ICES REPORT 17-24

September 2017

Patient-Specific Vascular Modeling: Template-based Isogeometric Framework and the Case for CAD

by

Benjamin Urick, Travis M. Sanders, Shaolie S. Hossain, Yongjie Zhang, Thomas J. R. Hughes

The Institute for Computational Engineering and Sciences
The University of Texas at Austin
Austin, Texas 78712

Reference: Benjamin Urick, Travis M. Sanders, Shaolie S. Hossain, Yongjie Zhang, Thomas J. R. Hughes, "Patient-Specific Vascular Modeling: Template-based Isogeometric Framework and the Case for CAD ," ICES REPORT 17-24, The Institute for Computational Engineering and Sciences, The University of Texas at Austin, September 2017.

Patient-Specific Vascular Modeling: Template-based Isogeometric Framework and the Case for CAD

Benjamin Urick[a], Travis M. Sanders[a], Shaolie S. Hossain[a,b,*], Yongjie Zhang[c], Thomas J.R. Hughes[a]

[a]Institute for Computational Engineering and Sciences
The University of Texas at Austin, Austin, Texas, USA

[b]Department of Molecular Cardiology
Texas Heart Institute, Houston, Texas, USA

[c]Department of Mechanical Engineering
Carnegie Mellon University, Pittsburgh, Pennsylvania, USA

Abstract: Patient-specific vascular modeling involves three main steps: image processing, analysis suitable model generation, and computational analysis. Analysis suitable model generation currently suffers from difficulties and complications, which often necessitate manual intervention and crude approximations. Because the modeling pipeline spans multiple disciplines, the benefits of integrating a computer-aided design (CAD) component for the geometric modeling tasks has been largely overlooked. In this paper, we present a CAD-integrated template-based modeling framework that streamlines the construction of solid NURBS (non-uniform rational B-spline) vascular models for performing isogeometric finite element analysis. Examples of arterial models for mouse and human circles of Willis and a porcine coronary tree are presented.

1 Introduction

Patient-specific modeling is a relatively new paradigm in medical planning that utilizes computational tools on anatomical and physiological data to individualize patient care. It has the potential to optimize surgical procedures and to improve diagnosis and treatment of a number of common illnesses. To these ends, image-based patient-specific modeling techniques have been used to study various organs and tissue dynamics including the heart, the brain, bones, teeth, kidneys, tumors, lungs and the cardiovascular system [1]. Patient-specific vascular modeling (PSVM)

*Corresponding author email address: shossain@ices.utexas.edu

in particular has enjoyed a great deal of attention from researchers since the pioneering work of Taylor *et al.* in [2,3], and remains an active field of study [4–6]. PSVM makes possible the detection of coronary artery disease [7], treatment planning such as stent placement and optimizing bypass graft location [8,9], nanoparticulate drug delivery [10–13] and abdominal aortic aneurysm and cerebral aneurysm assessment [14,15].

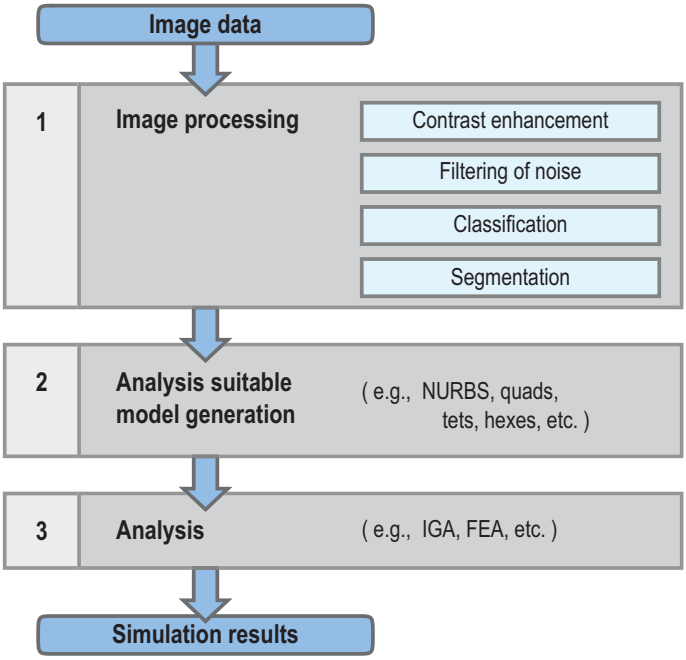

Fig. 1 Flowchart depicting the main steps in a typical vascular patient-specific modeling pipeline: 1) Image processing, 2) Analysis suitable model generation, and 3) Analysis.

PSVM involves three fundamental steps: (1) image processing, (2) analysis suitable model generation, and (3) analysis (Figure 1). First, the scanned images are segmented, or classified into meaningful regions to extract the geometry of the volumes of interest. Next, an analysis suitable solid geometric model is generated from the segmented images. Typically, solid models of blood vessels are constructed by (a) extracting a set of points (a contour) approximating the inner boundary (the lumen-wall interface) of a vessel on an offset stack of segmented two-dimensional (2D) image slices, (b) interpolating the contour with a curve, (c) lofting a surface through the interpolated curves, (d) joining the surfaces together to form a network, and (e) meshing, or infilling, the bounded surfaces to define the volume of the vascular network [16]. In the past several years, there has been an increased tendency to utilize three-dimensional (3D) segmentation methods to extract triangulated surface meshes from which volumetric meshes are generated [17–19]. Finally, a finite element, finite difference, or finite volume method is used to solve the incompressible Navier-Stokes equations on the generated mesh to simulate the velocity and pressure profiles of the blood flow in the vasculature [20–22].

Among the existing patient-specific analysis techniques, finite-element-based isogeometric analysis (IGA) [23] has gained significant traction in recent years [24–29]. The development of IGA was motivated by a desire to seamlessly integrate engineering design and computational analysis. Prior to the conception of IGA,

performing finite element analysis (FEA) on a computer-aided designed (CAD) object required the creation of an analysis suitable mesh that only approximated the original CAD object. Not only does this reduce the accuracy of the final solution, but for large objects, creating the analysis suitable mesh can take a design team a few weeks or more to complete, consuming up to 80% of the total analysis time [30]. To circumvent the mesh generation process and its unnecessary geometry approximations altogether, IGA uses the same basis functions for approximating the solution space as the ones used in representing the geometry in CAD software. In this isoparametric framework, the analysis suitable model becomes equivalent to the CAD model. As a consequence, IGA has the distinct advantage of performing analysis on exact geometries, even for the coarsest mesh. In addition, the IGA approach can take advantage of extensive CAD software functionality in producing geometric models and corresponding geometric information. Furthermore, once a geometric model is created in CAD, a traditional FE mesh can be constructed if desired, making a CAD-integrated approach suitable for both traditional FEA as well as IGA.

The engineering design standard for modeling curves and surfaces in commercial CAD software, including AutoCAD®[31], SolidWorks®[32], Rhinoceros® (Rhino)[33], and CATIA[34], are Non-Uniform Rational B-Splines, or NURBS. There are many benefits in using NURBS for design as well as for analysis. For design, NURBS are not only capable of representing conic sections exactly but they are very convenient for free-form surface modeling [35]. For analysis, NURBS have excellent approximation properties important for computing accurate solutions. Particularly for vascular modeling, a NURBS-based parameterization of the fluid domain has many advantages. For example, a smooth geometric representation is imperative for obtaining accurate solutions, especially for critical near-wall quantities such as wall shear stress and wall shear stress gradient [36]. A cylindrical parameterization of the vasculature is more natural due to the inherent tubular structure of arteries. This allows for simple local refinement schemes of the mesh near the artery walls, which is necessary for accurately capturing flow features in the boundary layer [25]. NURBS-based parameterizations also make possible geometric sensitivity analysis for uncertainty quantification [37]. Additionally, the recent implementation [38,39] to convert NURBS data to a format readable by ABAQUS®, a popular analysis software, has made the traditional design-through-analysis process even simpler.

Though CAD software is the standard for creating and manipulating geometric models in production engineering design (e.g., automotive, aerospace and commercial products), it also offers many specific advantages for modeling patient-specific vascular geometry. For example, using a CAD graphical user interface (GUI) provides the developer the ability to rotate, pan, and zoom in geometric features while troubleshooting code. This can save developers significant time compared with debugging by the conventional way of interpreting textual outputs. Additionally, developers can interface with CAD software at even lower levels. For example, a developer can build custom applications using the CAD geometric library, or kernel. The geometric operations in the CAD kernel have been extensively optimized to offer the developer very fast routines. Alternatively, one can use external applications with real-time CAD interfaces or even run scripts and macros through the CAD software itself. Finally, a CAD- based modeling approach offers the ability to quickly edit and/or manipulate geometric features. For example, stents, catheters, bypass grafts, and so on can be incorporated into vascular models to inform treatment planning.

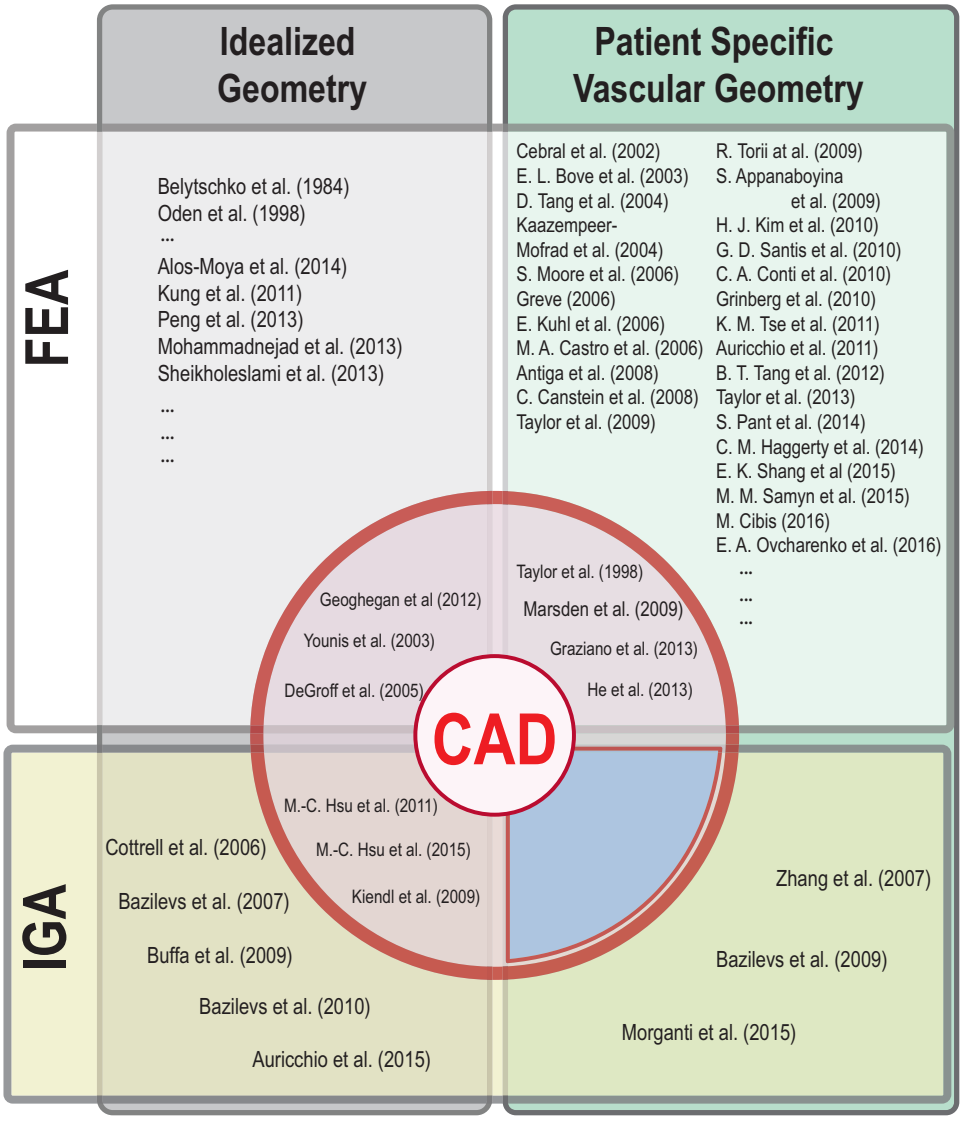

Fig. 2 Categories of modeling pipelines. Left: Idealized geometry pipelines, including design of engineering parts as well as geometric simplifications such as cylindrical tubes used to approximate complex vasculature. Right: Patient-specific vascular modeling pipelines. The figure is also split vertically by analysis approach; finite element analysis (FEA) pipelines (top) and isogeometric analysis (IGA) pipelines (bottom). The center of the figure represents pipelines that utilize CAD software. The domain of CAD integrated patient-specific IGA (blue region) is currently severely underutilized.

Despite these inherent advantages, there are no pipelines that currently exist utilizing CAD to reconstruct patient-specific NURBS-based vascular models for IGA. To illustrate this point, a Venn diagram representing the state-of-the-art in computational modeling pipelines is provided (Figure 2). The diagram is split vertically into two categories based on geometry; idealized geometries (left) and patient-specific vascular geometries (right). Idealized geometries include descriptions explicitly defined and created by a designer, whereas patient-specific vascular geometries originate from medical images, sampled through scanned sources. Horizontally, the diagram is divided by analysis type; pipelines that use the finite element method (top) and those that use IGA (bottom). Finally, each of these four sections is subdivided

into two by the red circle representing utilization of CAD software somewhere in the pipeline (inside the circle) or not (outside the circle).

