane until $\boldsymbol{p}_\Pi$ is reached again. In order to avoid numerical round-off error and guarantee returning to $\boldsymbol{p}_\Pi$, exact geometric predicates are used in our algorithm [40].

The basic idea is to move vertices in the vicinity to the plane during the traversal and thus obtain a sequence of segments within the plane that can be used to split the mesh. Let$\boldsymbol{p}_k$ be the current vertex encountered in the k-th step and moved to the plane, where $\boldsymbol{p}_0 = \boldsymbol{p}_\Pi$ and $\boldsymbol{p}_{k-1}$is on the plane. First, we determine which is the next triangle cut by the plane by examining all triangles incident to $\boldsymbol{p}_k$. To force the traversal in one direction, the previous vertices are marked with the step number at which they were encountered and the triangles which have vertices marked with previous step numbers are skipped. Figure 3(a) illustrates a typical case in which the triangle $\Delta\boldsymbol{p}_a\boldsymbol{p}_k\boldsymbol{p}_b$ is to be considered. The next vertex $\boldsymbol{p}_{k+1}$ to move to the plane is selected from two vertices $\boldsymbol{p}_a$ and $\boldsymbol{p}_b$. To minimize geometric changes, we choose the one with smaller distance to the point $\boldsymbol{p}_i$, which is the intersection point of the edge $\overline{\boldsymbol{p}_a\boldsymbol{p}_b}$ with the plane. Then, the chosen vertex (e.g.$\boldsymbol{p}_a$) is moved to $\boldsymbol{p}_i$, i.e. $\boldsymbol{p}_{k+1} = \boldsymbol{p}_a = \boldsymbol{p}_i$. Note there is a special but simpler case where $\boldsymbol{p}_a$ or $\boldsymbol{p}_b$ is already on the plane (rarely in the middle of the traversal and always when returning to $\boldsymbol{p}_0$) and do not need to move. Furthermore, we consider two extreme cases that have to be carefully handled for the purpose of robustness, as illustrated in Fig. 3(b) and (c). Degenerate triangles with zero area can be generated when attempting to move one vertex from a triangle to the plane while the other two vertices are already on the plane. In Fig.3(b), $\boldsymbol{p}_a$ is about to move to the plane based on our rule. However, the triangle $\Delta\boldsymbol{p}_{k-1}\boldsymbol{p}_k\boldsymbol{p}_a$ will then be degenerate. To avoid this difficulty, an edge flipping is performed prior to moving $\boldsymbol{p}_a$, i.e. replace the edge $\overline{\boldsymbol{p}_{k-1}\boldsymbol{p}_a}$ with $\overline{\boldsymbol{p}_k\boldsymbol{p}_c}$. Likewise when returning to $\boldsymbol{p}_0$ as shown in Fig.3(c), the degenerate triangle $\Delta\boldsymbol{p}_a\boldsymbol{p}_0\boldsymbol{p}_1$ will also be degenerate, which could be eliminated by an edge flipping. Unlike the previous case, the edge flipping is done after moving $\boldsymbol{p}_a$ because the degeneration cannot be detected until reaching to $\boldsymbol{p}_0$. Finally, the mesh is split along the contour of edges linking the sequence of vertices that have been encountered during the traversal.

5.2 Morphing algorithm

On one side of the plane that split the mesh, we need to generate a transition region (shown in gray in Fig. 2(a)) that smoothly varies from the device geometry of a tubular shape to subject-specific geometry, as depicted in Fig. 4. Because the planar contour from splitting the mesh is generally not circular, the first step is to make it circular. From the vertices on

the contour on the splitting plane Π_1, the centroid $\boldsymbol{c}_1$ and radius r_1 of the circle are computed as the averaged coordinate and the averaged distance to the centroid or prescribed from device geometry in the deployed state, respectively. Next, all vertices are moved radially to the circle. Let the user-provided length of the transition region be L_t. Starting from $\boldsymbol{c}_1$, we find the point $\boldsymbol{c}_2$ on the centerline (shown as a dotted line) so that the distance along the centerline between $\boldsymbol{c}_1$ and $\boldsymbol{c}_2$ is L_t. A second plane Π_2 is obtained whose normal direction is determined as before by minimization of cross-sectional area. The surface between two planes is treated as the transition region. In order to vary smoothly from a circle on Π_1 to a non-circular shape on Π_2 and beyond, the final surface is weighed between the original surface and a tubular shape along the centerline, where the weight is determined by the distance to Π_1. This is achieved by modifying the coordinate of each vertex $\boldsymbol{p}$ using

$$\boldsymbol{p} \leftarrow \frac{\|\boldsymbol{c}_p - \boldsymbol{c}_1\|}{L_t} \cdot \boldsymbol{p} + (1 - \frac{\|\boldsymbol{c}_p - \boldsymbol{c}_1\|}{L_t}) \cdot \left(\boldsymbol{c}_p + \frac{\boldsymbol{p} - \boldsymbol{c}_p}{\|\boldsymbol{p} - \boldsymbol{c}_p\|} r_1\right)$$

, where $\boldsymbol{c}_p$ is the point on the centerline which is the closet to $\boldsymbol{p}$. As a consequence, the vertices on Π_1 is on the circle, whereas those exactly on Π_2 stays at their original locations, if any.

5.3 Bridging algorithm

After the surface corresponding to the diseased portion (depicted as dark in Fig.2 (a)) is removed, two open circular contours are left. On either contour, the vertices are redistributed evenly. Our task is to generate a tubular surface bridging them. First, a centerline of the removed surface is calculated with its endpoints exactly located at the centroids of the contours, which is ensured by our centerline extraction algorithm. The centerline is then modified for the purpose of surgical planning using heuristics although some optimization-based techniques are possible [41]. We only consider the former for the scope of this paper. Following the centerline, we create a series of contours with their centroids on the centerline. Each contour is composed of evenly distributed vertices. The surface is completed using triangular strips connecting vertices from adjacent contours. Because the radii of the two ending contours and the numbers of vertices on them are generally different, we simply choose to vary them linearly for intermediate contours. To orient and position the contours, a normal direction and a reference vertex need to be determined for each contour. As in Fig. 5, let $\boldsymbol{p}_i^0$ and $\boldsymbol{c}_i$ denote the reference vertex and the centerline node at the i-th contour C_i, respectively. The normal of the $(i-1)$-th contour is C_{i-1} is $\vec{\boldsymbol{n}}_{i-1} = (\boldsymbol{c}_i - c_{i-1})/\|\boldsymbol{c}_i - c_{i-1}\|$. Starting from one of the ending contours, we arbitrarily select its reference vertex $\boldsymbol{p}_i^0$. Define $\vec{x}$ and $\vec{y}$ as local axes of a contour with $\vec{n} \perp \vec{x}, \vec{y}$ and $\vec{x} = (\boldsymbol{p}^0 - \boldsymbol{c})/\|\boldsymbol{p}^0 - \boldsymbol{c}\|$. The reference vertex $\boldsymbol{p}_i^0$ can be iteratively determined by

$$\boldsymbol{p}_i^0 = \boldsymbol{c}_i + r_i \cdot \vec{\boldsymbol{n}}_{i-1} \times \vec{\boldsymbol{x}}_{i-1} \times \vec{\boldsymbol{n}}_i$$

, where r_i is the radius of the i-th contour. The reference vertex of the other ending contour is chosen to be the vertex closest to that of the previous contour. Let N_{i-1} and N_i be the numbers of vertices on two adjacent contours. Since they are usually not equal, a triangular strip cannot be formed trivially. We employ the angular distance $\theta_i^k = 2\pi/N_i$ of a vertex $\boldsymbol{p}_i^k$ with respect to the reference vertex $\boldsymbol{p}_i^0$ to construct the triangular strip incrementally. Denote the current vertices on both contours considered as k_{i-1} and k_i, respectively. Initially, k_{i-1} =

$k_i = 0$. If $\theta_i^{k_i+1} < \theta_{i-1}^{k_{i-1}+1}$, the triangle $\Delta \boldsymbol{p}_{i-1}^{k_{i-1}} \boldsymbol{p}_i^{k_i} \boldsymbol{p}_i^{k_i+1}$ is added and $k_i \leftarrow k_i + 1$. Otherwise, the triangle $\Delta \boldsymbol{p}_{i-1}^{k_{i-1}} \boldsymbol{p}_i^{k_i} \boldsymbol{p}_{i-1}^{k_{i-1}+1}$ and $k_{i-1} \leftarrow k_{i-1} + 1$.

5.4 Complete algorithm

With the above algorithms described, we introduce the complete algorithm for stenting with a tubular shape as follows:

1. Input a model and identify the diseased region that is to be modified by virtual stenting.
2. Split the input model (at location 1 and 2 in Fig. 2(a)) into three parts.
3. Obtain the centerline for each part.
4. Remove the diseased part, leaving a gap in between.
5. Circularize open contours on the other two parts. Smooth transitions are then morphed.
6. Bridge the gap using a tubular surface as the stented region.
7. Output the stented model.

The output model of our algorithm is guaranteed to be “water-tight”. In addition, no further remeshing is necessary since the quality of triangles in the stented region is good enough for the flow computation.

6 Stenting with a Bifurcated Shape of a Small Angle

We now consider the device of a bifurcated shape deployed around a bifurcation. It is useful primarily in surgical planning for endovascular treatment of abdominal aortic aneurysm with stent grafts. In particular, we are only dealing with the case in which the angle between two child branches is minimally small and leave the case with an arbitrary angle to the next section. In addition, the calibers of two child branches are often similar.

6.1 Bisecting algorithm

The algorithms described in the previous section are directly applicable with few modifications in the parent vessel and child vessels far apart from the bifurcation. However, a technique to construct the surface at the bifurcation (illustrated in Fig. 6(a)), as a region dividing the parent vessel into two child branches, needs to be developed. We propose the bisecting algorithm for this task. Suppose the open contours from the parent vessel and two child vessels have been elongated to the planes Π_1 and Π_4 along the centerlines. The idea is to create a series of contours that gradually vary from one circle in the parent vessel to two circles in either child vessel. The minimum bounding circle for each individual contour is placed along the centerline and its position and radius is determined using a similar technique as in the bridging algorithm. At the plane Π_1 in Fig. 6, the contour is originally a circle. From Π_1 to Π_3, The contour starts to cleave in the middle and its shape, circumscribed by its minimum bounding circle, is produced by a union of two overlapping circles. To prepare for the division, the circles move further apart as closer to Π_3. At Π_3, the bisecting line is inserted to divide the contour into two. Meanwhile, the centerline guiding the parent vessel is switched to two centerlines, either for one child vessel. The child contours are completely separated, become two circles, and move apart from each other from Π_3 to Π_4. The surface is completed by connecting them with triangular strips as before. Instead of a single reference vertex, two reference vertices (*a* and *b* in Fig. 6(b)) are used to divide the contour in half. Triangles are independently generated in either half. It is worth

mentioning that the triangle strips between the contour on Π_3 and two contours of its immediate adjacency towards Π_4 is special. The bisecting line thus corresponds to two minor sectors: a' to b' and $a''$to b', yielding two inner triangle strips, as well as two outer triangle strips for major sectors.

6.2 Complete algorithm

We modify the previous algorithm for a tubular shape and the complete algorithm for stenting with a bifurcated shape of a small angle as follows:

1. Input a model and identify the diseased region that is to be modified by virtual stenting.
2. Split the input model (at location 1, 2 and 3 in Fig. 2(b)) into four parts.
3. Obtain the centerline for each part.
4. Modify the centerline for the diseased part to that of the device.
5. Remove the diseased part, leaving a gap in between.
6. Circularize open contours on the other three parts. Smooth transitions are then morphed.
7. Elongate the open contours from the three parts along the centerlines closer to the bifurcation.
8. Close the rest of the gap using the bisecting algorithm.
9. Output the stented model.