The vast majority of patient-specific modeling pipelines presented in the literature have adopted the traditional finite element method for their analysis technique (top-right quadrant) [7,40–45], with little or no CAD integration. Exceptions include Taylor *et al*. [3] and Graziano *et al*. [46], who incorporated CAD software for creating 3D geometries before forming a tetrahedral mesh from the geometry and finally applying finite element analysis. Others, including He *et al.* [47] and Marsden et al. [48], have used CAD to edit and manipulate geometries after segmentation and reconstruction. Since the introduction of IGA in 2005 [23], it has been used widely on modeled geometry, or geometry designed by a user (bottom left quadrant) [49,50] including medical devices such as heart valves [51] and vascular stents [28]. Though IGA was invented to fully integrate the design process with analysis, many researchers have not incorporated CAD technology within their IGA framework [52,53]. A similar trend is observed in patient-specific IGA (bottom right quadrant), which has been limited in use thus far. This is primarily because of the challenges in creating analysis-suitable NURBS geometries from imaging data using purpose built codes, without the assistance of a CAD library. Despite the difficulty, Morganti *et al.* have embedded a CAD-designed aortic valve into a patient-specific aorta geometry using Matlab [29]. When it comes to vascular NURBS modeling for IGA, the template approach proposed by Zhang *et al.* [25] is the only vascular network pipeline available in the literature. Though this approach allows for handling complex branching networks in a NURBS framework for IGA, the modeling process is still prohibitively slow for large, complex networks. Additionally, sampling the arterial geometry, which is required for an anatomically correct model, is difficult to automate and therefore slow to execute without utilizing CAD libraries. In the finite-element based vascular modeling community, this operation is often circumvented to speed up modeling time by representing the geometry as uniform diameter tubes [54,55]. This approach is inappropriate for many applications where flow features are strongly determined by local geometry variations such as in atherosclerosis [56,57] or aneurysms [41].

Considering the benefits of a NURBS-based vascular modeling approach, especially in the context of IGA, and the absence of an efficient, anatomically accurate, and semi-automatic modeling procedure, there is an overwhelming need for a NURBS-based modeling pipeline that can efficiently process complex vascular models for a large population of patient data. To these ends, this paper proposes a CAD-integrated vascular modeling pipeline (see the blue region in Figure 2). Leveraging CAD software and its built-in geometric modeling and editing operations allows for quick code development for ensuring the two critical vascular modeling requirements; handling arbitrary vascular topology, and accurately sampling the imaging data to impose anatomical accuracy. This novel pipeline makes possible the efficient creation of complex NURBS-based patient-specific vascular geometric models. `

The paper is organized as follows. In Section 2 we briefly review the concept of template-based vascular NURBS modeling. In Section 3 we discuss the prominent features of our proposed CAD-integrated modeling pipeline. To instantiate the procedure, an example creating an authentic vascular model of a mouse circle of Willis from a computerized tomography (CT) scan of a mouse brain is followed. In Section 4 we present three example applications of our methodology including a

mouse circle of Willis model, a complex porcine coronary artery tree model, and a human circle of Willis model. Finally, in Section 5 we discuss various key aspects of our modeling approach and future work.

2. Template-based vascular modeling approach

The CAD-based vascular modeling pipeline presented in this paper builds on the template-based methodology introduced by Zhang *et al.* in [25]. This technology has been widely utilized in IGA based cardiovascular research [12,13,24,26,27,58]. A brief review of the relevant steps is outlined below.

To create an analysis suitable model from the triangulated surface mesh generated from the segmented imaging data, first Voronoi and Delaunay diagrams are used to extract a centerline path, or skeleton, of the geometry. From there, an analysis template is formed from the skeleton by sweeping the control polygon of the cross-sections along the skeletal path. Zhang *et al.* [25] provide an extensive set of templates for handling a variety of possible arterial branch intersections. Once the geometry template has been laid out through the skeleton-based sweeping method, the geometry of interest must be fit to the template by adjusting the control points of the NURBS mesh. First, the interpolatory control points are moved along the radial direction to the true surface. Then, the non-interpolatory points are placed at the intersection of the tangent lines to the true surface passing through the two neighboring interpolatory points (see [25] for further details). This approach enforces G^1-continuity of the surface circumferentially and the analysis-suitable model is now complete. It should be noted that the geometry error of the NURBS model with respect to the imaging data can be improved by increasing the number of cross-sections assigned to the template and/or by increasing the number of control points used to define the cross-section itself. Computational fluid dynamics (CFD) analysis can now be implemented using the model to inform diagnosis, prognosis, and treatment.

The pioneering approach outlined above helped bring together the seemingly disparate components of medical image processing, analysis suitable model generation, and IGA. Inventing coupling between these steps is a difficult task considering the significant differences in technologies utilized, skills required, and user specializations. The template approach does well in defining a mapping from imaging data to geometric objects defined as multivariate splines, but is limited in its ability to handle topologically complex arterial networks. Hence, an advanced framework leveraging CAD functionality is necessary to successfully manage these difficulties.

The traditional engineering design-through-analysis pipeline starts with the use of a CAD system for the geometric modeling portion of the process and then uses this CAD model as a basis for computer-aided engineering (CAE) operations, in particular, computational analysis [56,57]. This paradigm has a long development history and, even as a highly refined methodology, continues to evolve in order to address the growing demands of both geometric modelers and engineering analysts [61,62]. As documented by Hughes *et al.* [23,30], moving from finite element analysis (FEA) to IGA in CAE allows for a higher degree of integration between CAD and CAE which saves time, reduces cost and increases the accuracy of the simulation. Recognizing this, Breitenberger *et al.* [63], Hsu *et al.* [50], and

Belibassakis *et al.* [64] have developed IGA tools integrated directly into CAD packages for the design and analysis of many structures, ranging from wind turbine blades to ship-hulls.

The majority of PSVM implementations to date follow the process shown in Figure 1 [29,42,47,65–84], starting with medical imaging data as an input and then processing the data into a suitable model for analysis, and finally applying computational analysis. This is a similar pipeline to the traditional engineering design-through-analysis pipeline described above, but with the additional medical image-processing task. In most PSVM pipelines, the modeling power offered by a CAD program is not employed because of the assumption that if the source of the modeling data is sampled from images, then the tools offered by a CAD resource are not appropriate [85,86]. This suggests that generating models from scratch is the only modeling capability and paradigm of a CAD system [87]. CAD systems are often misunderstood as a technology limited to modeling machined mechanical parts and not complex freeform, organic geometries, especially by the medical imaging and biomechanical communities [88]. The truth is that CAD software is agnostic to data source and should rather be thought of as a workbench of geometric and topological tools – a geometric library for all modeling purposes. Proof of this can be seen in CAD systems that explicitly offer resources for the medical field [89–91], or offer CAD based services and products for patient-specific modeling [92–94]. The patient-specific vascular modeling pipeline introduced herein extensively uses CAD libraries and functionalities.

3. Method

In this section, the CAD-integrated pipeline developed for the purposes of constructing spline representations of patient-specific vascular networks is detailed. The main steps include post-processing of a triangulated artery surface mesh generated from segmented 3D imaging data, extraction of spline centerlines from the triangulated surface mesh, construction of spline surfaces from the triangulated surface mesh and centerlines, and finally creating solid, or trivariate, splines from the spline surfaces. The workflow is outlined in Figure 3. The underlying mathematical equations and algorithms are beyond the scope of this paper, but can be found in standard CAD literature.

3.1 Triangulated Surface Mesh Post-Processing and Centerline Extraction

The triangulated surface mesh generated from segmented imaging data may contain geometric features and degeneracies that prohibit downstream processes. These items include duplicate vertices, duplicate edges and faces, non-manifold vertices and edges, holes and openings at arterial branch ends (missing end caps), and so on. These features are especially detrimental to the sensitivity of the centerline-producing algorithm. A mesh healing and repairing step is carried out semi-manually using MeshLab [38]. Like any geometry cleaning operation, care must be taken not to distort, smooth or remove important characteristics and attributes found in the raw data. This is largely carried out visually at the discretion of the user, although MeshLab does offer valuable metrics as feedback when applying various operations.

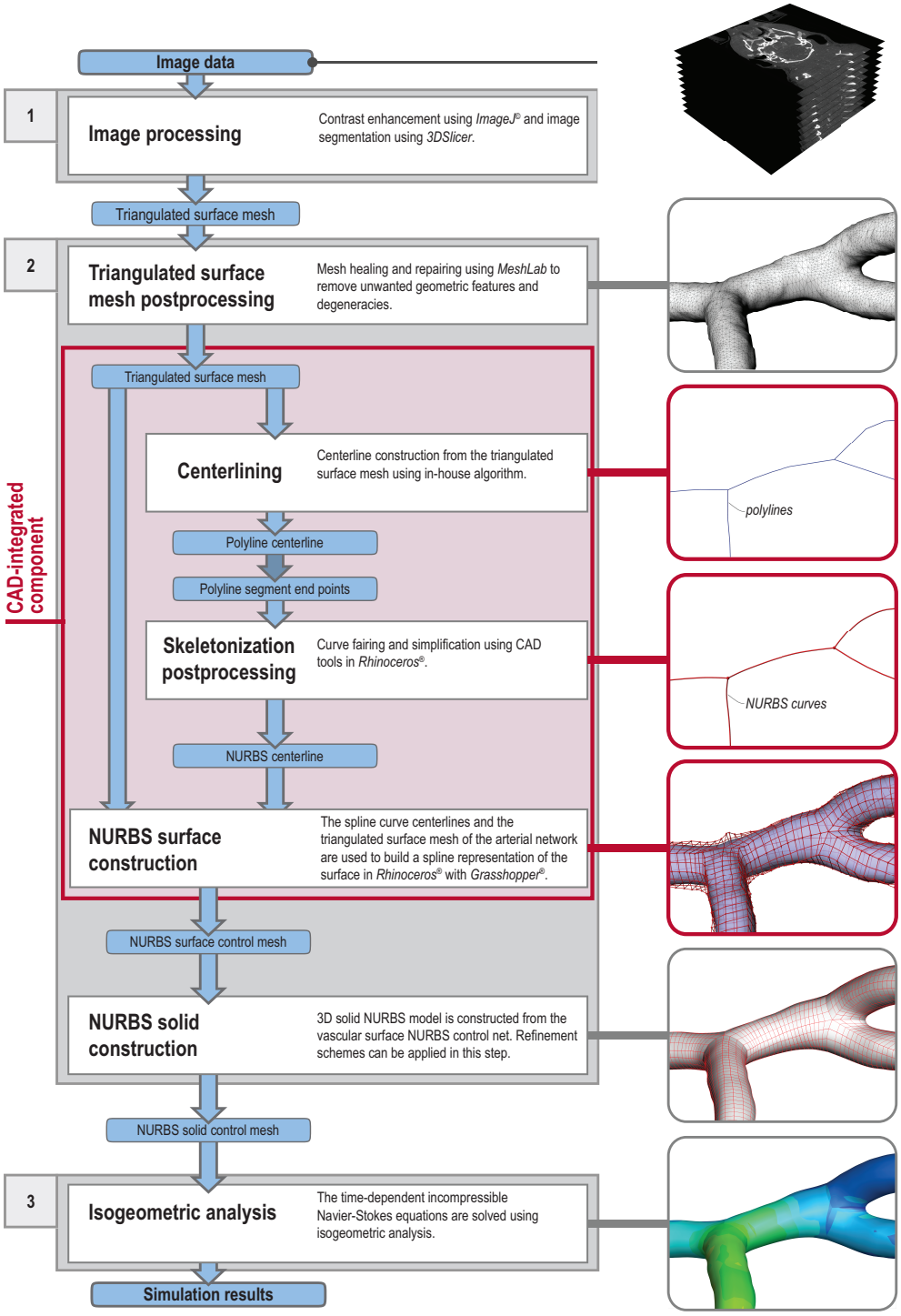

Fig. 3 The detailed vascular modeling pipeline implementation, including the CAD-integrated component.

In a sweep-based algorithm, centerline data is necessary for establishing the topology of the arterial network, as well as acting as the guide to the surface generator [95]. Many programs exist for centerline extraction, including starlab-mcfskel [96], VMTK [97], and Mimics Innovation Suite [98]. For our CAD-integrated pipeline, a simple procedure was implemented in Grasshopper [99] to compute samples of the centerline. In the triangulated surface mesh, each vertex along with its associated surface normal vector is used to compute one centerline sample. The centerline

sample for a specific vertex is the point halfway between the vertex and the point on the mesh lying across from it. The point across from the vertex is defined as the intersection of the surface mesh with the inward normal vector of this vertex. This procedure has worked well and produces practical results for vascular geometries. The centerline samples are then imported into the CAD software, and a subset of the points are manually selected as control points for the centerline spline representations.

3.2 NURBS Surface Construction

Next, we describe the algorithms used to build a spline representation of the arterial surface from both the triangulated surface mesh and the corresponding spline curve centerlines. We start by defining a few frequently used terms while walking through an example for a single artery branch with arbitrary shape (Figure 4).

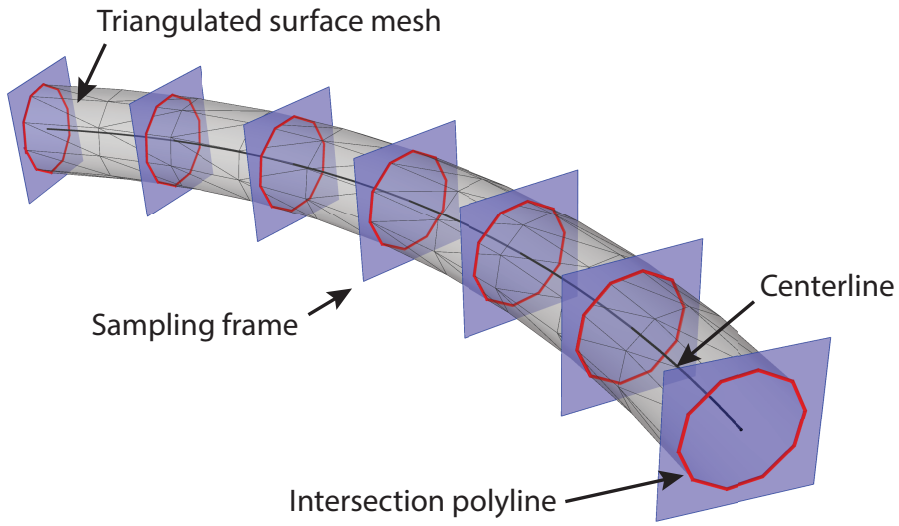

Fig. 4 Defining the sampling frame scaffold. The triangulated surface mesh and its associated centerline are provided as inputs. Sampling frames (blue cross-sections) are evenly spaced along the centerline. The sampling frame scaffold is defined as the set union of all the sampling frames. Once the scaffold is set, the intersection of the triangulated surface mesh with each sampling frame is computed with a Rhino command (red polylines).

Assume that a triangulated surface mesh, along with the centerline of the geometry, is provided as the input. Sampling frames are used to sample the surface mesh, in other words, to compute the intersection of the triangulated surface mesh with the sampling frame surface. The standard sampling frame is a square-shaped surface lying orthogonal to and centered about the centerline of the artery (the blue frames in Figure 4). A folded sampling frame is simply a standard frame that is allowed to fold like a greeting card about either of two possible folding axes (Figure 5).

The collection of all sampling frames for the model is referred to as a sampling frame scaffold (Figure 4). A sampling frame scaffold is said to be smoothly varying if the changes in orientation and folding angle of neighboring sampling frames is minimized. The result of sampling the triangulated surface mesh on the sampling frames is a polyline, or a continuous line composed of multiple line segments (Figure 4). The polylines from each sampling frame are then interpolated with a NURBS curve using the sampling frame template (Figure 5). Here, 9 control points are used to interpolate the line using the control point weights associated with a 9-control-point

circle. Other templates, including 13- and 17-control-point circles can be used to minimize the error in the geometry approximation. For this algorithm (though it is not required in general), C^0 continuity is enforced at four control points (1,3,5,7, as well as the repeated control point 9 at the location of control point 1). The sampling frame is able to fold about two axes: one passing through control points 1 and 5 and the other passing through control points 3 and 7. These two axes are referred to as the folding axes of the sampling frame. Optimal locations for control points 2, 4, 6, and 8 must be computed to best interpolate the intersecting polyline of the triangulated surface mesh with the sampling frame. Finally, the NURBS curves are lofted along the centerline to form a NURBS surface.

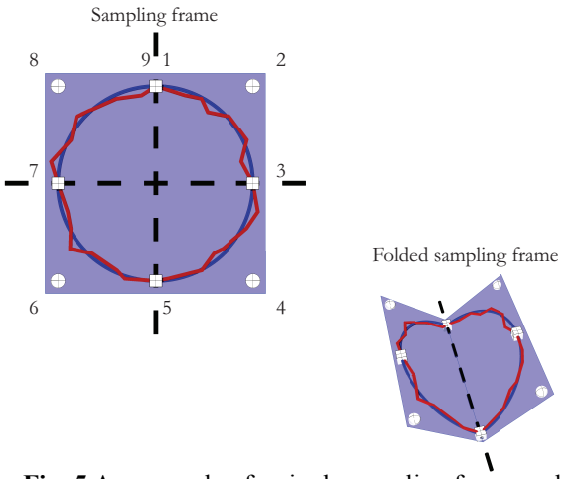

Fig. 5 An example of a single sampling frame and the 9 control points used to interpolate (blue curve) a polyline (red line). The two possible folding axes (black dashed lines) are shown. Interpolatory controls points (white cubes) lie on the folding axes so that the interpolated curve lies on the sampling frame even after folding (b). Spherical control points (2,4,6,8) are the control points to be adjusted within the sampling frame to interpolate the sampled polyline.

There are a few additional terms to define when handling arterial networks with more than one branch. An intersection point is the point where three or more arterial centerlines meet (Figure 8). An intersection is said to be conformal if the sampling frames along each centerline involved in the intersection do not cross each other. For this to occur, the centerlines of the branches at each intersection must share half of their sampling frame with their appropriate neighbor. The templated approach from [23] is used to ensure conformal intersections. Though it is possible to implement any type of intersection, only planar intersections, or intersections for which the involved centerline branches join on a plane, are implemented in the current algorithm since these are by far the most commonly seen in patient data. Also, to simplify the language, all intersections are assumed to be bifurcations (one arterial branch splitting into two) in the following description of the algorithm. The generalization for other planar intersections is straightforward.

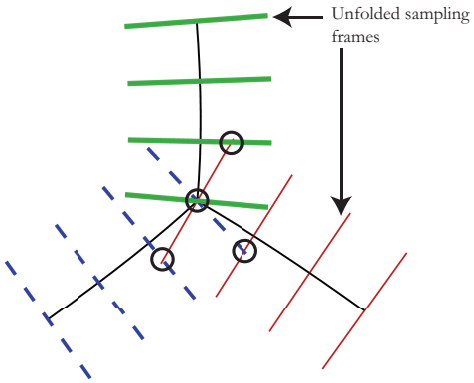

Fig. 6 Intersecting sampling frames, like the ones circled here, are prohibit the creation of analysis-suitable geometries. The sampling frames on each branch (light red, dashed blue, bold green) overlap each other (circled) near the intersection point. To prevent this, a folding scheme is introduced to specifically handle the sampling frames near intersection points

The Grasshopper script is divided into three main tasks; (1) creating smoothly varying sampling frames along the arterial network centerlines with conformal intersections, (2) sampling the triangulated surface mesh on each sampling frame and

finally, (3) lofting a smooth NURBS surface. In the following subsections, each of these steps is explained in detail.

3.2.1 Sampling Frame Scaffolding

The goal of this task is to create a smoothly varying sampling frame scaffold with conformal intersections. To achieve this, a modularized framework is employed leveraging the powerful and robust algorithms included in the CAD software Rhino and Grasshopper, along with the available RhinoCommon SDK (software development kit).

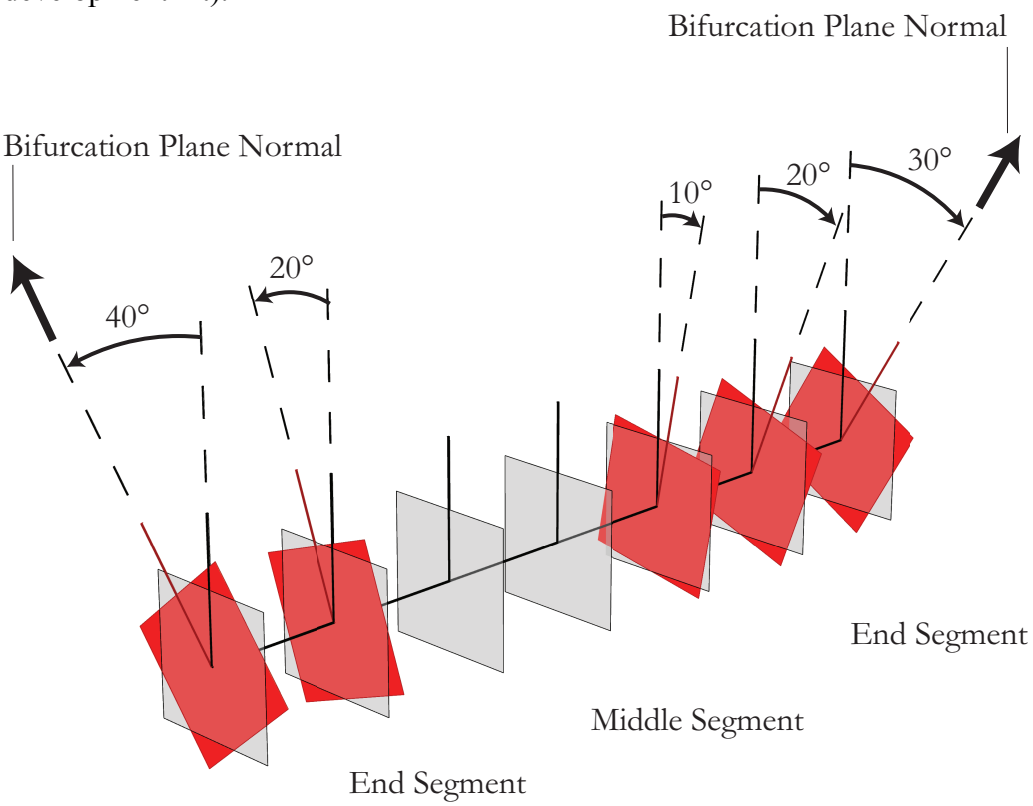

Fig. 7 An example of sampling frame rotations near intersection points. Both ends of the branch shown are connected to two other branches. Therefore, each end is an intersection point. The normal of the intersection plane is the labeled solid black line on each end. To create the sampling frame scaffold for this branch, first, the dashed sampling frames are evenly spaced and oriented according to the local normal and bi-normal vectors of the curve. Next, for the sampling frames on the ends, the amount of rotation required to align either the normal or bi-normal vector with the intersection plane normal vector is computed (40° for the left side and 30° for the right). Finally, the frames are sequentially rotated back to align with the sampling frames in the middle section.

For a single artery, the naïve scaffold presented in Figure 4 works perfectly well. However, for a network of connected arteries, the naïve approach must be improved for two reasons: (i) to form a valid, parameterized NURBS surface, sampling frame scaffolds must have conformal intersections (Figure 6) and therefore the frames must appropriately fold near intersections, and (ii) for branches with an intersection at each end, properly folding the sampling frames on both ends requires an incremental rotation of the frames about the centerline (Figure 7).

To prevent sampling frames from crossing each other, the frames are folded near each intersection point. To accomplish this, first, an intersection plane (Figure 8) is defined at each intersection point using three points lying on each centerline at a specified

offset distance ε from the intersection point. The intersection of the triangulated surface mesh with this intersection plane forms three separate polylines (Figure 8). The closest points on each of these polylines to the intersection point are found, and then lines are formed from each of these points to the intersection point itself. Extruding these lines perpendicular to the intersection plane defines the sampling frames at the intersection for the three centerlines involved. This procedure guarantees that the sampling frames have conformal intersections.

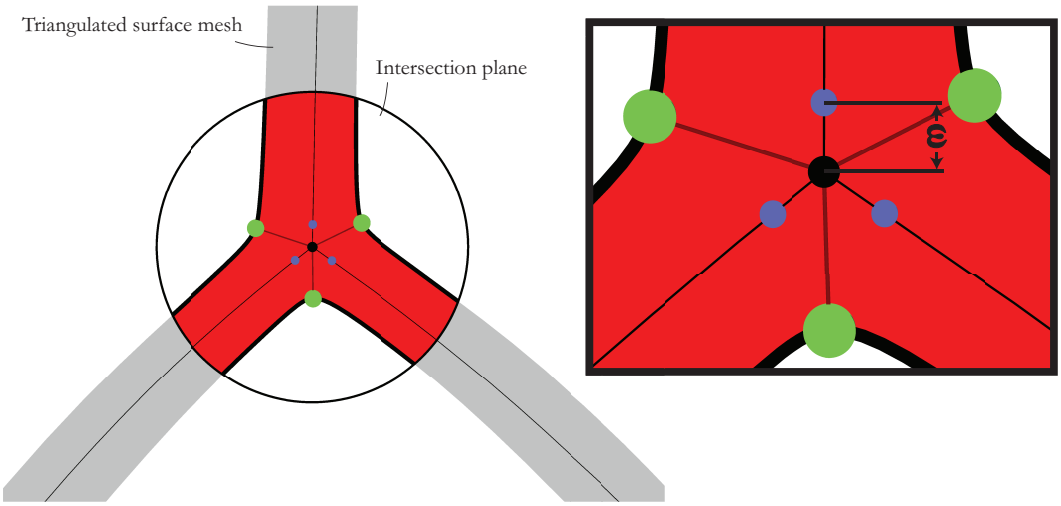

Fig. 8 Three centerline branches join at an intersection point (center black dot). An intersection plane (circle) is defined by three points (blue dots), one on each branch, a small distance ϵ from the intersection point. The intersection of the intersection plane with the triangulated surface mesh is computed and shown as 3 bold and black polylines. The closest point on each of these polylines is computed (green dots) and a line is drawn to connect these points to the intersection point. These lines help to define the folding of the sampling frames required on each branch at this intersection point.