7 Stenting with a Bifurcated Shape of an Arbitrary Angle

Another case for stenting with a bifurcated shape involves an arbitrary angle between two child branches. It mostly happens when the diseased region is close to the bifurcation and a side branch comes off the main vessel, e.g. in coronary arteries. In addition, the caliber of the side branch is smaller than that of the main vessel. It is often necessary to deploy two stents: one for the main trunk and the other for the side branch. Thus, their combination is of a bifurcated shape.

7.1 Joining algorithm

As in the bisecting algorithm, suppose the open contours from upstream and downstream of the main vessel and the side branch have been elongated to the planes Π_1, Π_2 and Π_3 along the centerlines as shown in Fig. 7. The gap of the main trunk between Π_1 and Π_2 can be simply closed by the bridging algorithm along the centerline. The open contour at Π_3 from the side branch is then further extended into the main trunk and closed with a planar triangulation at Π_4. A second triangular mesh is now obtained and need to be merged with the mesh of the main trunk. This becomes a problem of the Boolean operation, i.e. union, on meshes. We adopt an efficient algorithm based on Nef polyhedron for this task, which guarantee robustness and correctness [42]. Another relevant and good feature of this algorithm is that it maintains the geometry for all triangles except at the intersection. However, the quality of new triangles at the intersection is unsuitable and thus improved by local surface simplification using edge collapse [43]. A further smoothing of the intersection region may be considered if it is too sharp.

7.2 Complete algorithm

The final algorithm for stenting with a bifurcated shape of an arbitrary angle can be modified from that for a small angle, by substituting Step 8 as follows:

8 Close the gap for the main trunk using the bridging algorithm and join the side branch to the main trunk.

8 Intervention with a Bypass Graft

The last case we consider is the virtual intervention with a bypass graft. It is an alternative way to treat the vascular disease by not fixing the focal lesion itself (e.g. coronary stenosis) but bypassing the blood flow with a graft. In terms of the algorithm, it is actually simpler than the previous cases since the diseased portion of the model is kept intact and only the graft needs to be merged with the model. Two ending contours are located at the desired spots and immersed in the input model to ensure a proper connection. A centerline between them is specified to guide the graft. The bridging algorithm can be then used for generating the mesh for the graft and the two ending contours are closed by planar triangulations. Finally, the graft is merged with the model by using the joining algorithm.

9 Stenting with Strut Pattern Mapping

To make the post-operative model more realistic, the strut pattern of a deployed stent or stent graft can be further placed on the stented model to represent the geometric effects of struts. Because there are various designs of strut patterns from many stent and stent grafts, it is impractical to define and assign the patterns analytically. Furthermore, different patterns generally exhibit when a given stent is deployed in different patients, since the stent is forced to adapt to patient-specific vascular geometry (e.g. curvature). We propose a novel technique to assign an arbitrary strut pattern on the patient-specific geometry. The central idea is to map a strut pattern in 2D to the stenting region of the triangular surface mesh in 3D by rolling it up along the centerline. The strut pattern is defined in the 2D image whose intensity profile indicates the strut geometry (width and thickness). Pixel values are non-zero for the strut, zero for the background. To avoid aliasing artifacts and thus toothing effects on the mesh, the image should have a sufficient spatial resolution, i.e. the pixels at the strut should be dense enough to represent the width of the strut. We determine the number of pixels across the width of the strut should be at least 10 from our experience. The image should also have proper boundary conditions to reflect the periodicity of strut patterns.

9.1 Coordinate transformation

As shown in Fig. 8, points in the image are described by 2D Cartesian coordinates (x, y), whereas points on the triangular surface are described by 3D cylindrical coordinates (r, ϕ, z). Individually, the radial distance r, the azimuth ϕ, and the longitudinal distance z of a point is given with respect to its closest centerline node, the local $\vec{x}$ axis, and the centerline node at one end of the stenting region, respectively. To map the strut pattern, a correspondence between homologous points in (x, y) and $(r, \phi\ z)$ needs to be established. Instead of the forward mapping, we define the mapping inversely, i.e. from $(r, \phi\ z)$ to (x, y) because it can ensure a complete and non-overlapping coverage of the stenting region. The transformation then becomes $x=\frac{\phi}{2\pi}W, y=\frac{z}{L}H$, where W, H, L are the image width, height, and the total centerline length of the stenting region, respectively.

9.2 Refinement algorithm

Because the triangle size of the surface mesh is often insufficient to resolve a typical strut width, mesh refinement is necessary to represent the shape of the strut. Rather than refining the stent graft portion globally, it is more favorable to refine the mesh locally along the strut path to avoid excessively increasing the number of mesh elements and thus the computational cost. We adopt the well-known $\sqrt{3}$-subdivision scheme since it allows slower increase of the mesh complexity and local adaptive refinement [44]. The key question is to

determine which triangles to select for refinement. Through coordinate transformation, the triangle is mapped in the 2D image plane as illustrated in Fig. 9. The triangle is selected for refinement if it intersects with any of strut pixels, which has nonzero intensities. Apparently, it is sufficient to consider only pixels in the bounding box of the triangle (dashed line in Fig. 9). There are three cases for the intersection to occur: (a) Any vertex of the triangle is inside the strut pixel (e.g. rectangle a); (b) Any vertex of the strut pixel is inside of the triangle (e.g. rectangle b); (c) Any edge of the triangle intersects with any edge of the strut pixel (e.g. rectangle c), which is simpler by noting pixels are axis-aligned rectangles.

9.3 Displacement algorithm

Furthermore, vertices within the width of the strut are displaced to represent strut thickness characterized by the intensity of the image. The strut is modeled as concavity on the mesh since the stent is normally embedded in the domain of blood flow. After refinement, the vertices are displaced according to the interpolated intensity of its mapping in 2D. The direction of the displacement of a vertex is towards its closest node on the centerline. Finally, a smoothing of the displaced vertices (based on curvature flow [45]) is helpful to reduce the toothing effects, although optional if the resolution of the image is sufficiently high.

9.4 Complete algorithm

1. Input a stented model and an image of the strut pattern.
2. Extract the centerline of the stented region.
3. Refine the mesh along the struts for a user-specified number of steps.
4. Displace the vertices on the struts.
5. Output the final model.

It should be noted that the proposed algorithm applies to the cases of bifurcated shapes with both small and arbitrary angles. In the former case, the stented region is still of a tubular shape if a single strut ring does not span on both parent and child vessels. In the latter case, the mapping is first applied on the main trunk and then the side branch is joined and mapped with a strut pattern.

10 Results

We demonstrate that our virtual stenting techniques can be applied to construct geometric models with deployed stents or stent grafts. Four examples include thoracic aortic aneurysm, abdominal aortic aneurysm, aortic coarctation, and coronary artery stenosis. The images (CT or MR) in these examples are from de-identified pre-existing imaging studies at Stanford University Medical Center and utilized to construct the pre-operative models using our aforementioned methods. In each example, we also supply the clinically-relevant results of flow computation using these models. In all examples, appropriate boundary conditions were used and tuned to match measured or physiological flow and pressure. It is assumed that blood behaves as a Newtonian fluid and the vessel walls are rigid. Blood flow is solved using a custom stabilized finite element solver developed in our group.

10.1 Thoracic and Abdominal Aortic Aneurysms

An aortic aneurysm is a dilation of the aorta, commonly found in the thoracic and abdominal sections. It is usually associated with an underlying weakness of aortic walls, in which biomechanical factors are hypothesized to play an important role. Endovascular treatment for thoracic and abdominal aortic aneurysm using stent grafts provides an alternative and

favorable way to open surgical repair. However, it has long-term risks of complications including stent graft migration, i.e. movement of the graft away from the desired location. The cumulative effect of net forces (tensile forces and shear forces) acting upon the graft is thought to contribute to the migration. Here, we quantify shear stress and net forces in pre-operative and post-operative models using virtual stenting and blood flow computation.

Figure 10(a) demonstrates the steps to virtually deploy the stent graft as a tubular shape and map strut patterns to a model of thoracic aneurysm. The actual meshes close to the distal end of the stent graft are displayed in Fig. 11. They are the pre-operative, the results of splitting, morphing, and bridging algorithms, respectively. In Fig. 10(b), the stent graft with the bifurcation of a small angle is deployed in the model of abdominal aneurysm. Notice that the strut patterns are hypothetical in both cases.

In Fig. 12, we show the mean wall shear stress (MWSS) of the pre-operative model and post-operative models with and without strut patterns, which is computed as the time-averaged amplitude of shear stress on the wall. For both cases, low MWSS are observed in the aneurysm regions. After virtual stent grafts are deployed, MWSS is higher in the same regions due to reduced vessel caliber and increased blood velocity.

In Fig. 13, we show the net forces on the aneurysms and stent grafts (gray regions), which are calculated as the surface integral of tensile and shear stresses. The arrows depict the forces at peak systole. In both cases, the forces are directed radially outwards with respect to curved paths due to the area difference between interior and exterior led by the curvature. The magnitude of the force is smaller in the stent graft than that in the aneurysm due to reduced areas. This holds true for the whole cardiac cycle as shown in the plots. The strut patterns do not have a significant impact on the magnitudes of the forces as their contributions to the surface integration are minimal.

10.2 Aortic Coarctation

Aortic coarctation is a congenital defect of the narrowing of the aorta. When it occurs, the heart must pump harder to force blood through the narrow part of the aorta. It manifests as an increased pressure drop across the coarctation. As an alternative treatment option to surgery, balloon angioplasty with stenting is often performed for children and adults with native coarctation or recoarctation after surgery. Using virtual stenting and blood flow computation, we here compare the pressure and velocity fields before and after stenting for a quantitative assessment of the benefits of stenting.

Figure 14 shows the pre-operative and post-operative models of aortic coarctation. Similar to the aneurysm cases, the stented region is a tubular shape. The difference is the deployed stent dilates the vessel to relieve the obstruction. In Fig. 15, we show the computed pressure and velocity fields at systole before and after stenting. It is clear that the mean pressure gradient dramatically decreases by 38.4mmHg from pre-operative to post-operative. In addition, the narrow flow jet directed towards the vessel wall is widened and weakened. Consequently, the vorticity and fluctuation of the flow downstream is reduced. These results suggest stenting may benefit the patient hemodynamically and can be a favorable treatment option.

10.3 Coronary Artery Stenosis

Stenosis in coronary artery occurs when plaque builds up and constricts the blood flow to the heart. As a physiological gold standard of diagnosis, Fractional Flow Reserve (FFR) is calculated as the ratio of pressure at a distal location to the pressure in the ascending aorta. A stenosis with FFR less than 0.75 is considered as a significant lesion that needs to be intervened [46]. Using virtual stenting and blood flow computation, we here quantify the

FFRs of pre-operative and two post-operative states: stenting in PCI and bypass grafting in CABG.

Figure 16 illustrates the pre-operative and post-operative models of coronary artery stenosis. No strut pattern is mapped since the FFRs and in turn the pressures are influenced essentially by changes in caliber and insignificantly by strut patterns [47]. In Fig. 16(b), there are two branches coming off the main vessel adjacent to the stenosis. Therefore, the stenting region is created with a bifurcation with an arbitrary angle. In Fig. 16(c), a bypass graft is placed between the aorta and the vessel downstream of the stenosis. In Fig. 17, we show the simulated pressure field and calculated FFR values for all three models. The stenosis in the pre-operative model is significant since its FFR is less than 0.75. In the two post-operative models considered, both are improved with restored blood flow. It should be noted that the results are dependent on geometric specifications of the stent (e.g. diameter) and bypass graft (e.g. diameter and route).