To transition from a sampling frame folding angle of α ($< 180°$, see Figure 9) at the intersection point to a perpendicular sampling frame in the middle section of the artery segment, the sampling frames must successively unfold moving away from the intersection point. This is done by adding an angle of $\beta = \frac{180°-\alpha}{n}$ to each successive sampling frame for n number of frames, where n is specified by the user. This approach naturally divides each arterial branch into three segments; the middle segment where standard sampling frames are used (lying perpendicular to the centerline), and two end segments where sampling frames are gradually folded into their corresponding intersection point. In practice, the two end segments can take some effort to handle correctly. For instance, centerlines might need to be adjusted near intersection points to define an appropriate intersection plane, which intersects the triangulated surface mesh in such a way that *three* distinct polylines are formed. Also, centerlines might need to be corrected to form an appropriate α angle for each of the three branches joining the intersection point. Ideally, for bifurcations, $\alpha \cong 120°$ is an optimal angle between each pair of branch centerlines. Very small α angles are more likely to lead to non-conformal sampling frames. Fortunately, CAD software allows for easy adjustments of curves through control point manipulation and an intuitive and powerful GUI.

For the second alteration to the naïve scaffolding approach, the orientation of the sampling frames about the centerlines is considered. The orientation of a standard (unfolded) sampling frame can be determined by the direction of the folding axes of the frame. As described above, the normal vector of the intersection plane determines the orientation of a sampling frame at an intersection point because one of the folding axes must be aligned with it to fold in a conformal way with the other branches involved in the intersection. If the other end of the branch is free (does not have an intersection point), then the problem of orienting the other frames is simple; namely, eliminate any relative rotation of the folding axes between neighboring sampling frames. If, however, the other end of the branch is another intersection point, the discrepancy between the orientations of the two sampling frames on either end of the branch must be reconciled. The approach taken here (Figure 7) is to (1) orient all the frames in such a way that eliminates the relative rotation of the folding axes between neighboring sampling frames; (2) compute the angle, β, between the normal vector of the intersection plane and the two possible folding axes of the sampling frame at the intersection point (choose the minimum angle between the two options for the folding axis); and finally (3) reorient the sampling frames in the end segments such that the orientation of each frame relative to the orientation of the sampling frame on the end is increased by $\frac{\beta}{n}$ degrees, where n is the number of sampling frames in the corresponding end segment. For example, in Figure 7, the sampling frame on the left-most end has $\beta = 40°$ while the frame on the right has $\beta = 30°$. Since the end segment on the right has three sampling frames, each subsequent frame is rotated by 10° each moving towards the middle segment.

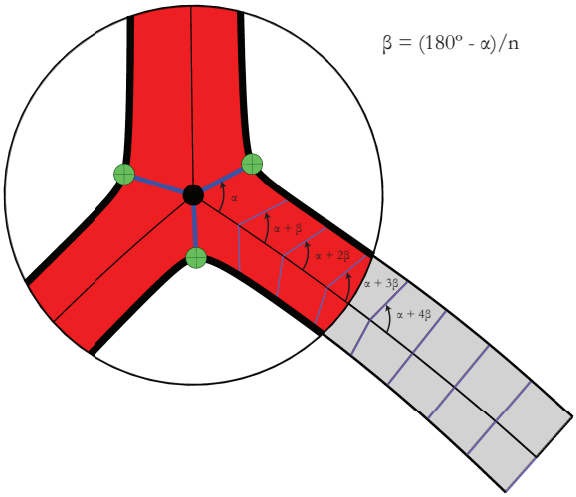

Fig. 9 Example of sampling frames folding to an intersection point (black dot). The sampling frame on the end is folded to an angle of α. Each sampling frame is then sequentially unfolded until the middle section sampling frames are reached.

In summary, the correct order of operations to form the sampling frame scaffold is to execute the following steps for each centerline branch of the arterial network. If both ends have intersection points, perform the sampling frame rotation steps, followed by the sampling frame folding steps. On the other hand if only one end has an intersection point, perform steps (1) and (2) of the sampling frame rotation steps. Then apply the rotation found in step (2) to every sampling frame on the branch, followed by the sampling frame folding steps. Finally, after every sampling frame on each branch has been positioned and oriented according to the folding and rotation scheme described above, the sampling frame scaffold is complete. The polylines defined by the intersection of each sampling frame with the surface mesh are computed.

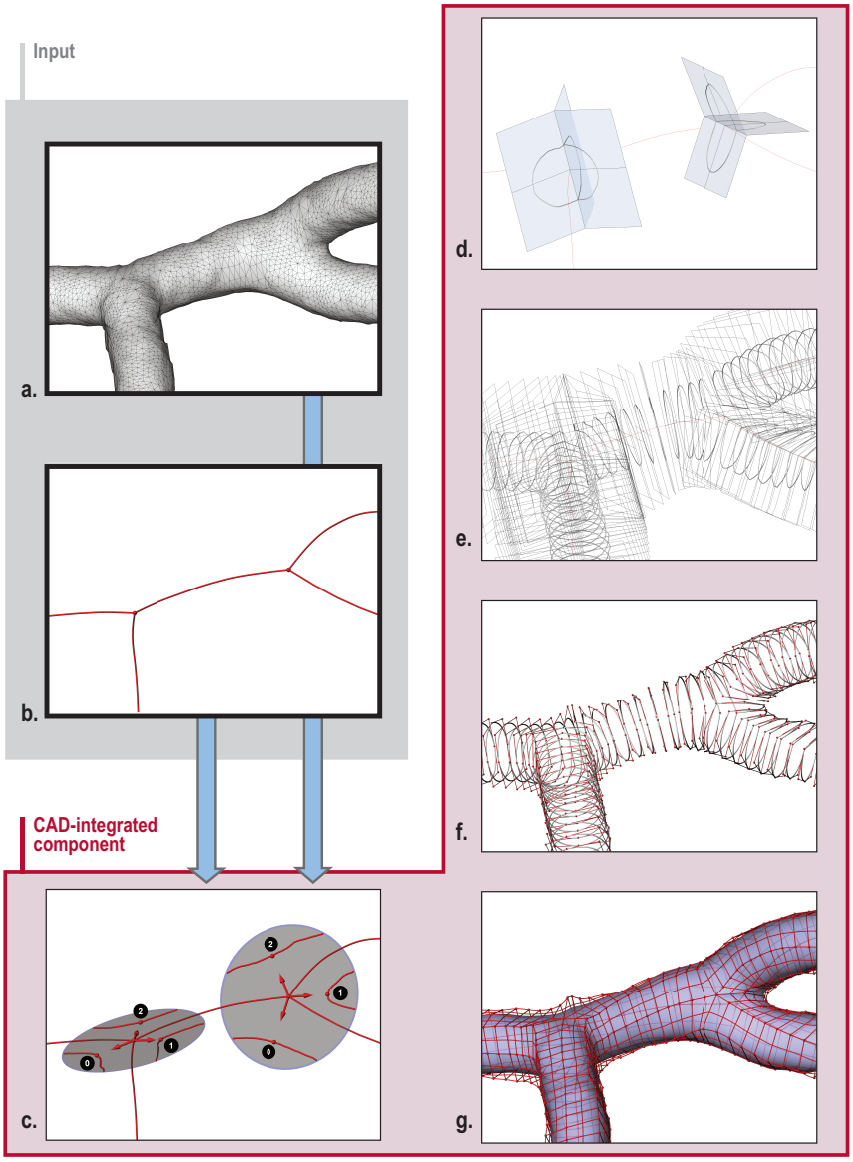

Fig. 10 The steps in the CAD-integrated workflow in creating a vascular model of a mouse circle of Willis. The input data consists of a) the triangulated surface mesh of the arterial network and b) the spline curve centerlines. Next, the CAD-integrated implementation sequentially processes the input by c) defining the intersection planes are each intersection point and d) defining the sampling frames at each intersection point. Next, e) all sampling frames are found by computing the proper rotation and folding angle for each frame and the mesh is sampled to form polylines on each frame. Then, f) the polylines are interpolated with an n-point NURBS curve (9-point template used here) on each frame and g) the NURBS curves are lofted to form the NURBS surface.

3.2.2 Interpolation and Lofting

To form a NURBS surface, each polyline must be interpolated with a B-spline curve and then all the frames' B-spline curves can be lofted to form a NURBS surface. For this stage, the polylines from each sampling frame are provided as the input. The interpolation scheme produces the correctly ordered and oriented NURBS curves as output. The location of the template control points can then be adjusted to best fit the polylines. The only requirement for this algorithm is that the template must have interpolatory control points on the two folding axes. This is to ensure C^0 continuity at

these points, which implies that the template can fit onto a folded frame (see Figure 5). Various interpolation algorithms and fitting operations can be dynamically compared and swapped in real-time. Topological information, including which control points on adjacent branches are collocated, is also produced in this step. Figure 10 depicts an overview of the various steps involved in generating the NURBS surface mesh of a portion of a mouse circle of Willis vasculature. The control points are ordered for direct exporting along with the center point of each sampling frame. The center point is necessary for creating the volume mesh. Depending on the use of the data in downstream stages outside the CAD package, various export schemes can be utilized to write out file types of differing formats; either standard CAD formats or user-specific formats.

3.3 Volumetric NURBS Mesh Generation

A 3D solid NURBS model is created from the NURBS surface generated above. A control net in three coordinate directions with respective weights must be specified along with a trivariate knot vector set to generate the solid NURBS model. Figure 11 illustrates an idealized artery segment. The coordinate directions in the parametric space are labeled ζ, η, and ξ representing the circumferential, radial, and axial directions, respectively (Figure 11a). Each cross section (holding ξ constant) contains 9 circumferential control points where the first and last control points coincide, and two control points in the radial direction. The 9 innermost control points (in green) are then collapsed into a single point on the centerline (Figure 11b). This forms the cross-section for the lumen, whose surface edge is a smooth interpolation of the bounding triangulated surface mesh that can be connected to neighboring cross-sections to create the solid NURBS cylinder. Figure 11c shows the solid NURBS control net for a subject-specific artery segment. Any choice of degree and knot complexity can be introduced in the three coordinate directions in order to capture desired geometric details and to provide higher order of continuity for more accurate solutions.

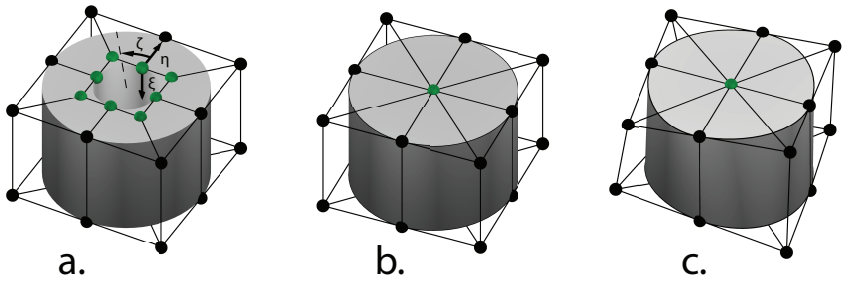

Fig. 11 The definition of a solid NURBS model of an idealized artery: a) a cylindrical annulus with nine outer control points (one duplicate point) and nine inner control points (green), b) the inner control points are moved to coincide on the centerline of the cylinder, thus constructing a solid NURBS model of an idealized arterial lumen. c) The solid NURBS control net for a subject-specific arterial segment.

4 Results

In this section we present applications of our modeling approach with three example geometries, two in which realistic blood flow features were simulated (see Figures 12, 13, and 14). Figure 15 provides error maps and corresponding histograms as a measure of modeling accuracy.

4.1 The Circle of Willis of a Mouse

A 3D image stack of a mouse circle of Willis (Figure 12a) with an isotropic resolution of (19 um)^3 was obtained from a previous study conducted by Zbigniew *et al.* as detailed in [100]. Image segmentation was performed using a vascular level set segmentation module in 3D Slicer [101] which utilizes two methods: a fast marching upwind gradient initialization and a geodesic active contours
evolution [102]. Because of the complex vasculature surrounding the arteries of interest, seeding did not entirely isolate the circle of Willis artery structure from other vascular features (e.g., veins). Further segmentation procedures, including centerline extraction and an extruding method, were required to fully isolate the arterial vessels (Figure 12b). From there, an automatic mesh generator in 3D Slicer was used to create a triangulated surface mesh of the segmented arteries (Figure 12c).

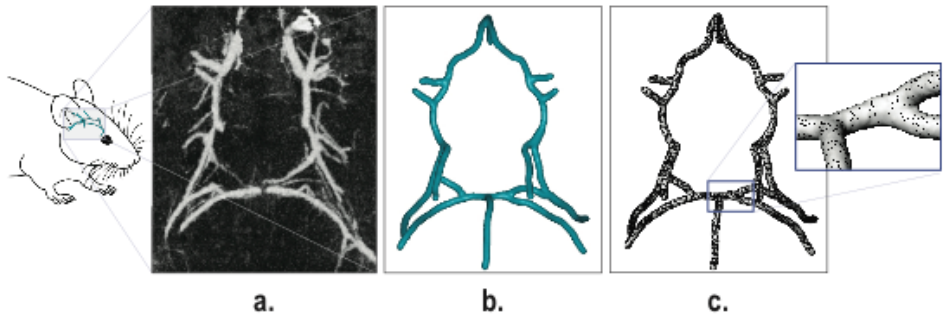

Fig. 12 Image processing: a) medical image capture, b) image segmentation, and c) triangulated surface mesh generation.

This triangulated surface mesh (Figure 13a) is then fed as an input into our CAD-based vascular modeling pipeline to generate a hexahedral control net (Figure 13b), from which a 24-patch solid NURBS mesh (Figure 13c) is constructed following the procedures outlined in section 3. The resulting quadratic NURBS mesh is C^1 everywhere except at the interfaces of two branches, and consists of 112,128 control nodes and 91,080 hexahedral Bézier elements. A boundary layer mesh, where the finest boundary element thickness is of the order of 10^{-8} meter, is implemented for more accurate computation of near wall quantities such as wall shear stress. Finally, blood flow simulations are performed adopting the general solution strategy outlined below.