For reference, the mesh statistics of all models considered in this section are listed in Table 1.

11 Conclusion and Future Work

We have presented a virtual intervention method that can be used in personalized surgical planning using blood flow computation. The method aims to modify a patient-specific geometric model represented as a triangular mesh to incorporate the deployed device. The modification is achieved by clipping the diseased portion of the model and bridging the gap with the device surface guided by a centerline. We have further extended the algorithm to handle devices with bifurcated shapes with either small or arbitrary angles that are very common in the complex vasculature. Next, an algorithm to address another important scenario of bypass grafting is also described. Finally, we describe a method to map a given strut pattern defined in 2D to the stented model. We demonstrate our methods can be applied in clinically-relevant surgical planning for aortic aneurysm, aortic coarctation and coronary artery stenosis. Through these applications, we call for developing model construction techniques for a shift from retrospective to prospective analysis, which could make more impact in clinical practice.

This work can be extended in several directions. First, the proposed methods do not model the exact device geometry after deployment but instead can be used to provide plausible post-operative configurations that may help to compare against the pre-operative state or among alternative post-operative states. Compared to the purely geometric approach as in this paper, the former may be partially realized by solving the biomechanical process. Realistic physical effects, such as stent foreshortening or elongation due to deployment, as well as deformations of stent wires and cells, can be explicitly captured. It should be noted that numerous assumptions are required to be made considering the complexities and uncertainties of interactions among blood, vessel wall and device, thus the accuracy of this approach can still be limited. In addition, there are also challenges in predicting long-term shape changes due to vascular remodeling and adaption. Therefore, a purely geometric approach should be useful and can be an essential first step in many cases. Second, the specification of the final device geometry is currently an input to our methods as the design parameters of the device (e.g. the dimension of a stent) or part of the surgical plan to be evaluated (e.g. the location of the deployed stent and the path it follows). More image-based guidance can be provided. For instance, the lumen boundary of an aneurysm and the feasible space for a coronary bypass are given to constrain the geometry of the device. Third, the applications of our approach for other parts of the vascular system (e.g. covered stents or stent grafts for carotid artery, pulmonary artery, etc.) can be studied, in which development

of new techniques for more surgical procedures may be required. For example, the current methods cannot preserve the patient-specific vascular geometry opposite to the aneurysm orifice in the case of a saccular aneurysm. Additionally, our approach cannot handle porous stents mostly used in cerebral aneurysms for which it is necessary to keep the outer surface representing the aneurysm sac and mesh in between the stent wires and the aneurysm dome. Finally, the importance of user interfaces of model construction for clinical uses is often underappreciated and it is a big challenge to make it user-friendly, minimally time-consuming, and guaranteed robust.

Acknowledgments

Guanglei Xiong was supported by a Stanford Graduate Fellowship. This work was supported in part by the National Institutes of Health under Grant P50 HL083800. The authors acknowledge Dr. Andrea Les, Dr. Tina Morrison, and Jessica Shih for collecting the image data used in this work. We acknowledge the National Science Foundation Grant No. CNS-0619926 for computer resources. We also thank the anonymous reviewers for their helpful comments.

References

1. Steinman DA, Taylor CA. Flow Imaging and Computing: Large Artery Hemodynamics. Annals of Biomedical Engineering. 2005; 33(12):1704–1709. [PubMed: 16389516]
2. Taylor CA, Steinman DA. Image-Based Modeling of Blood Flow and Vessel Wall Dynamics: Applications, Methods and Future Directions. Annals of Biomedical Engineering. 2010; 38(3): 1188–1203. [PubMed: 20087775]
3. Zarins CK, Taylor CA. Endovascular Device Design in the Future: Transformation from Trial and Error to Computational Design. Journal of Endovascular Therapy. 2009; 16 Suppl 1:12–21.
4. Wilson NM, Arko FR, Taylor CA. Predicting Changes in Blood Flow in Patient-Specific Operative Plans for Treating Aortoiliac Occlusive Disease. Computer Aided Surgery. 2005; 10(4):257–277. [PubMed: 16393794]
5. Cacho F, Doblare M, Holzapfel GA. A Procedure to Simulate Coronary Artery Bypass Graft Surgery. Medical and Biological Engineering and Computing. 2007; 45(9):819–827. [PubMed: 17671805]
6. Pekkan K, De Zelicourt D, Ge L, Sotiropoulos F, Frakes D, Fogel MA, Yoganathan AP. Physics-Driven CFD Modeling of Complex Anatomical Cardiovascular Flows-a TCPC Case Study. Annals of Biomedical Engineering. 2005; 33(3):284–300. [PubMed: 15868719]
7. Marsden AL, Bernstein AJ, Reddy VM, Shadden SC, Spilker RL, Chan FP, Taylor CA, Feinstein JA. Evaluation of a Novel Y-Shaped Extracardiac Fontan Baffle Using Computational Fluid Dynamics. Journal of Thoracic and Cardiovascular Surgery. 2009; 137(2):394–403. [PubMed: 19185159]
8. Ladisa JF, Guler I, Olson LE, Hettrick DA, Kersten JR, Warltier DC, Pagel PS. Three-Dimensional Computational Fluid Dynamics Modeling of Alterations in Coronary Wall Shear Stress Produced by Stent Implantation. Annals of Biomedical Engineering. 2003; 31(8):972–980. [PubMed: 12918912]
9. Ho H, Mithraratne K, Schmid H, Sands G, Hunter P. Computer Simulation of Vertebral Artery Occlusion in Endovascular Procedures. International Journal of Computer Assisted Radiology and Surgery. 2010; 5(1):29–37. [PubMed: 20033514]
10. Stuhne GR, Steinman DA. Finite-Element Modeling of the Hemodynamics of Stented Aneurysms. Journal of Biomechanical Engineering. 2004; 126(3):382–387. [PubMed: 15341176]
11. Cebral JR, Lohner R. Efficient Simulation of Blood Flow Past Complex Endovascular Devices Using an Adaptive Embedding Technique. IEEE Transactions on Medical Imaging. 2005; 24(4): 468–476. [PubMed: 15822805]
12. Li Z, Kleinstreuer C, Farber M. Computational Analysis of Biomechanical Contributors to Possible Endovascular Graft Failure. Biomechanics and Modeling in Mechanobiology. 2005; 4(4):221–234. [PubMed: 16270200]

13. Fung GS, Lam SK, Cheng SW, Chow KW. On Stent-Graft Models in Thoracic Aortic Endovascular Repair: A Computational Investigation of the Hemodynamic Factors. Computers in Biology and Medicine. 2008; 38(4):484–489. [PubMed: 18342843]
14. Figueroa CA, Taylor CA, Chiou AJ, Yeh V, Zarins CK. Magnitude and Direction of Pulsatile Displacement Forces Acting on Thoracic Aortic Endografts. Journal of Endovascular Therapy. 2009; 16(3):350–358. [PubMed: 19642798]
15. Howell BA, Kim T, Cheer A, Dwyer H, Saloner D, Chuter TaM. Computational Fluid Dynamics within Bifurcated Abdominal Aortic Stent-Grafts. Journal of Endovascular Therapy. 2007; 14(2): 138–143. [PubMed: 17484528]
16. Molony DS, Callanan A, Morris LG, Doyle BJ, Walsh MT, Mcgloughlin TM. Geometrical Enhancements for Abdominal Aortic Stent-Grafts. Journal of Endovascular Therapy. 2008; 15(5): 518–529. [PubMed: 18840041]
17. Taylor CA, Draney MT, Ku JP, Parker D, Steele BN, Wang K, Zarins CK. Predictive Medicine: Computational Techniques in Therapeutic Decision-Making. Computer Aided Surgery. 1999; 4(5):231–247. [PubMed: 10581521]
18. Wang KC, Dutton RW, Taylor CA. Improving Geometric Model Construction for Blood Flow Modeling. IEEE Engineering in Medicine and Biology Magazine. 1999; 18(6):33–39. [PubMed: 10576070]
19. Steinman DA. Image-Based Computational Fluid Dynamics Modeling in Realistic Arterial Geometries. Annals of Biomedical Engineering. 2002; 30(4):483–497. [PubMed: 12086000]
20. Antiga L, Ene-Iordache B, Remuzzi A. Computational Geometry for Patient-Specific Reconstruction and Meshing of Blood Vessels from MR and CT Angiography. IEEE Transactions on Medical Imaging. 2003; 22(5):674–684. [PubMed: 12846436]
21. Shim MB, Gunay M, Shimada K. Three-Dimensional Shape Reconstruction of Abdominal Aortic Aneurysm. Computer-Aided Design. 2009; 41(8):555–565.
22. Zhang YJ, Bazilevs Y, Goswami S, Bajaj CL, Hughes TJR. Patient-Specific Vascular Nurbs Modeling for Isogeometric Analysis of Blood Flow. Computer Methods in Applied Mechanics and Engineering. 2007; 196(29–30):2943–2959. [PubMed: 20300489]
23. Bekkers EJ, Taylor CA. Multiscale Vascular Surface Model Generation from Medical Imaging Data Using Hierarchical Features. IEEE Transactions on Medical Imaging. 2008; 27(3):331–341. [PubMed: 18334429]
24. Martin D, Boyle FJ. Computational Structural Modelling of Coronary Stent Deployment: A Review. Computer Methods in Biomechanics and Biomedical Engineering. 2010
25. Yim PJ, Cebral JJ, Mullick R, Marcos HB, Choyke PL. Vessel Surface Reconstruction with a Tubular Deformable Model. IEEE Transactions on Medical Imaging. 2001; 20(12):1411–1421. [PubMed: 11811840]
26. Yim PJ, Vasbinder GBC, Ho VB, Choyke PL. Isosurfaces as Deformable Models for Magnetic Resonance Angiography. IEEE Transactions on Medical Imaging. 2003; 22(7):875–881. [PubMed: 12906241]
27. Cebral JR, Lohner R, Choyke PL, Yim PJ. Merging of Intersecting Triangulations for Finite Element Modeling. Journal of Biomechanics. 2001; 34(6):815–819. [PubMed: 11470121]
28. Sethian, JA. Level Set Methods and Fast Marching Methods. Cambridge, U.K.: Cambridge University Press; 1999.
29. Cebral JR, Hernández M, Frangi AF. Computational Analysis of Blood Flow Dynamics in Cerebral Aneurysms from CTA and 3D Rotational Angiography Image Data. In Proceedings of the International Congress on Computational Bioengineering. 2003:191–198.
30. Caselles V, Kimmel R, Sapiro G. Geodesic Active Contours. International Journal of Computer Vision. 1997; 22(1):61–79.
31. Antiga L, Piccinelli M, Botti L, Ene-Iordache B, Remuzzi A, Steinman D. An Image-Based Modeling Framework for Patient-Specific Computational Hemodynamics. Medical and Biological Engineering and Computing. 2008; 46(11):1097–1112. [PubMed: 19002516]
32. Lorensen, WE.; Cline, HE. Marching Cubes: A High Resolution 3d Surface Construction Algorithm; Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques; 1987. p. 163-169.

33. Chan SL, Purisima EO. A New Tetrahedral Tesselation Scheme for Isosurface Generation. Computers & Graphics. 1998; 22(1):83–90.
34. Boissonnat JD, Oudot S. Provably Good Sampling and Meshing of Surfaces. Graphical Models. 2005; 67(5):405–451.
35. Ford MD, Hoi Y, Piccinelli M, Antiga L, Steinman DA. An Objective Approach to Digital Removal of Saccular Aneurysms: Technique and Applications. British Journal of Radiology. 2009; 82:S55–S61. [PubMed: 20348537]
36. Appanaboyina S, Mut F, Lohner R, Putman CA, Cebral JR. Computational Fluid Dynamics of Stented Intracranial Aneurysms Using Adaptive Embedded Unstructured Grids. International Journal for Numerical Methods in Fluids. 2008; 57(5):475–493.
37. Xiong G, Figueroa CA, Xiao N, Taylor CA. Simulation of Blood Flow in Deformable Arteries Using Subject-Specific Geometry and Spatially-Varying Wall Properties. International Journal for Numerical Methods in Biomedical Engineering. 2010
38. Duraiswamy N, Schoephoerster RT, Moreno MR, Moore JE. Stented Artery Flow Patterns and Their Effects on the Artery Wall. Annual Review of Fluid Mechanics. 2007; 39:357–382.
39. Hager WW, Zhang HC. A New Conjugate Gradient Method with Guaranteed Descent and an Efficient Line Search. Siam Journal on Optimization. 2005; 16(1):170–192.
40. Yap CK. Towards Exact Geometric Computation. Computational Geometry. 1997; 7(1–2):3–23.
41. Rozza G. On Optimization, Control and Shape Design of an Arterial Bypass. International Journal for Numerical Methods in Fluids. 2005; 47(10–11):1411–1419.
42. Hachenberger P, Kettner L, Mehlhorn K. Boolean Operations on 3d Selective Nef Complexes: Data Structure, Algorithms, Optimized Implementation and Experiments. Computational Geometry. 2007; 38(1–2):64–99.
43. Garland, M.; Heckbert, PS. Surface Simplification Using Quadric Error Metrics; Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques; 1997. p. 209-216.
44. Kobbelt, L. Square Root 3-Subdivision; Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques; 2000. p. 103-112.
45. Desbrun, M.; Meyer, M.; Schröder, P.; Barr, AH. Implicit Fairing of Irregular Meshes Using Diffusion and Curvature Flow; Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques; 1999. p. 317-324.
46. Pijls NHJ, Debruyne B, Peels K, Vandervoort PH, Bonnier HJRM, Bartunek J, Koolen JJ. Measurement of Fractional Flow Reserve to Assess the Functional Severity of Coronary-Artery Stenoses. New England Journal of Medicine. 1996; 334(26):1703–1708. [PubMed: 8637515]
47. Williams AR, Koo BK, Gundert TJ, Fitzgerald PJ, Ladisa JF Jr. Local Hemodynamic Changes Caused by Main Branch Stent Implantation and Subsequent Virtual Side Branch Balloon Angioplasty in a Representative Coronary Bifurcation. Journal of Applied Physiology. 2010; 109(2):532–540. [PubMed: 20507966]

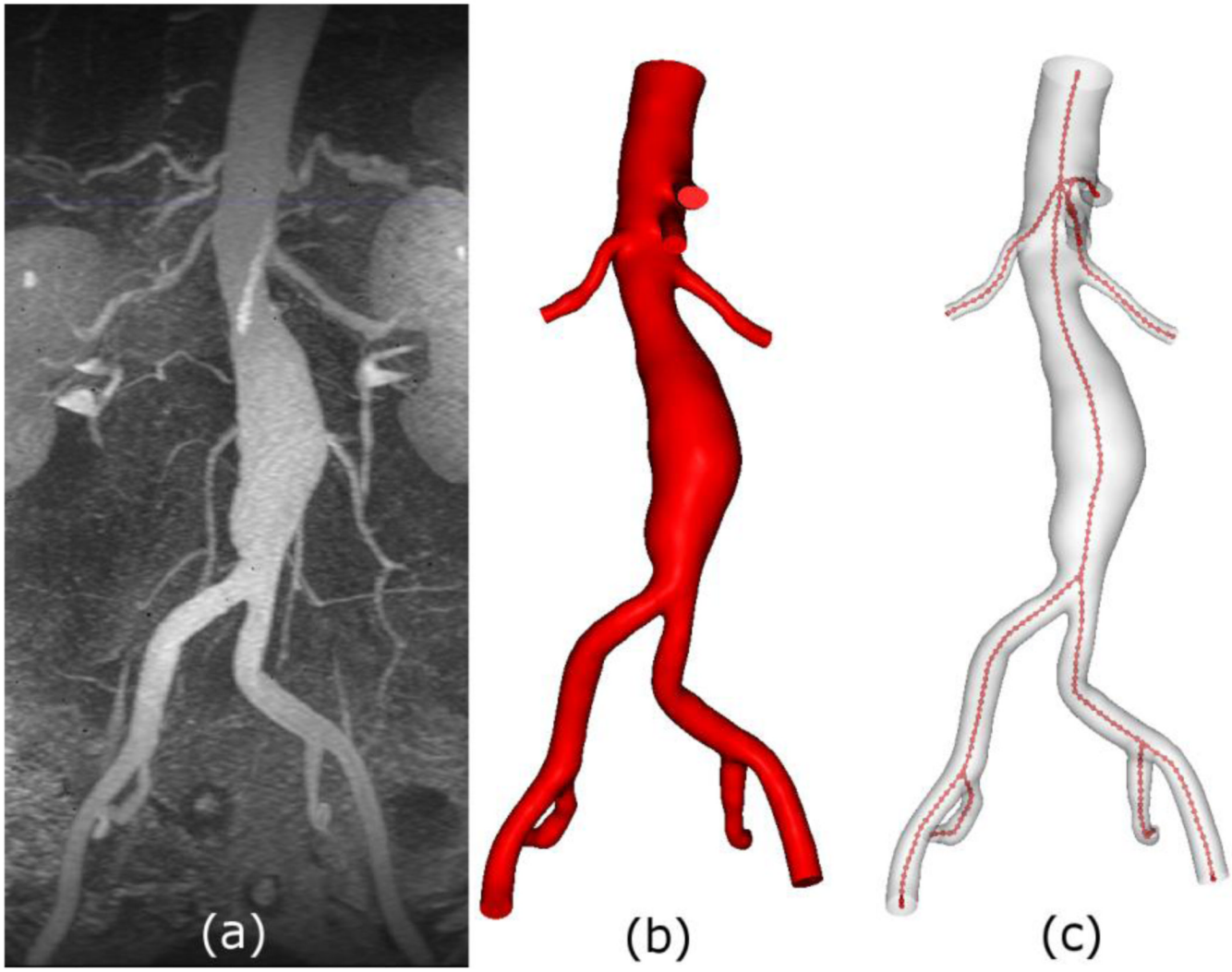

Figure 1.
An example of vascular model construction and centerline extraction. (a) a 3D image of abdominal aorta; (b) the geometric model; (c) the centerline.

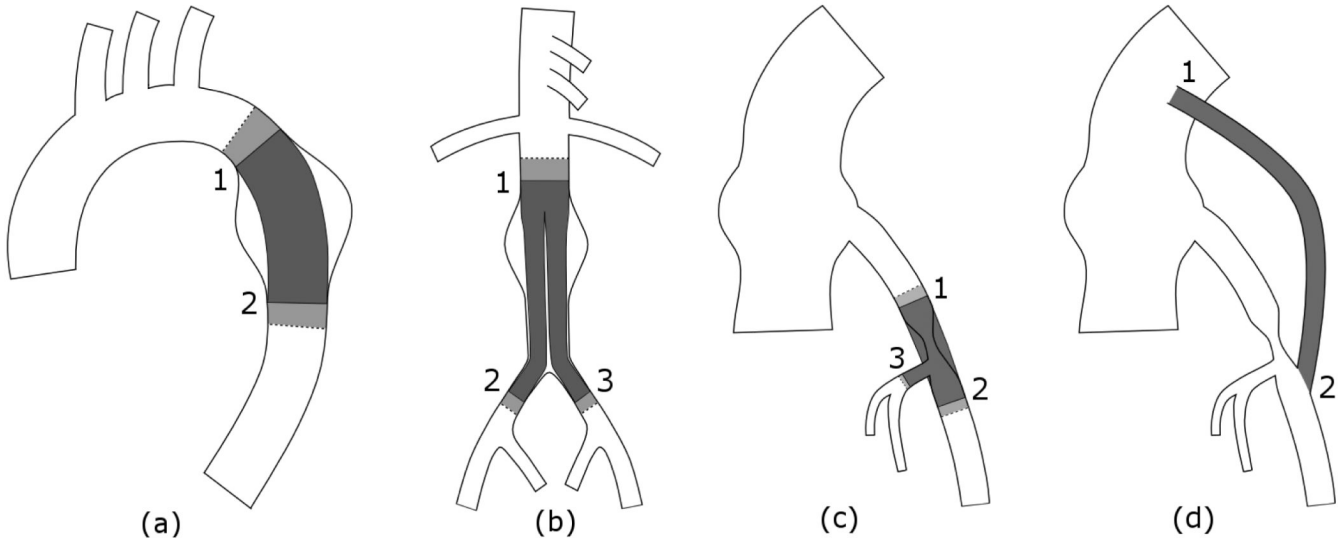

Figure 2.

Illustration of stent grafts for thoracic aortic aneurysm (a) and abdominal aortic aneurysm (b), as well as stents (c) and bypass grafts (d) for coronary stenosis. The geometric changes led by the device deployment are marked as dark, whereas the transitions between original subject-specific and device geometry are in gray.

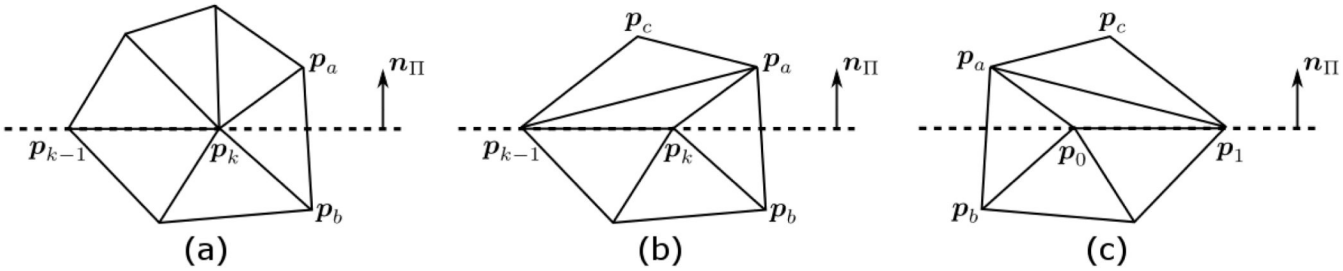

Figure 3.

The splitting algorithm. (a) A normal case of moving nodes in the vicinity to the splitting plane (depicted as dashed lines); (b) and (c) two extreme cases involving degenerate triangles.