Blood flow was modeled by the time-dependent incompressible Navier-Stokes equations subjected to the following boundary conditions. A no-slip boundary condition was implemented at the rigid vessel wall. A time varying pulsatile inflow velocity with a parabolic profile taken from [103] was prescribed at the two internal carotid artery inlets as well as the inlet of the basilar artery, while a traction-free boundary condition was set at each of the remaining branch outlets. Blood was considered to be a Newtonian fluid with a density of 1060 kg/m^3 and a dynamic viscosity of 0.0035 Pa-s. A Navier-Stokes solver within an IGA framework was utilized to run the flow simulations for one cardiac cycle with a time-step of 0.05 seconds. The details of the methodology including the governing equations and the solution approach can be found in [13,27] and references therein. The results are presented in Figure 13d in terms of time-averaged wall shear stress (TAWSS) [12]. TAWSS has implications in thrombus formation and stroke. Pathologically high WSS regions are susceptible to clot formation leading to stroke in subjects affected by the

Moyamoya disease, a rare condition caused by blocked arteries at the base of the brain.

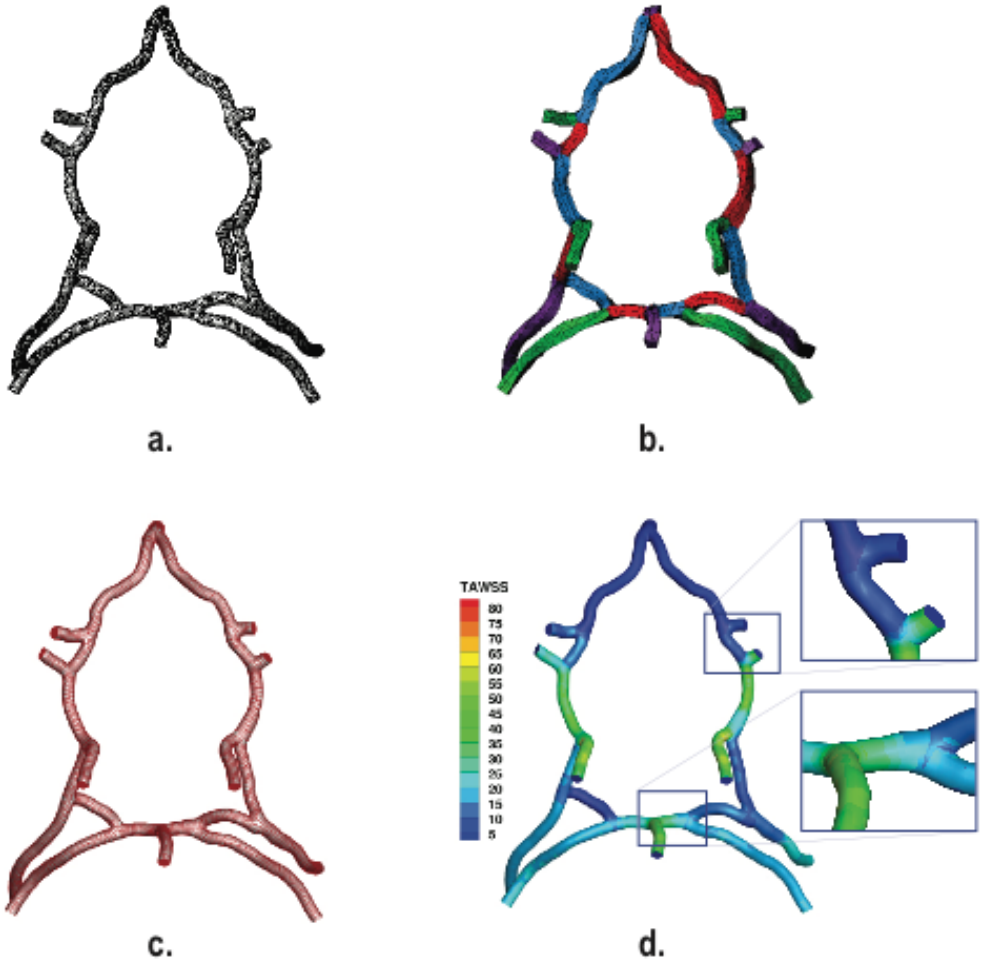

Fig. 13 CAD-based vascular modeling pipeline applied to an example problem: a) triangulated surface mesh of the circle of Willis network in a mouse, b) 24-patch hexahedral NURBS control mesh, c) analysis suitable solid NURBS mesh, and d) flow analysis, where the results are reported in terms of spatial distribution of time averaged wall shear stress (TAWSS) in Pascal (Pa).

4.2 Porcine Coronary Artery Tree

3D MRI images of a porcine coronary artery tree were acquired at a resolution of 0.4 x 0.4 x 0.4 mm^3 and reconstructed at 0.14 x 0.14 x 0.2 mm^3 resolution. Segmentation of the arterial lumen was performed in ITK-SNAP using a region-growing algorithm. From the segmentation, a triangulated surface mesh was generated (Figure 14a) and the centerlines of the arterial network were extracted (Figure 14b), which were then processed through our CAD-based vascular modeling pipeline to create a NURBS control net. Finally, a 107-patch solid quadratic NURBS-mesh (Figure 14c) that is C^1almost everywhere and consisting of 399,685 control nodes and 260,271 hexahedral Bézier elements was constructed for performing IGA. Following the general methodology described in the previous example, flow simulations were carried out (Figure 14d).

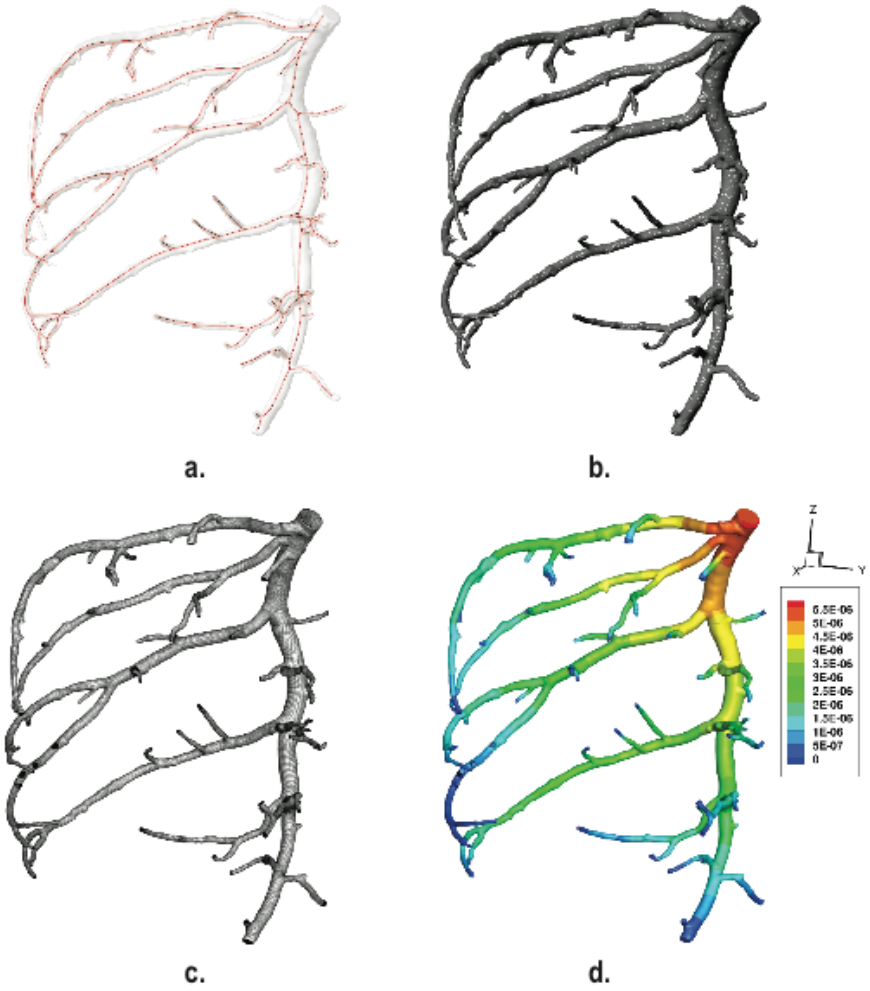

Fig. 14 CAD-based vascular modeling pipeline applied to a porcine coronary artery network: a) triangulated surface mesh medical imaging export, b) arterial centerlines, c) 107-patch analysis suitable solid NURBS mesh, and d) flow analysis, where the results are reported in terms of Pressure in kPa

5 Discussions

5.1 PSVM Pipelines: CAD-based PSVM Versus Current Implementations

As demonstrated above, CAD systems offer a tremendous benefit to developers building PSVM pipelines, modelers within the model creation stage, and analysts in the simulation phase. In lieu of such resources, an enormous gap must be filled with manual labor, original code, and the misapplication of non-CAD software for the purposes of generating and modifying geometry. The practice of developing specialized pipelines in this way fails to benefit from the advantages offered from the use of a CAD system, including robustness, speed, accuracy, interoperability, user-friendliness, and simple and intuitive geometric manipulations [104,105]. Whether the source of the geometry is from interpolated imaging data or instantiated directly by a designer, the use of CAD infrastructure is critical for creating and editing any geometric model. As helpful as purpose-built vascular modeling utilities such as VMTK and pyformex are, without CAD integration they suffer from needlessly

recreating and supporting CAD functionality without an optimized geometric-modeling core that is regularly updated and enhanced by specialists.

As an example, the visualization toolkit (VTK) library does not natively support NURBS surfaces or solid model Boolean operations [106] which is the backbone of engineering surface and solid modeling applications [105], and hence is not used to develop CAD models. Alternatively, in a review of 7 vascular modeling applications, it was found that none of these applications integrate a CAD component; rather, the majority are founded on VTK, a library for visualization post-processing tasks, not geometric modeling [107]. The huge development and maintenance cost of a geometric modeling and analysis component is not often appreciated outside of the computer-aided technology (CAx) community. This task has actually shaped the entire CAD industry itself by dividing CAD systems into two key parts: a core geometric processing component, called the kernel, and the CAD application's higher level tasks and interface framework, which draws directly on the CAD kernel's functionality. In a recent survey of 60 industry leading CAD applications, a mere two CAD kernels were found to provide the geometric modeling basis for the majority of these products [108]. Recreating the functionality of these powerful kernels is unnecessary for developing geometric modeling applications. While recognized within the CAx community, this understanding needs greater traction with those working to develop PSVM software tools.

Given the vast number of unique PSVM strategies in use ([5,40,41,48,54,74,109–126], to cite a few), it is appropriate to say that no single approach has proven to be dominant in practice. Instead, many methods with many interchangeable components are commonplace. This trend in PSVM naturally lends itself to a modular design and hierarchical software development. Whether it is a centerline algorithm, vessel boundary interpolation, defining quality metrics, or any other step, multiple approaches for each element in the pipeline must be efficiently trialed, developed, implemented, and compared. An object-oriented programming approach built on CAD infrastructure effectively facilitates this task. Fundamental geometric modeling objects and methods are not unnecessarily reinvented. The presented approach does not limit the scope of development in any way as external libraries, such as ITK or VTK, can easily be coupled in the development process. These libraries and tools are used to complement the modeling pipeline rather than being misused for geometric modeling operations. For instance, once the first step of PSVM (image processing) is complete, the data is geometric in nature and therefore are no longer manipulated by an image-processing library. Instead geometric data is handled entirely in the CAD system.

The field of geometric modeling itself is replete with its own professionals and technicians, even for scanned data from imaging sources. In fact, the entire field of "reverse engineering" is well founded on the common task of rebuilding CAD models from sampled point cloud data. The subtleties in creating models are best mastered as a craft through years of experience with such tools.

5.2 The Benefits of a CAD-based PSVM Pipeline

CAD systems allow for various modes of interaction for the user. This grants the developer the option to use their preferred choice in their own pipeline, as well as the power to implement a multi-leveled development strategy. Schemes can be quickly prototyped and tested at a high level and then, when refined, ported to streamlined

low-level code, similar to how programmers often port MATLAB and python prototyping to C and C++ executable files. Listed below are interface modes, from high-level to low-level, available from the leading free, open-source, or proprietary CAD packages commonly used in practice [31–34,127–135]:

- CAD application use through conventional graphical user interface or GUI, with an array of file formats to import and export from
- Script or macro execution within CAD application
- User executable plugins or add-ins within the CAD application
- Custom external applications built for real-time interaction with CAD application
- Custom applications built on top of CAD kernel

These interfaces are generally supported with multiple programming and scripting languages in each application, so the user has the option of developing in C++, C#, Python, Java, JavaScript, Visual Basic, VBScript, Delphi, or another language. This contrasts a second incorrect assumption about CAD within the PSVM community: that traditional CAD applications do not have a multi-level user interface that supports scripting, user specific code, and inclusion of external libraries[136]. Additionally, CAD systems may offer visual programming languages, app development tools for tablet, mobile, and other hand held devices, and rendering development frameworks [33]. The APIs (application program interface), SDKs (software development kit), and libraries available from CAD developers are made friendly to use by detailed documentation, templates, wizards, as well as integration in numerous IDEs (integrated development environments). The available resources and interface modalities make barriers to use and integrate CAD functionality extremely low and the learning curve very small. Another important, yet subtle, point should also be made: too frequently researchers in mathematical modeling and analysis overlook the benefits a good graphics environment offers. Using the established GUI of a CAD program allows for visual inspection of algorithm execution, real time parameter manipulation with windowing controls, rendering capabilities, annotation and dimensioning tools, visual interrogation with panning and rotating objects, as well as graph and data plotting. The ability to quickly and easily personalize this environment helps refocus research efforts on more critical tasks.

Unfortunately, the model generation process has reduced model quality so much that in some cases the “patient-specific” label may no longer be applicable. For instance, to accurately compute near wall quantities in computational fluid dynamics, a smooth boundary of the fluid domain is imperative [41,137,138]. A NURBS-based model of vascular networks is ideal for smoothly representing the patient-specific geometry. In addition, an anatomically accurate model, as opposed to simplifying the geometry to

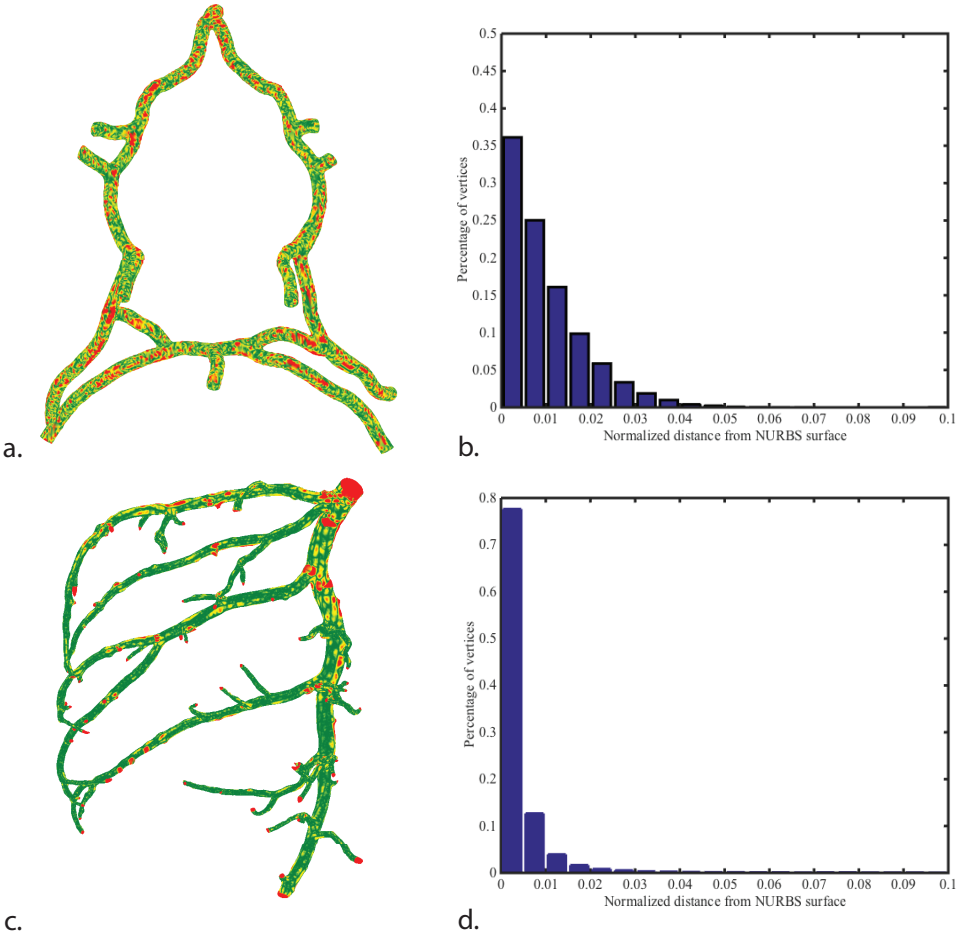

Fig. 15 Model accuracy: (a) and (c) are the color maps of the distance from the original triangulated mesh vertices to the modeled NURBS surface for the mouse circle of Willis model and pig coronary artery model, respectively. (b) and (d) show respective histograms of the same information where the *x*-axis represents the normalized distance from triangulated mesh vertices to the NURBS surface and the *y*-axis represents percentage of vertices belonging to each bin.

uniform-diameter pipes, is necessary for studying arterial diseases such as atherosclerosis and aneurysms [57,138–141]. In a review of 65 Fontan modeling studies, the majority of the approaches were shown to use idealized tubes of constant diameter to model the arterial blood flow, the consequence of the difficulties involved in building patient specific vascular models [142]. Thus, the quality of the analysis suffers from a lack of tools to efficiently build precise patient-specific models. Utilizing the geometric libraries of a CAD system allows for accurate cross-sectional sampling of the triangulated surface mesh along the branch centerlines. Figure 15 shows the accuracy of the NURBS geometries used in simulation (Section 4) with respect to the sampled surface meshes. The left images depict color maps of the distance from the triangulated mesh vertices to the NURBS surface for the mouse circle of Willis model (a) and the porcine coronary artery model (c). Histograms of the same data are provided in (b) and (d) where the *x*-axis represents the distance normalized by the maximum luminal diameter. The distribution of the histograms, with most of the NURBS surface showing less than a 5% error in luminal diameter, suggests the model indeed closely follows the anatomy of the sampled mesh. Reducing this error can be accomplished by increasing the number of sampling frames placed along the centerline and increasing the number of control points used to interpolate the polyline intersection on each sampling frame.

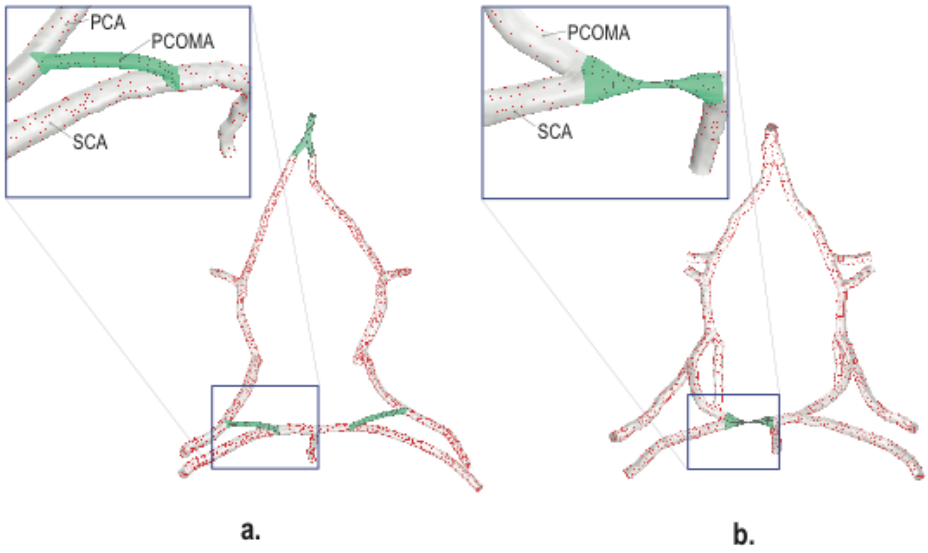

Fig. 16 CAD feature addition: (a) sometimes arterial branches with slow blood flow are difficult to see on CT scans using liposomal contrast agents, but can be created in CAD for analysis, and (b) a blockage was introduced in the superior cerebellar artery (SCA) to study the effects of occlusions observed in subjects with the moyamoya disease. Here PCA is the posterior cerebral artery and PCOMA is the posterior communicating artery.

The complexities involved in creating patient specific models have shown to not only affect model quality, but also increase processing time [56,121,143], limiting the pipeline throughput. Because PSVM pipelines are often semi-automated, reconstruction of vascular geometries is generally burdened by the limited number of tools for completing the task in a suitable amount of time. Consequently, model creation times for a CAD-integrated approach vastly outperform alternative approaches by at least an order of magnitude. This observation is based on our experience with arterial networks of varying levels of network complexity. The most complex model, the porcine coronary artery geometry presented in this work, would be too arduous to attempt with the previous NURBS-based PSVM pipeline [25], while the CAD-integrated pipeline produced a NURBS-based model in approximately 20 hours. Similarly, the 24-branch mouse circle of Willis model was generated in roughly three hours, as opposed to requiring three weeks using a purpose-built code. Similarly, a three-branch healthy coronary artery model [10,25] took minutes, instead of hours, to reconstruct. The quicker turn-around this code provides is crucial for implementing NURBS-based vascular modeling in a clinical setting. Further improvements in the efficiency of model construction can be accomplished with machine learning approaches. Because the final geometry in this approach is available to the user as a CAD object, thousands of variations of a single geometry can be created and used to train a machine learning algorithm to drastically decrease model generation time.

Finally, the ability to quickly create geometric features is a useful advantage of a CAD-based modeling approach. Geometric features not originating from imaging data can be efficiently edited and/or manipulated using a CAD library. Such features may arise through missing detection in the images themselves such as small arteries or by the insertion of objects entirely absent from the images. These may include stents, catheters, potential bypass grafts, a theoretical clot or stenosis, or left ventricle assist devices (LVADs). An instantiation of this idea is contained in the heart-valve

work of M.-C. Hsu *et al.* [51] in which hard-to-image heart valves are created in a CAD design system and immersed in an isogeometric arterial flow model. Many unique cases highlighting the necessity of feature addition using our CAD-integrated pipeline have been experienced (Figure 16). In the study of mouse cerebral vasculature, vessels connecting the major arteries, for example the posterior communicating artery (PCOMA) connecting the superior cerebellar artery (SCA) and the posterior cerebral artery (PCA), could be difficult to clearly identify on the CT images under certain physiological conditions although their existence is confirmed in histological studies. In addition to the limitation of imaging resolution, this could also be due to lack of sufficient liposome-laden blood flow through these smaller caliber vessels, making them less visible. To circumvent this limitation, these branches can be added after-the-fact (Figure 16a), and then various probable configurations as well as disease scenarios can be studied with the help of flow simulation. Similarly, the effect of blockages, a common feature in the circle of Willis of Moyamoya disease patients, on the local flow features could also be investigated by introducing constrictions in the most probable locations determined based on a comprehensive study of a large population CT image data (Figure 16b).

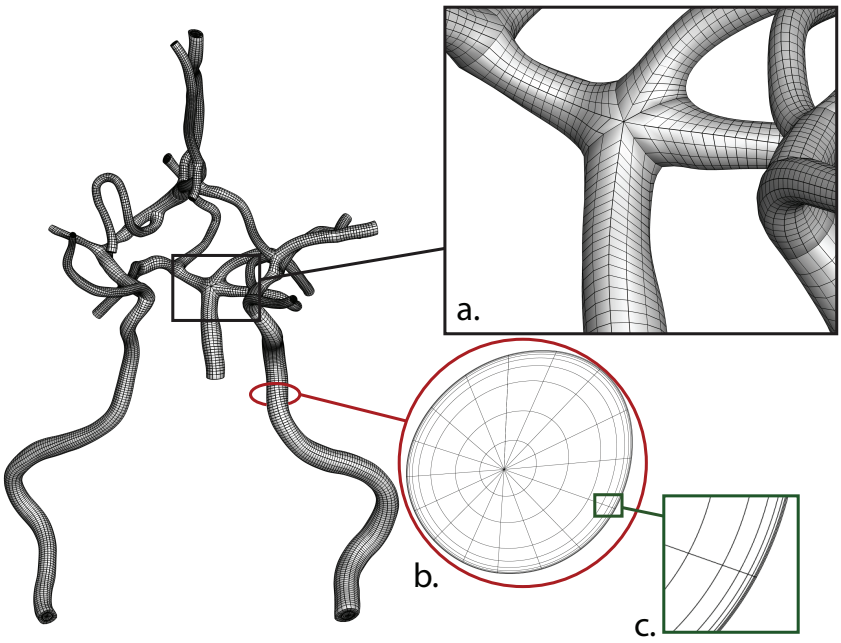

Fig. 17 The circle of Willis artery structure in a human patient. (a) A planar trifurcation is shown as well as (b) a cross-section of the geometry and (c) a close-up of the mesh boundary refinement.

5.3 Template-based PSVM Methods: Pluses and Minuses

The template-based PSVM modeling paradigm introduced by Zhang *et al.* [25] has enabled and accelerated PSVM IGA technology [12,13,24,26,27,58]. The difficulties of taking discrete voxel-based imaging and expressing it in a smooth representation is a huge undertaking. A template-based method is conceptually straightforward to understand. This makes the methodology implementable for a vast number of developers, immediately yielding superior results to those mentioned utilizing the standard PSVM pipeline.

Moving forward, there are critical issues that need to be resolved for the template-based methodology to become fully automated. First, a reliable procedure needs to be developed to automatically select the templates required for the downstream fitting operations. This is a non-trivial process as it is not simply an exercise in topology, but potentially also geometry as certain attributes make particular configurations more applicable to others [144]. As evidenced in Section 3, the "pipe-fitting" algorithms can be refined to give the most accurate interpolation, but it is the "pipe joint selection" that is critical for this endeavor to even begin.

Certain challenging issues seem to reappear while utilizing the template-based method with real patient-specific data. Organic models can often include complicated configurations such as two intersection points being very close to one another. Even trifurcations, though rare, can appear. In the prior case, this can make the conformal unfolding and twisting of the sampling frames (Section 3.2.1) very difficult to achieve. Figure 17 shows the solid mesh of the circle of Willis of a human patient with a planar trifurcation. To handle non-planar trifurcations, another template would need to be coded taking into account all joint geometries possible with four converging branches. This is very difficult to generalize to *n*-furcations. Also, when two arterial branches with much different radii (> 2:1) are attached to each other, the centerline in the larger vessel may need to be shifted towards the start of the smaller vessel to form compliant sampling frames near the intersection point. Finally, template-based methods can be sensitive to bad mesh data. If the input mesh is not watertight, contains degeneracies, or shows features related to regions of poor image quality, the template-based process could give poor results. PSVM algorithms require robustness in interpolating scanned data sources of unpredictable localized quality.