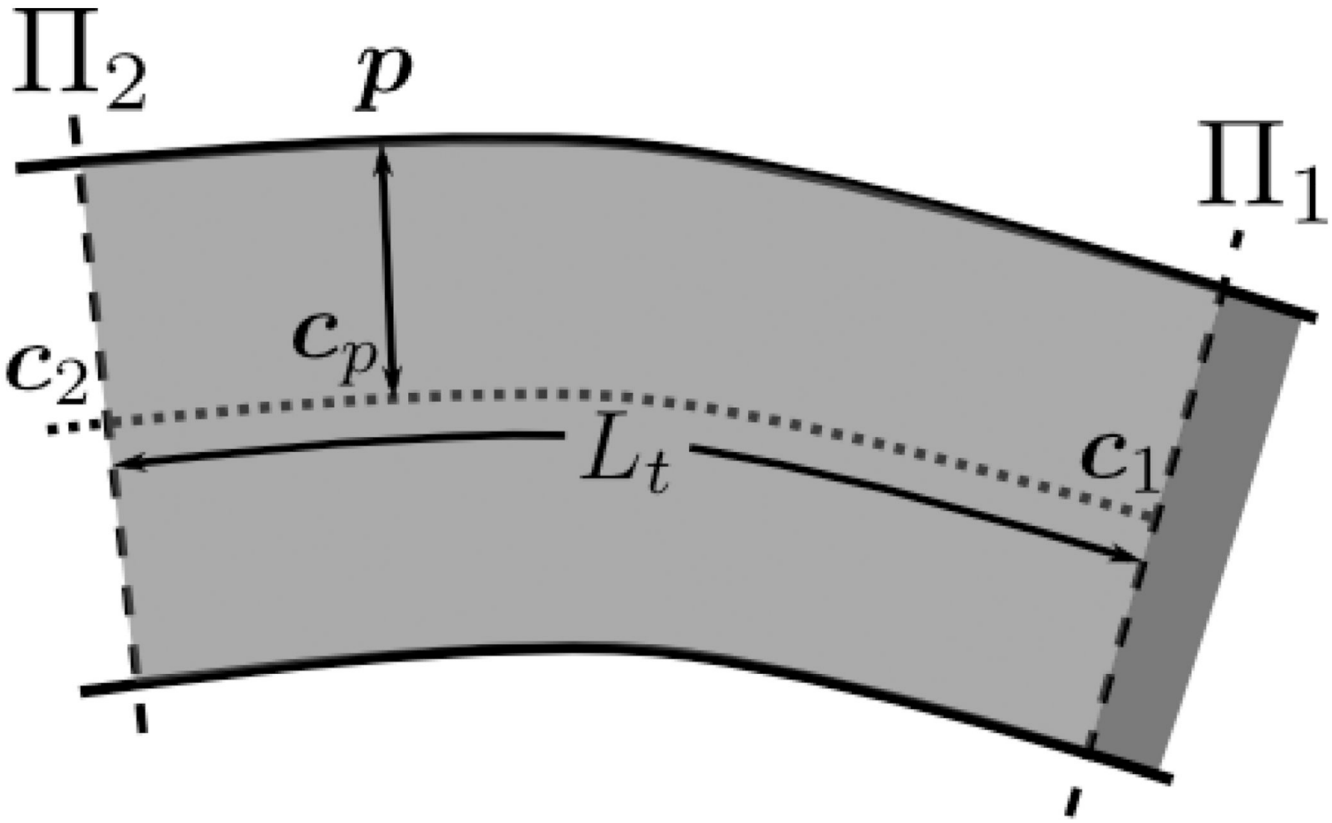

Figure 4.
The morphing algorithm. Gray and dark colors mark the transition region and the device geometry, respectively. Dotted and dashed lines are the centerline and planes, respectively.

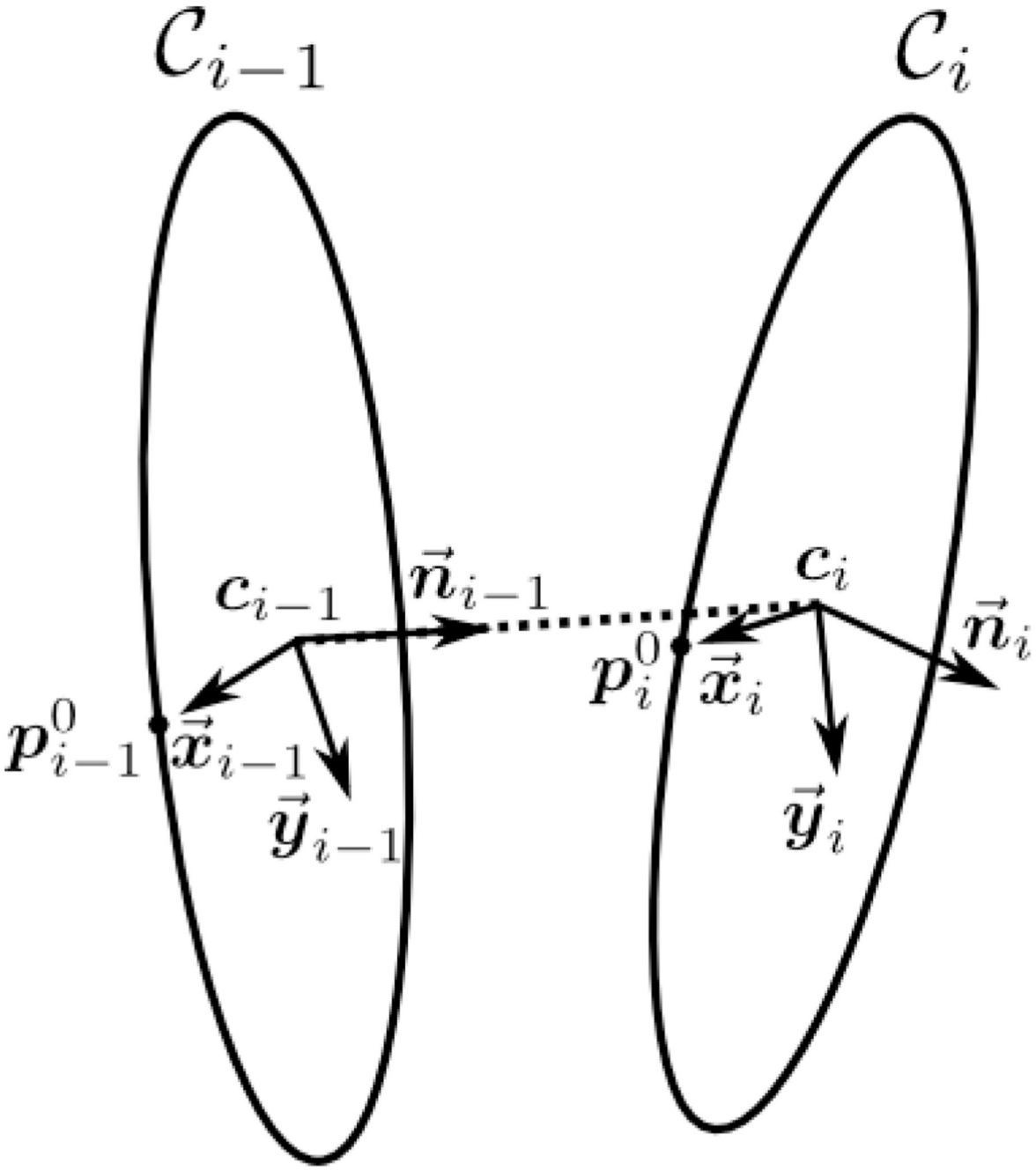

Figure 5.
The bridging algorithm. The dotting line is the centerline.

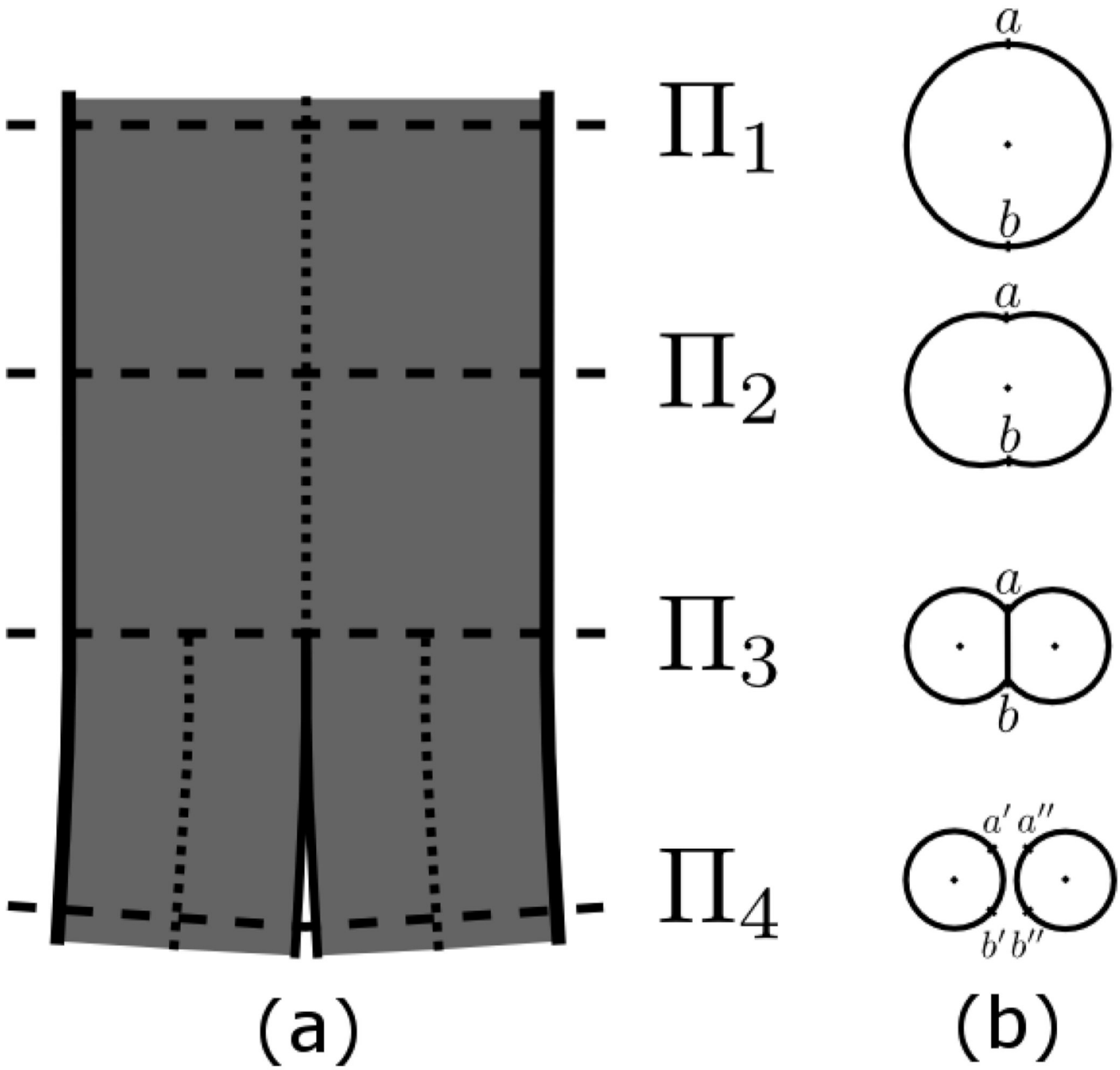

Figure 6.
The bisecting algorithm. (a) The parent vessel bisects into two branches at the bifurcation. Dotted and dashed lines are the centerline and planes, respectively. (b) Four contours at different planes are shown.