5.4 Future Directions in CAD-based PSVM

The steps taken in the pipeline outlined in Section 3 represent an initial implementation of a CAD-integrated, template-based patient-specific vascular modeling solution. One immediate improvement would be to consolidate the solid model creation into the CAD-integrated NURBS surface program. Currently, the trivariate building operation (see Section 3.3) is done with a purpose-built code using control point information extracted from the NURBS surface geometry. Incorporating this step into the CAD process, instead of interfacing with an outside program, produces a simpler pipeline. More interpolation methods with a sensitivity analysis between geometric precision and analysis behavior would be beneficial.

Transitioning to T-spline technology as a more compact and natural data structure for PSVM applications is another option moving forward. The first implementation might take the form of a solid (trivariate) T-spline that interpolates the triangulated surface mesh with interior parameterizations suitable for element extraction [145]. The unstructured T-spline definition allows for local refinement as well as a means of handling extraordinary points [146], an unavoidable feature of these vascular network geometries. A more direct interpolation of the voxel data should also be investigated so as to minimize the number of geometric approximations in the pipeline. Removing operational dependences on centerlines would go a long way in addressing some of the issues discussed with template-based methods. This also requires the development of specific metrics in which to determine accuracy and quality of these models, as only general measures have currently been defined [147]. Without a CAD framework, these goals would be at least difficult, if not impossible.

6 Summary and Conclusion

We presented a CAD-integrated patient-specific vascular modeling pipeline for performing NURBS-based IGA. This work builds upon and seeks to improve the template-based vascular modeling pipeline developed by Zhang *et al.* [27], which introduced a state-of-the-art approach but did not incorporate the benefits of CAD infrastructure. In our approach, a CAD system was leveraged as it offers a framework that is robust, intuitive, easy to learn and highly customizable. Based on a review of many of the existing FEA procedures, we argued that benefits of the presented approach are quicker processing time and more accurate reconstruction of complex models.

We outlined the three main stages of our modeling pipeline: (1) image processing, (2) analysis suitable model generation, and (3) analysis. We detailed two example applications of our methodology, and made observations about model processing time and accuracy. Finally, we discussed the critical advantages of using a CAD-integrated approach in the patient-specific modeling pipeline, outlined some drawbacks in the current implementation, and presented ideas for future work.

ACKNOWLEDGEMENT Support from the Farish Stump Foundation, Portuguese CoLab grant no UTA06-894, and William & Ella Owens Medical Research Foundation grant no UTA17-000357 are gratefully acknowledged. The authors would also like to thank Zbigniew Starosolski and Ananth Annapragada at the Texas Children's Hospital, Raja Muthupillai at St. Luke's Hospital, and Andrea Gobin and Doris Taylor at Texas Heart Institute for providing the imaging data, and for their help with image processing.

CONFLICT OF INTEREST STATEMENT On behalf of all authors, the corresponding author states that there is no conflict of interest.

References

[1] C.A. Taylor, C.A. Figueroa, Patient-specific modeling of cardiovascular mechanics, Annu. Rev. Biomed. Eng. 11 (2009) 109–134. doi:10.1146/annurev.bioeng.10.061807.160521.

[2] C.A. Taylor, T.J.R. Hughes, C.K. Zarins, Finite Element Modeling of Three-Dimensional Pulsatile Flow in the Abdominal Aorta: Relevance to Atherosclerosis, Ann. Biomed. Eng. 26 (1998) 975–987. doi:10.1114/1.140.

[3] C.A. Taylor, T.J.R. Hughes, C.K. Zarins, Computational investigations in vascular disease, Comput Phys. 10 (1996) 224–232.

[4] L. Antiga, M. Piccinelli, L. Botti, B. Ene-Iordache, A. Remuzzi, D.A. Steinman, An image-based modeling framework for patient-specific computational hemodynamics, Med. Biol. Eng. Comput. 46 (2008) 1097–1112. doi:10.1007/s11517-008-0420-1.

[5] C.A. Taylor, D.A. Steinman, Image-Based Modeling of Blood Flow and Vessel Wall Dynamics: Applications, Methods and Future Directions, Ann. Biomed. Eng. 38 (2010) 1188–1203. doi:10.1007/s10439-010-9901-0.

[6] H.J. Kim, I.E. Vignon-Clementel, J.S. Coogan, C.A. Figueroa, K.E. Jansen, C.A. Taylor, Patient-Specific Modeling of Blood Flow and Pressure in Human Coronary Arteries, Ann. Biomed. Eng. 38 (2010) 3195–3209. doi:10.1007/s10439-010-0083-6.

[7] C.A. Taylor, T.A. Fonte, J.K. Min, Computational Fluid Dynamics Applied to Cardiac Computed Tomography for Noninvasive Quantification of Fractional Flow ReserveScientific Basis, J. Am. Coll. Cardiol. 61 (2013) 2233–2241. doi:10.1016/j.jacc.2012.11.083.

[8] C. Rogers, D.Y. Tseng, J.C. Squire, E.R. Edelman, Balloon-Artery Interactions During Stent Placement, Circ. Res. 84 (1999) 378–383. doi:10.1161/01.RES.84.4.378.

[9] S. Sankaran, M.E. Moghadam, A.M. Kahn, E.E. Tseng, J.M. Guccione, A.L. m, Patient-Specific Multiscale Modeling of Blood Flow for Coronary Artery Bypass Graft Surgery, Ann. Biomed. Eng. 40 (2012) 2228–2242. doi:10.1007/s10439-012-0579-3.

[10] S.S. Hossain, S.F.A. Hossainy, Y. Bazilevs, V.M. Calo, T.J.R. Hughes, Mathematical modeling of coupled drug and drug-encapsulated nanoparticle transport in patient-specific coronary artery walls, Comput. Mech. 49 (2012) 213–242. doi:10.1007/s00466-011-0633-2.

[11] S.S. Hossain, T.J.R. Hughes, P. Decuzzi, Vascular Deposition Patterns for Catheter-Injected Nanoparticles in an Inflamed Patient-specific Arterial Tree, Biomech. Model. Mechanobiol. (2013). doi:10.1007/s10237-013-0520-1.

[12] S.S. Hossain, Y. Zhang, X. Fu, G. Brunner, J. Singh, T.J.R. Hughes, D. Shah, P. Decuzzi, Magnetic resonance imaging-based computational modelling of blood flow and nanomedicine deposition in patients with peripheral arterial disease, J. R. Soc. Interface. 12 (2015) 20150001. doi:10.1098/rsif.2015.0001.

[13] S.S. Hossain, Y. Zhang, X. Liang, F. Hussain, M. Ferrari, T.J. Hughes, P. Decuzzi, In silico vascular modeling for personalized nanoparticle delivery, Nanomed. (2012). doi:10.2217/nnm.12.124.

[14] J.R. Cebral, S. Hendrickson, C.M. Putman, Hemodynamics in a Lethal Basilar Artery Aneurysm Just before Its Rupture, Am. J. Neuroradiol. 30 (2009) 95–98. doi:10.3174/ajnr.A1312.

[15] B.J. Doyle, A. Callanan, P.E. Burke, P.A. Grace, M.T. Walsh, D.A. Vorp, T.M. McGloughlin, Vessel asymmetry as an additional diagnostic tool in the assessment of abdominal aortic aneurysms, J. Vasc. Surg. 49 (2009) 443–454. doi:10.1016/j.jvs.2008.08.064.

[16] M.L. Neal, R. Kerckhoffs, Current progress in patient-specific modeling, Brief. Bioinform. 11 (2010) 111–126. doi:10.1093/bib/bbp049.

[17] M. Kim, D.B. Taulbee, M. Tremmel, H. Meng, Comparison of Two Stents in Modifying Cerebral Aneurysm Hemodynamics, Ann. Biomed. Eng. 36 (2008) 726–741. doi:10.1007/s10439-008-9449-4.

[18] V.L. Rayz, L. Boussel, M.T. Lawton, G. Acevedo-Bolton, L. Ge, W.L. Young, R.T. Higashida, D. Saloner, Numerical modeling of the flow in intracranial aneurysms: prediction of regions prone to thrombus formation, Ann. Biomed. Eng. 36 (2008) 1793–1804. doi:10.1007/s10439-008-9561-5.

[19] F.P.P. Tan, G. Soloperto, S. Bashford, N.B. Wood, S. Thom, A. Hughes, X.Y. Xu, Analysis of flow disturbance in a stenosed carotid artery bifurcation using two-equation transitional and turbulence models, J. Biomech. Eng. 130 (2008) 061008. doi:10.1115/1.2978992.
[20] C.A. Taylor, T.J.R. Hughes, C.K. Zarins, Finite element modeling of blood flow in arteries, Comput. Methods Appl. Mech. Eng. 158 (1998) 155–196. doi:10.1016/S0045-7825(98)80008-X.
[21] A. Valencia, H. Morales, R. Rivera, E. Bravo, M. Galvez, Blood flow dynamics in patient-specific cerebral aneurysm models: the relationship between wall shear stress and aneurysm area index, Med Eng Phys. 30 (2008) 329–40. doi:10.1016/j.medengphy.2007.04.011.
[22] V. Mihalef, R.I. Ionasec, P. Sharma, B. Georgescu, I. Voigt, M. Suehling, D. Comaniciu, Patient-specific modelling of whole heart anatomy, dynamics and haemodynamics from four-dimensional cardiac CT images, Interface Focus. 1 (2011) 286–296.
[23] T.J.R. Hughes, J.A. Cottrell, Y. Bazilevs, Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement, Comput. Methods Appl. Mech. Eng. 194 (2005) 4135–4195. doi:10.1016/j.cma.2004.10.008.
[24] Y. Bazilevs, V.M. Calo, Y. Zhang, T.J.R. Hughes, Isogeometric Fluid–structure Interaction Analysis with Applications to Arterial Blood Flow, Comput. Mech. 38 (2006) 310–322. doi:10.1007/s00466-006-0084-3.
[25] Y. Zhang, Y. Bazilevs, S. Goswami, C.L. Bajaj, T.J.R. Hughes, Patient-specific vascular NURBS modeling for isogeometric analysis of blood flow, Comput. Methods Appl. Mech. Eng. 196 (2007) 2943–2959. doi:10.1016/j.cma.2007.02.009.
[26] Y. Bazilevs, J.R. Gohean, T.J.R. Hughes, R.D. Moser, Y. Zhang, Patient-specific isogeometric fluid–structure interaction analysis of thoracic aortic blood flow due to implantation of the Jarvik 2000 left ventricular assist device, Comput. Methods Appl. Mech. Eng. 198 (2009) 3534–3550. doi:10.1016/j.cma.2009.04.015.
[27] S.S. Hossain, Mathematical Modeling of Coupled Drug and Drug-encapsulated Nanoparticle Transport in Patient-specific Coronary Artery Walls, Dissertation, University of Texas at Austin, 2009.
[28] F. Auricchio, M. Conti, M. Ferraro, S. Morganti, A. Reali, R.L. Taylor, Innovative and efficient stent flexibility simulations based on isogeometric analysis, Comput. Methods Appl. Mech. Eng. 295 (2015) 347–361. doi:10.1016/j.cma.2015.07.011.
[29] S. Morganti, F. Auricchio, D.J. Benson, F.I. Gambarin, S. Hartmann, T.J.R. Hughes, A. Reali, Patient-specific isogeometric structural analysis of aortic valve closure, Comput. Methods Appl. Mech. Eng. 284 (2015) 508–520. doi:10.1016/j.cma.2014.10.010.
[30] J.A. Cottrell, T.J.R. Hughes, Y. Bazilevs, Isogeometric Analysis : Toward Integration of CAD and FEA, 1st ed., Wiley, Hoboken, 2009.
[31] Autodesk AutoCAD, (n.d.). http://www.autodesk.com/products/autocad/overview (accessed April 21, 2016).
[32] SOLIDWORKS, (n.d.). http://www.solidworks.com/ (accessed April 21, 2016).
[33] Rhinoceros, (n.d.). https://www.rhino3d.com/ (accessed April 21, 2016).
[34] CATIA, (n.d.). http://www.3ds.com/products-services/catia/ (accessed April 21, 2016).
[35] L.A. Piegl, W. Tiller, The NURBS book, 2nd ed, Springer, Berlin ; New York, 1997.
[36] S. Prakash, C.R. Ethier, Requirements for Mesh Resolution in 3D Computational Hemodynamics, J. Biomech. Eng. 123 (2000) 134–144. doi:10.1115/1.1351807.
[37] S. Sankaran, L. Grady, C.A. Taylor, Impact of geometric uncertainty on hemodynamic simulations using machine learning, Comput. Methods Appl. Mech. Eng. 297 (2015) 167–190. doi:10.1016/j.cma.2015.08.014.
[38] Y. Lai, L. Liu, Y.J. Zhang, J. Chen, E. Fang, J. Lua, Rhino 3D to Abaqus: A T-spline Based Isogeometric Analysis Software Platform, in: Ed. Vol. Model. Simul. Sci. Eng. Technol. Book Ser. Devoted AFSI 2014 - Birthd. Celebr. Conf. Tayfun Tezduyar, Springer Publisher, 2015.
[39] Y. Lai, Y.J. Zhang, L. Liu, X. Wei, E. Fang, J. Lua, Integrating CAD with Abaqus: A practical isogeometric analysis software platform for industrial applications, Comput. Math. Appl. (n.d.). doi:10.1016/j.camwa.2017.03.032.
[40] E. Kuhl, R. Maas, G. Himpel, A. Menzel, Computational modeling of arterial wall growth, Biomech. Model. Mechanobiol. 6 (2006) 321–331. doi:10.1007/s10237-006-0062-x.