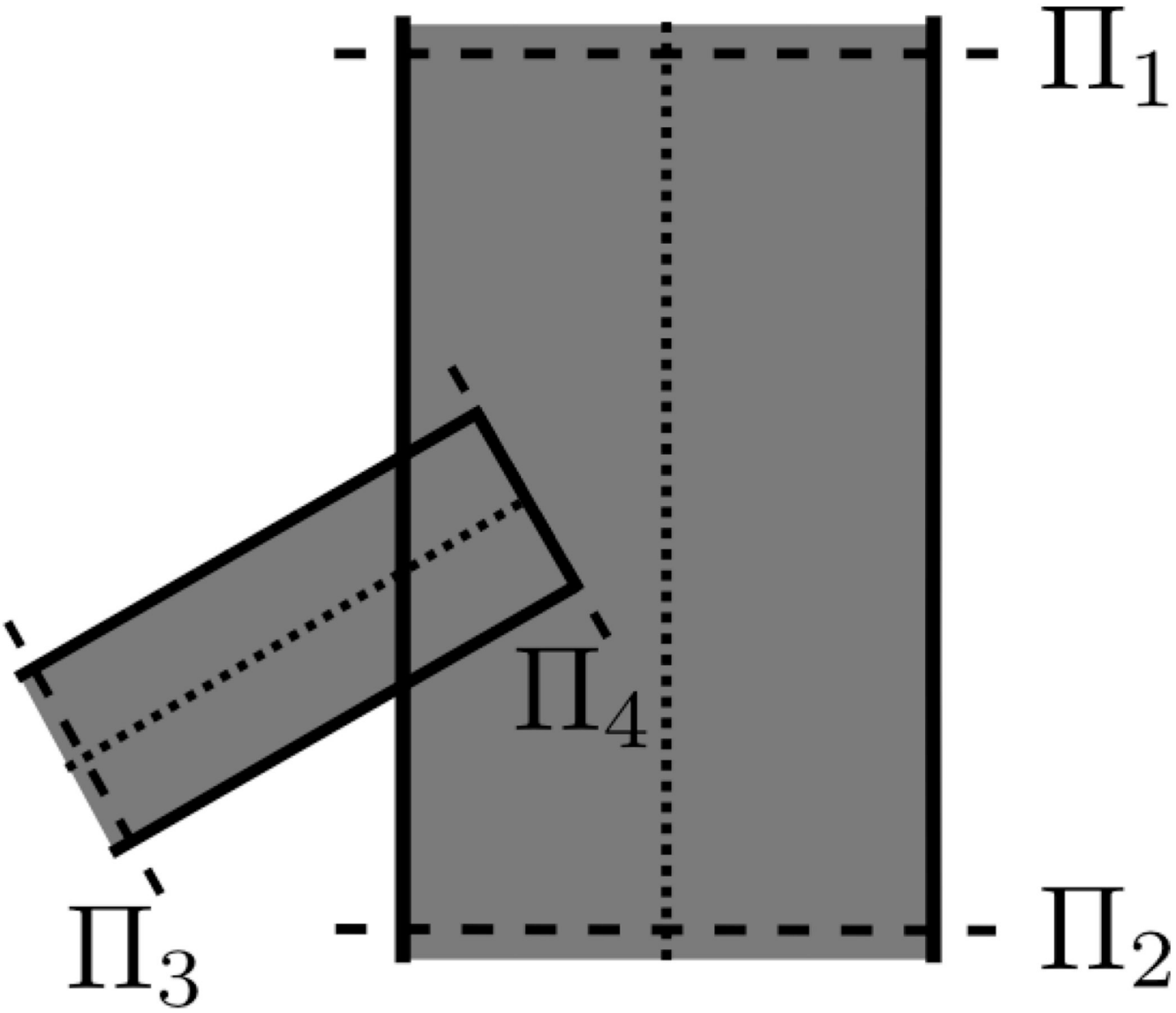

Figure 7.
The joining algorithm. Dotted and dashed lines are the centerline and planes, respectively.

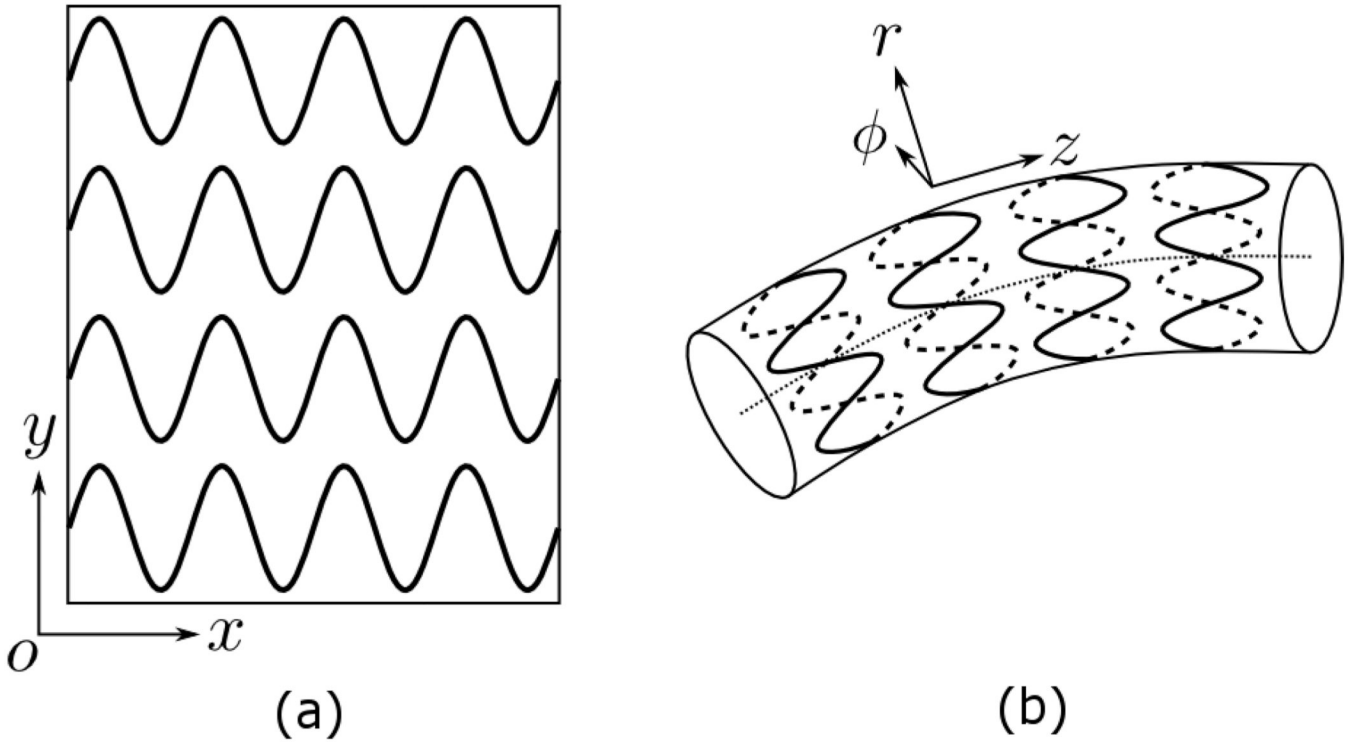

Figure 8.
Strut pattern mapping between 2D rectangular coordinates (a) and 3D cylindrical coordinates (b).

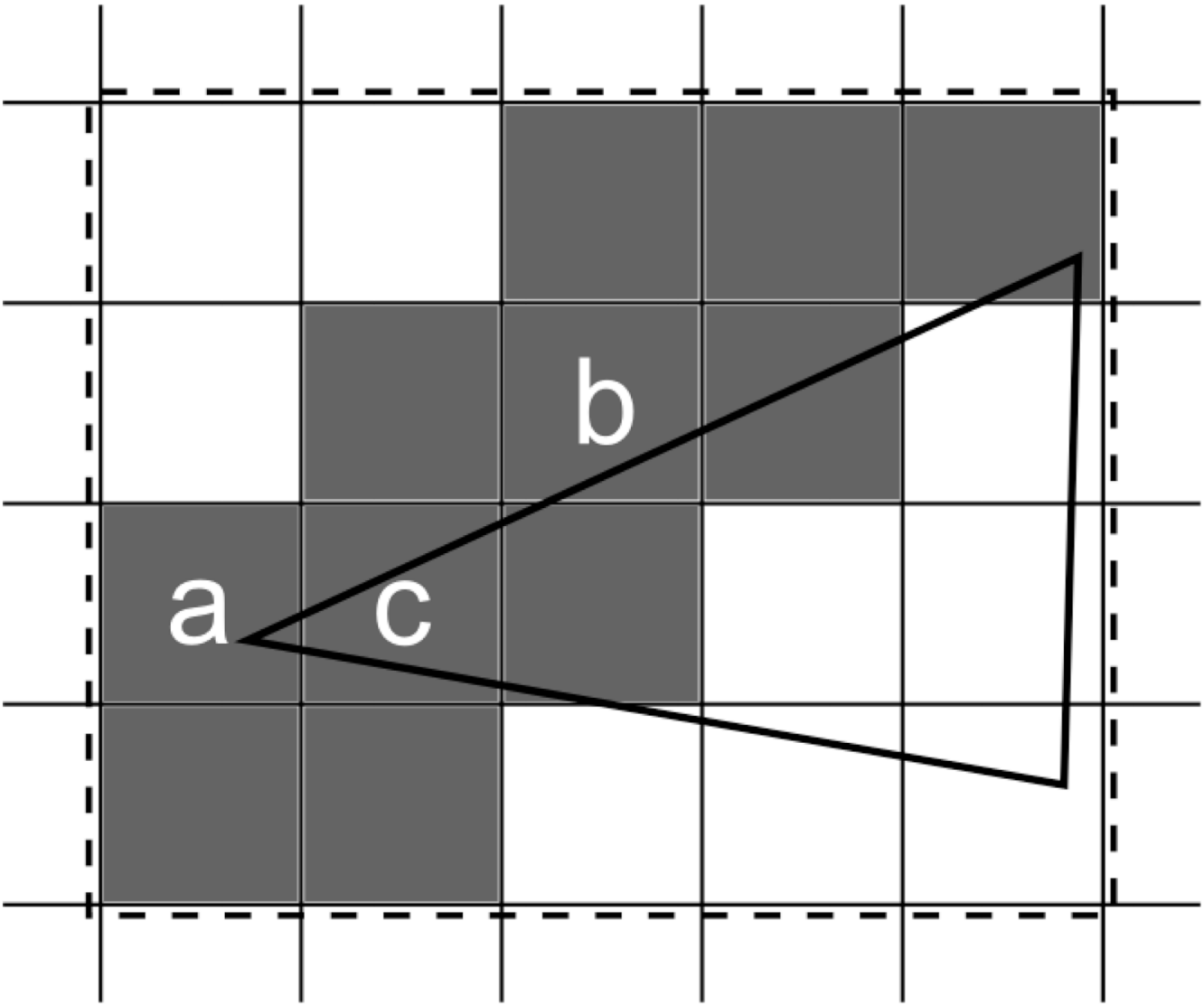

Figure 9.
Intersection test of a triangle and strut pixels (dark color) in 2D image plane.

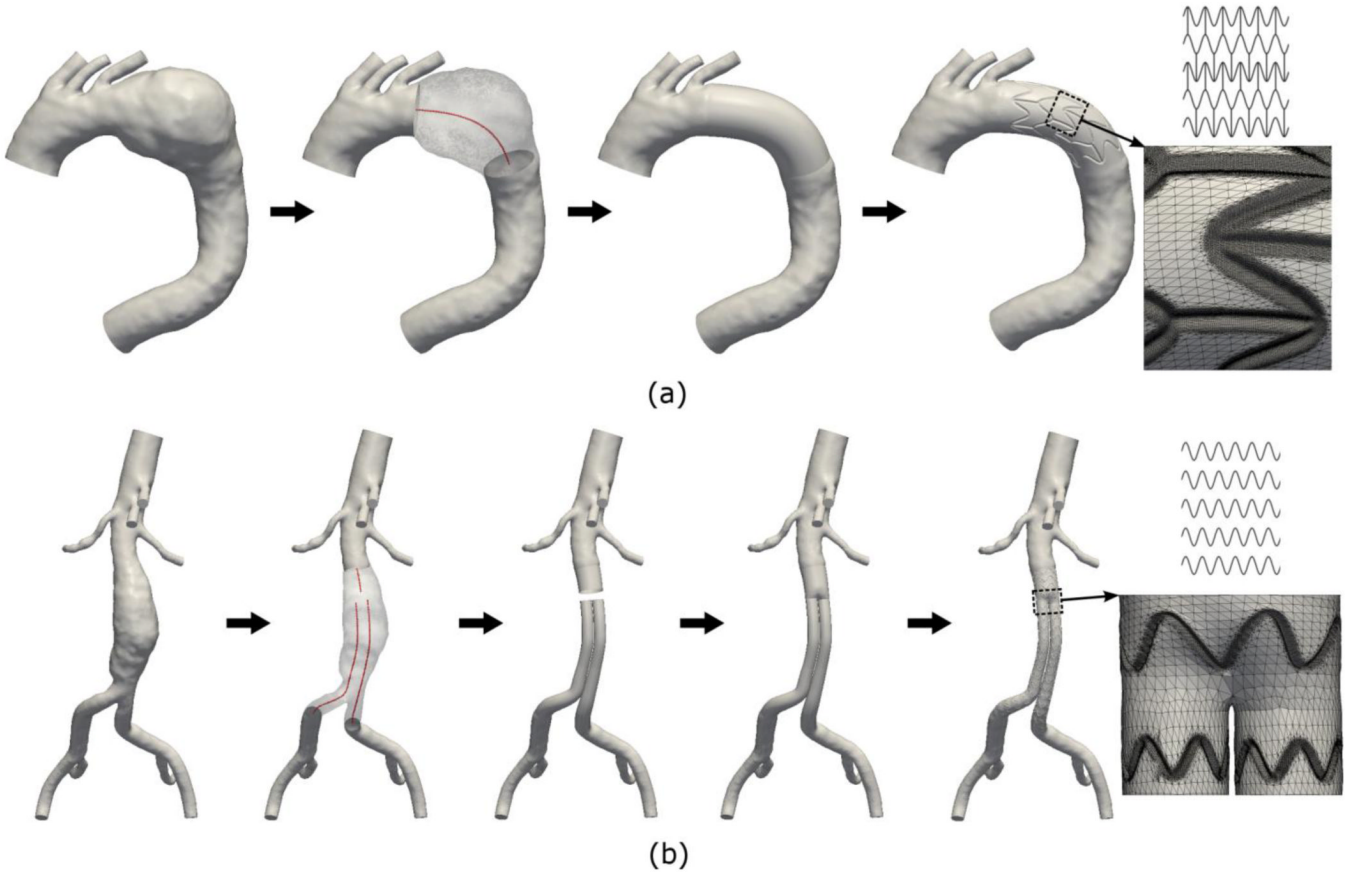

Figure 10.
Virtual stent grafting for thoracic (a) and abdominal (b) aortic aneurysms.

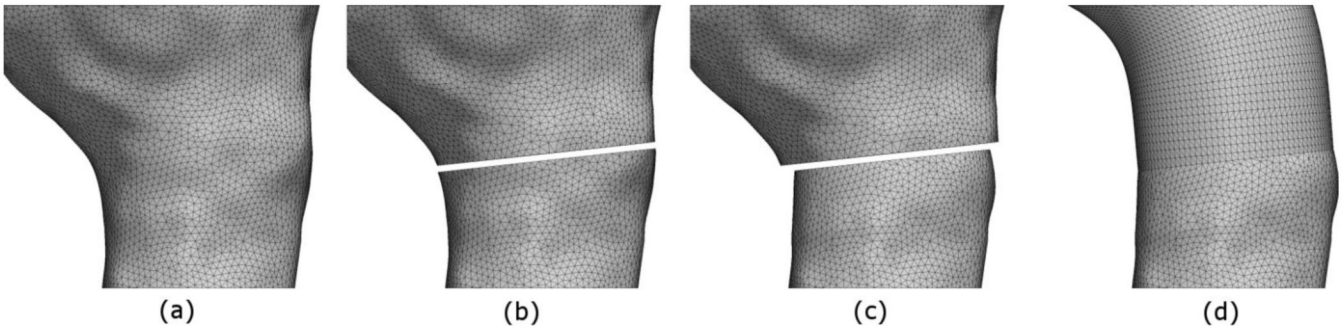

Figure 11.
The meshes (closer to the distal end) in the process of stent grafting for thoracic aortic aneurysm in Fig. 10(a). (a) the pre-operative mesh; (b) after splitting; (c) after morphing; (d) after bridging.

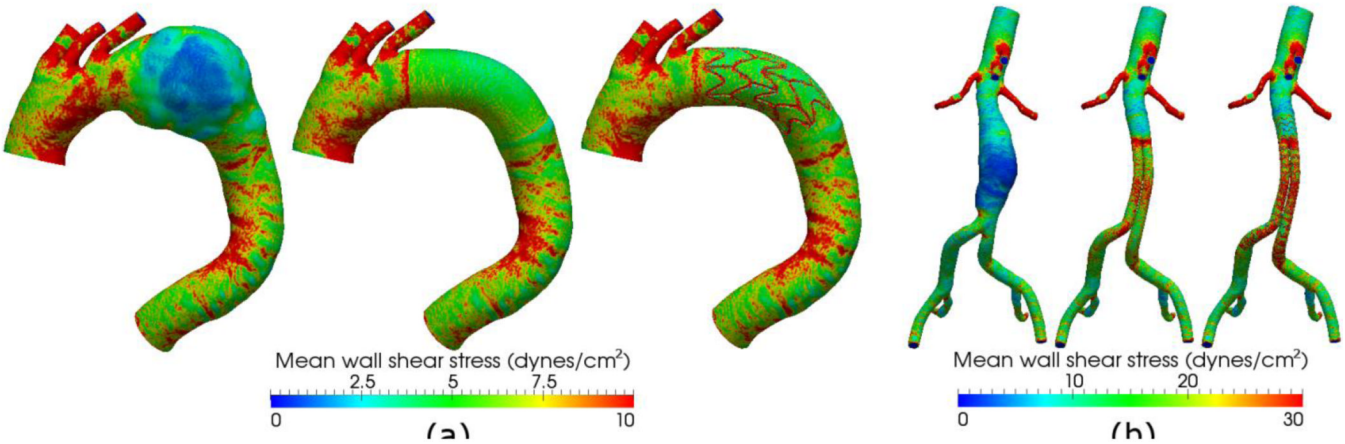

Figure 12.
Mean wall shear stress for the models of thoracic aorta (a) and abdominal aorta (b).

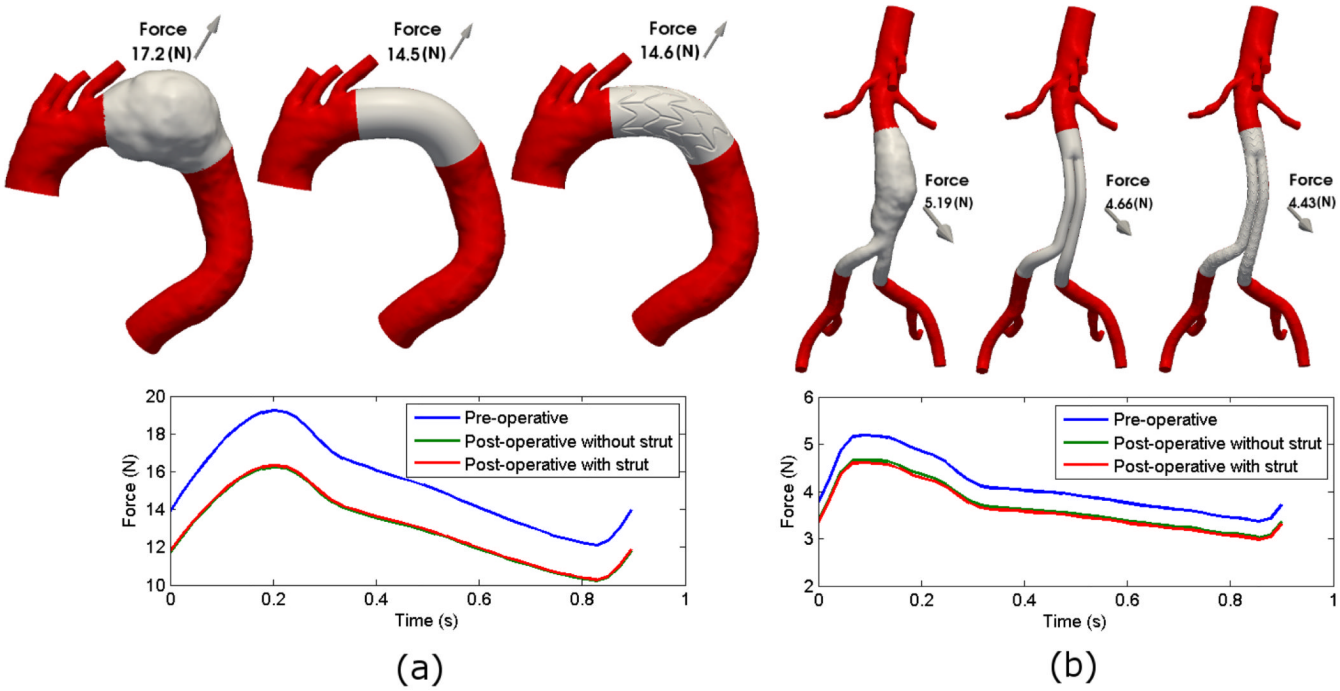

Figure 13.
Net forces for the models of thoracic aorta (a) and abdominal aorta (b).

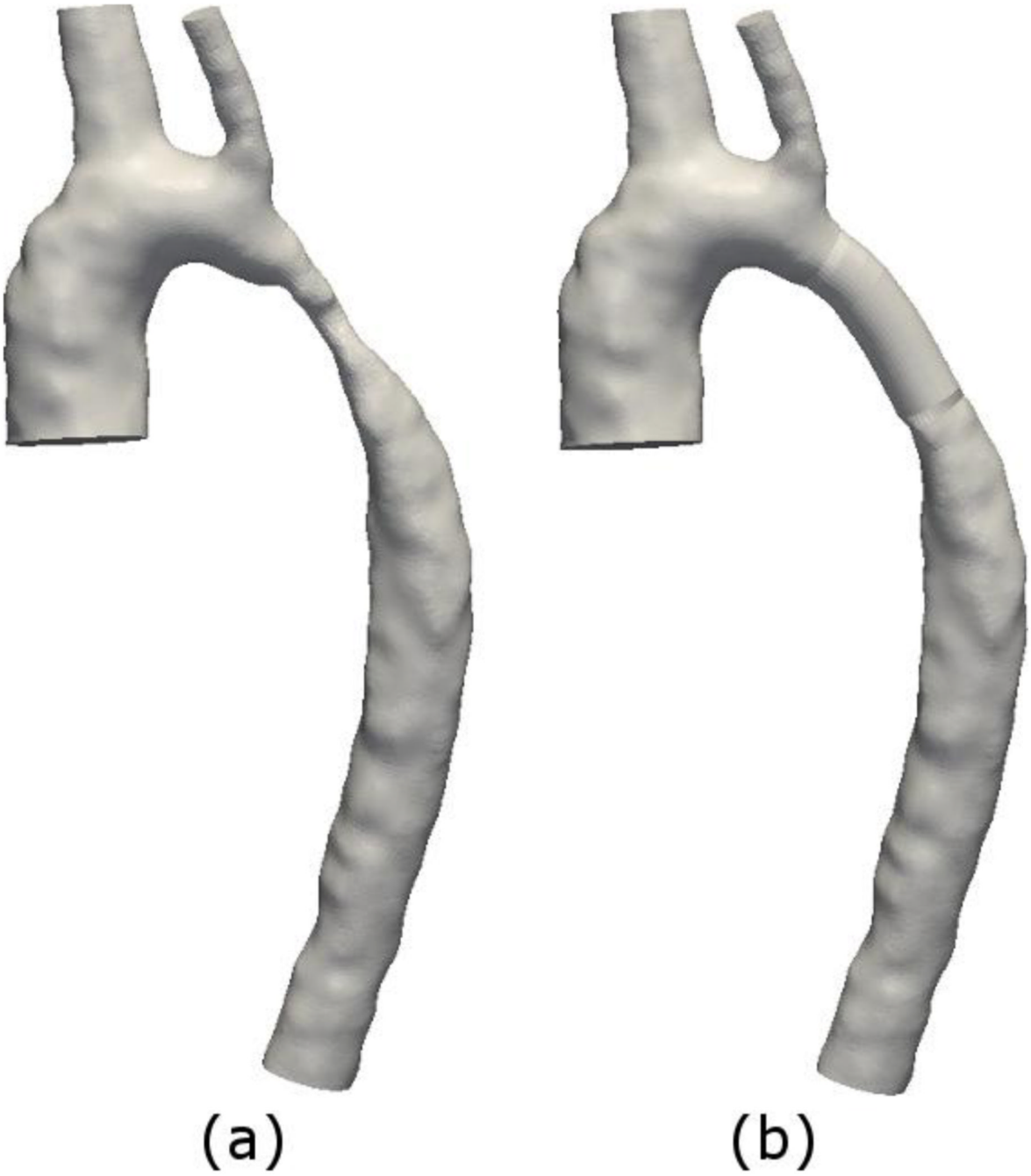

Figure 14.
Virtual stenting for aortic coarctation. (a) the pre-operative model; (b) the post-operative model.

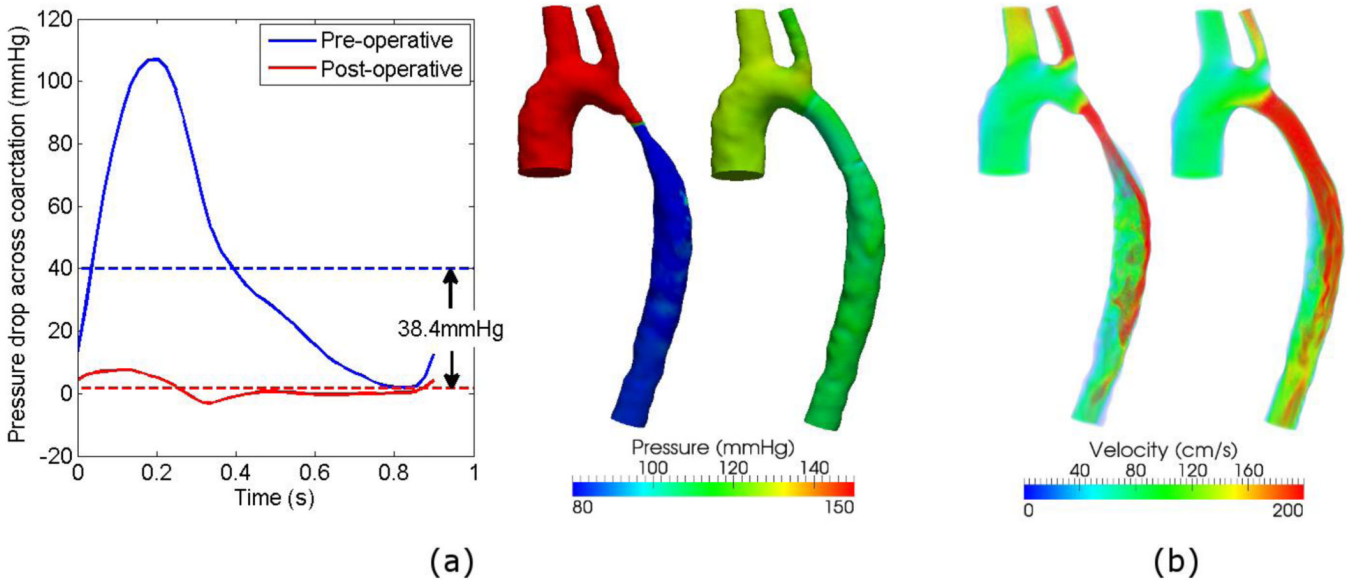

Figure 15.
The pressure (a) and velocity fields (b) before and after stenting of aortic coarctation.

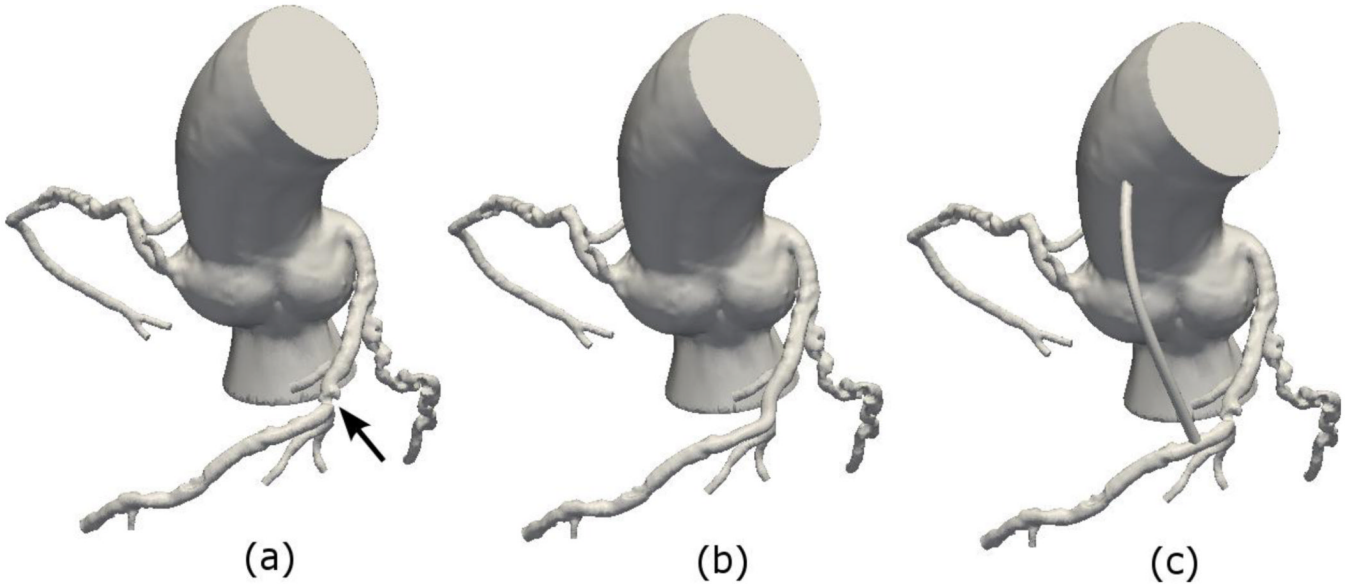

Figure 16.
Virtual stenting for coronary artery stenosis (arrow pointed). (a) pre-operative; (b) post-operative with stenting; (c) post-operative with bypass grafting.

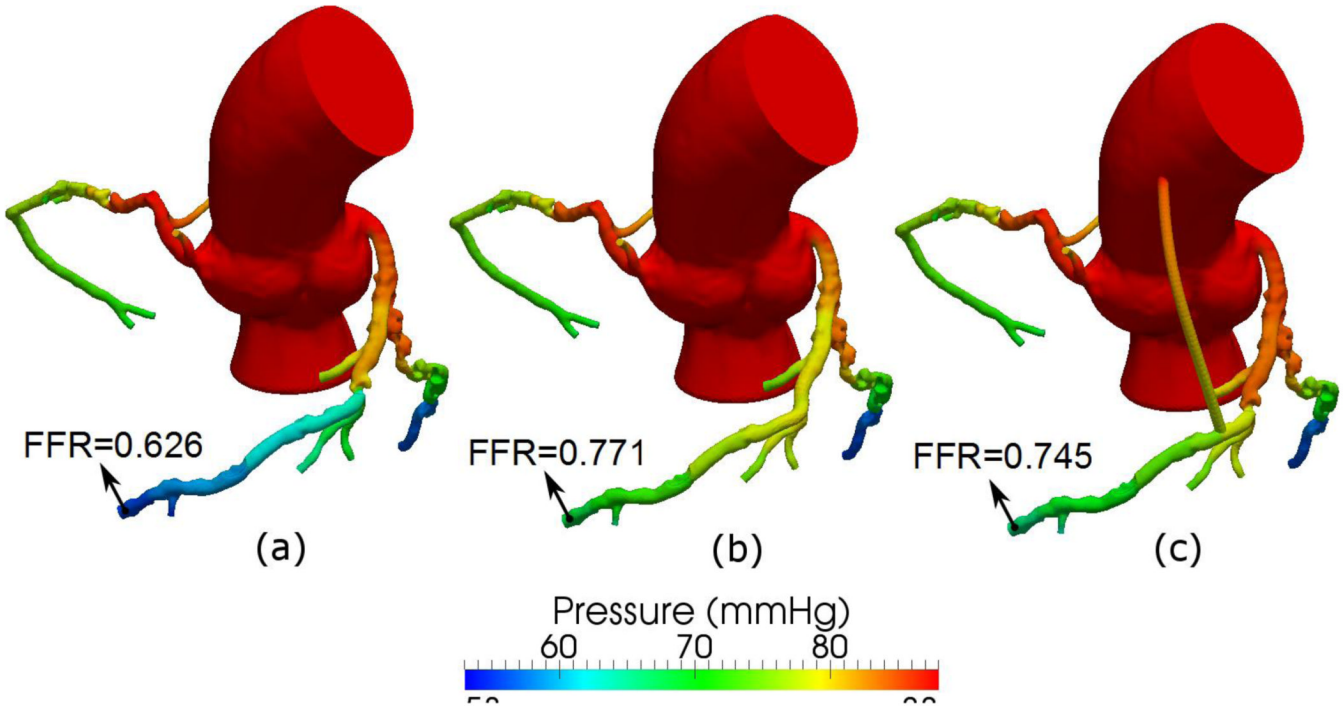

Figure 17.
The pressure field and FFR values for coronary artery stenosis. (a) pre-operative; (b) post-operative with stenting; (c) post-operative with bypass grafting.

Table 1

The mesh statistics of all models

Models		Vertices	Triangles
Thoracic aortic aneurysm	Pre-operative	30,860	61,716
	Post-operative (without strut)	25,811	51,618
	Post-operative (with strut)	180,469	360,934
Abdominal aortic aneurysm	Pre-operative	24,728	49,452
	Post-operative (without strut)	17,310	34,616
	Post-operative (with strut)	277,716	555,428
Aortic coarctation	Pre-operative	17,235	34,466
	Post-operative	16,715	33,404
Coronary artery stenosis	Pre-operative	108,293	216,586
	Post-operative (stent)	107,005	214,010
	Post-operative (bypass graft)	109,963	219,930