[41] M.A. Castro, C.M. Putman, J.R. Cebral, Patient-Specific Computational Modeling of Cerebral Aneurysms With Multiple Avenues of Flow From 3D Rotational Angiography Images, Acad. Radiol. 13 (2006) 811–821. doi:10.1016/j.acra.2006.03.011.
[42] J.R. Cebral, F. Mut, D. Sforza, R. Löhner, E. Scrivano, P. Lylyk, C. Putman, Clinical application of image-based CFD for cerebral aneurysms, Int. J. Numer. Methods Biomed. Eng. 27 (2011) 977–992. doi:10.1002/cnm.1373.
[43] Q. Huang, J. Xu, J. Cheng, S. Wang, K. Wang, J.-M. Liu, Hemodynamic Changes by Flow Diverters in Rabbit Aneurysm Models A Computational Fluid Dynamic Study Based on Micro–Computed Tomography Reconstruction, Stroke. 44 (2013) 1936–1941. doi:10.1161/STROKEAHA.113.001202.
[44] C. Karmonik, J. Bismuth, M.G. Davies, D.J. Shah, H.K. Younes, A.B. Lumsden, A Computational Fluid Dynamics Study Pre- and Post-Stent Graft Placement in an Acute Type B Aortic Dissection, Vasc. Endovascular Surg. 45 (2011) 157–164. doi:10.1177/1538574410389342.
[45] G.-Y. Suh, A.S. Les, A.S. Tenforde, S.C. Shadden, R.L. Spilker, J.J. Yeung, C.P. Cheng, R.J. Herfkens, R.L. Dalman, C.A. Taylor, Quantification of Particle Residence Time in Abdominal Aortic Aneurysms Using Magnetic Resonance Imaging and Computational Fluid Dynamics, Ann. Biomed. Eng. 39 (2010) 864–883. doi:10.1007/s10439-010-0202-4.
[46] F. Graziano, V.M. Russo, W. Wang, D. Khismatullin, A.J. Ulm, 3D Computational Fluid Dynamics of a Treated Vertebrobasilar Giant Aneurysm: A Multistage Analysis, Am. J. Neuroradiol. 34 (2013) 1387–1394. doi:10.3174/ajnr.A3373.
[47] Y. He, C.M. Terry, C. Nguyen, S.A. Berceli, Y.-T.E. Shiu, A.K. Cheung, Serial analysis of lumen geometry and hemodynamics in human arteriovenous fistula for hemodialysis using magnetic resonance imaging and computational fluid dynamics, J. Biomech. 46 (2013) 165–169. doi:10.1016/j.jbiomech.2012.09.005.
[48] A.L. Marsden, A.J. Bernstein, V.M. Reddy, S.C. Shadden, R.L. Spilker, F.P. Chan, C.A. Taylor, J.A. Feinstein, Evaluation of a novel Y-shaped extracardiac Fontan baffle using computational fluid dynamics, J. Thorac. Cardiovasc. Surg. 137 (2009) 394–403.e2. doi:10.1016/j.jtcvs.2008.06.043.
[49] J. Kiendl, K.-U. Bletzinger, J. Linhard, R. Wüchner, Isogeometric shell analysis with Kirchhoff–Love elements, Comput. Methods Appl. Mech. Eng. 198 (2009) 3902–3914. doi:10.1016/j.cma.2009.08.013.
[50] M.-C. Hsu, I. Akkerman, Y. Bazilevs, High-performance computing of wind turbine aerodynamics using isogeometric analysis, Comput. Fluids. 49 (2011) 93–100. doi:10.1016/j.compfluid.2011.05.002.
[51] M.-C. Hsu, D. Kamensky, F. Xu, J. Kiendl, C. Wang, M.C.H. Wu, J. Mineroff, A. Reali, Y. Bazilevs, M.S. Sacks, Dynamic and fluid–structure interaction simulations of bioprosthetic heart valves using parametric design with T-splines and Fung-type material models, Comput. Mech. 55 (2015) 1211–1225. doi:10.1007/s00466-015-1166-x.
[52] Y. Bazilevs, I. Akkerman, Large eddy simulation of turbulent Taylor–Couette flow using isogeometric analysis and the residual-based variational multiscale method, J. Comput. Phys. 229 (2010) 3402–3414. doi:10.1016/j.jcp.2010.01.008.
[53] Isogeometric analysis in electromagnetics: B-splines approximation, (n.d.). http://www.sciencedirect.com/science/article/pii/S0045782509004010 (accessed October 18, 2016).
[54] E.L. Bove, M.R. de Leval, F. Migliavacca, G. Guadagni, G. Dubini, Computational fluid dynamics in the evaluation of hemodynamic performance of cavopulmonary connections after the norwood procedure for hypoplastic left heart syndrome, J. Thorac. Cardiovasc. Surg. 126 (2003) 1040–1047. doi:10.1016/S0022-5223(03)00698-6.
[55] M. Ghaffari, C.-Y. Hsu, A.A. Linninger, Automatic Reconstruction and Generation of Structured Hexahedral Mesh for Non-planar Bifurcations in Vascular Networks, in: J.K.H. and R.G. Krist V. Gernaey (Ed.), Comput. Aided Chem. Eng., Elsevier, 2015: pp. 635–640. http://www.sciencedirect.com/science/article/pii/B9780444635785501018 (accessed March 23, 2016).
[56] L. Antiga, B. Ene-Iordache, L. Caverni, G. Paolo Cornalba, A. Remuzzi, Geometric reconstruction for computational mesh generation of arterial bifurcations from CT angiography, Comput. Med. Imaging Graph. 26 (2002) 227–235. doi:10.1016/S0895-6111(02)00020-4.

[57] D. Birchall, A. Zaman, J. Hacker, G. Davies, D. Mendelow, Analysis of haemodynamic disturbance in the atherosclerotic carotid artery using computational fluid dynamics, Eur. Radiol. 16 (2006) 1074–1083. doi:10.1007/s00330-005-0048-6.
[58] V.M. Calo, N.F. Brasher, Y. Bazilevs, T.J.R. Hughes, Multiphysics model for blood flow and drug transport with application to patient-specific coronary artery flow, Comput. Mech. 43 (2008) 161–177. doi:10.1007/s00466-008-0321-z.
[59] J.J. Shah, M. Mäntylä, Parametric and feature-based CAD/CAM: concepts, techniques, and applications, Wiley, New York, 1995.
[60] M.F. Hardwick, R.L. Clay, P.T. Boggs, E.J. Walsh, A.R. Larzelere, A. Altshuler, DART System Analysis, Sandia National Laboratories, Albuquerque, New Mexico 87185 and Livermore, California 94550, 2005.
[61] E. Cohen, T. Martin, R.M. Kirby, T. Lyche, R.F. Riesenfeld, Analysis-aware modeling: Understanding quality considerations in modeling for isogeometric analysis, Comput. Methods Appl. Mech. Eng. 199 (2010) 334–356. doi:10.1016/j.cma.2009.09.010.
[62] T.W. Sederberg, G.T. Finnigan, X. Li, H. Lin, H. Ipson, Watertight trimmed NURBS, ACM Trans Graph. 27 (2008) 79:1–79:8. doi:10.1145/1360612.1360678.
[63] M. Breitenberger, A. Apostolatos, B. Philipp, R. Wüchner, K.-U. Bletzinger, Analysis in computer aided design: Nonlinear isogeometric B-Rep analysis of shell structures, Comput. Methods Appl. Mech. Eng. 284 (2015) 401–457. doi:10.1016/j.cma.2014.09.033.
[64] K.A. Belibassakis, T.P. Gerostathis, K.V. Kostas, C.G. Politis, P.D. Kaklis, A.I. Ginnis, C. Feurer, A BEM-isogeometric method for the ship wave-resistance problem, Ocean Eng. 60 (2013) 53–67. doi:10.1016/j.oceaneng.2012.12.030.
[65] F. Auricchio, M. Conti, C. Ferrazzano, G.A. Sgueglia, A simple framework to generate 3D patient-specific model of coronary artery bifurcation from single-plane angiographic images, Comput. Biol. Med. 44 (2014) 97–109. doi:10.1016/j.compbiomed.2013.10.027.
[66] H. Bogunović, J.M. Pozo, M.C. Villa-Uriol, C.B.L.M. Majoie, R. van den Berg, H.A.F.G. van Andel, J.M. Macho, J. Blasco, L.S. Román, A.F. Frangi, Automated segmentation of cerebral vasculature with aneurysms in 3DRA and TOF-MRA using geodesic active regions: An evaluation study, Med. Phys. 38 (2011) 210–222. doi:10.1118/1.3515749.
[67] O. Brina, R. Ouared, O. Bonnefous, F. van Nijnatten, P. Bouillot, P. Bijlenga, K. Schaller, K.-O. Lovblad, T. Grünhagen, D. Ruijters, V.M. Pereira, Intra-Aneurysmal Flow Patterns: Illustrative Comparison among Digital Subtraction Angiography, Optical Flow, and Computational Fluid Dynamics, Am. J. Neuroradiol. 35 (2014) 2348–2353. doi:10.3174/ajnr.A4063.
[68] J.R. Cebral, M.A. Castro, S. Appanaboyina, C.M. Putman, D. Millan, A.F. Frangi, Efficient pipeline for image-based patient-specific analysis of cerebral aneurysm hemodynamics: technique and sensitivity, IEEE Trans. Med. Imaging. 24 (2005) 457–467.
[69] J.S. Coogan, F.P. Chan, C.A. Taylor, J.A. Feinstein, Computational fluid dynamic simulations of aortic coarctation comparing the effects of surgical- and stent-based treatments on aortic compliance and ventricular workload, Catheter. Cardiovasc. Interv. 77 (2011) 680–691. doi:10.1002/ccd.22878.
[70] H. Ho, S. Norris, K. Mithraratne, P. Hunter, 1D AND 3D BLOOD FLOW MODELLING FOR PATIENT SPECIFIC CEREBRAL VASCULATURE AND ANEURYSM, J. Biomech. 41, Supplement 1 (2008) S8. doi:10.1016/S0021-9290(08)70008-3.
[71] V.O. Kheyfets, L. Rios, T. Smith, T. Schroeder, J. Mueller, S. Murali, D. Lasorda, A. Zikos, J. Spotti, J.J. Reilly Jr., E.A. Finol, Patient-specific computational modeling of blood flow in the pulmonary arterial circulation, Comput. Methods Programs Biomed. 120 (2015) 88–101. doi:10.1016/j.cmpb.2015.04.005.
[72] A.S. Les, S.C. Shadden, C.A. Figueroa, J.M. Park, M.M. Tedesco, R.J. Herfkens, R.L. Dalman, C.A. Taylor, Quantification of Hemodynamics in Abdominal Aortic Aneurysms During Rest and Exercise Using Magnetic Resonance Imaging and Computational Fluid Dynamics, Ann. Biomed. Eng. 38 (2010) 1288–1313. doi:10.1007/s10439-010-9949-x.
[73] D. Ma, G.F. Dargush, S.K. Natarajan, E.I. Levy, A.H. Siddiqui, H. Meng, Computer modeling of deployment and mechanical expansion of neurovascular flow diverter in patient-specific intracranial aneurysms, J. Biomech. 45 (2012) 2256–2263. doi:10.1016/j.jbiomech.2012.06.013.

[74] A.L. Marsden, Optimization in Cardiovascular Modeling, Annu. Rev. Fluid Mech. 46 (2014) 519–546. doi:10.1146/annurev-fluid-010313-141341.
[75] F. Migliavacca, G. Dubini, Computational modeling of vascular anastomoses, Biomech. Model. Mechanobiol. 3 (2005) 235–250. doi:10.1007/s10237-005-0070-2.
[76] T. Moench, R. Gasteiger, G. Janiga, H. Theisel, B. Preim, Context-aware mesh smoothing for biomedical applications, Comput. Graph. 35 (2011) 755–767. doi:10.1016/j.cag.2011.04.011.
[77] J.A. Moore, B.K. Rutt, S.J. Karlik, K. Yin, C.R. Ethier, Computational Blood Flow Modeling Based on In Vivo Measurements, Ann. Biomed. Eng. 27 (1999) 627–640. doi:10.1114/1.221.
[78] R. Nagy, C. Csobay-Novák, A. Lovas, P. Sótonyi, I. Bojtár, Non-invasive in vivo time-dependent strain measurement method in human abdominal aortic aneurysms: Towards a novel approach to rupture risk estimation, J. Biomech. 48 (2015) 1876–1886. doi:10.1016/j.jbiomech.2015.04.030.
[79] S.-T. Park, K. Yoon, Y.B. Ko, D.C. Suh, Computational Fluid Dynamics of Intracranial and Extracranal Arteries using 3-Dimensional Angiography: Technical Considerations with Physician's Point of View, Neurointervention. 8 (2013) 92. doi:10.5469/neuroint.2013.8.2.92.
[80] J. Peiró, S.J. Sherwin, S. Giordana, Automatic reconstruction of a patient-specific high-order surface representation and its application to mesh generation for CFD calculations, Med. Biol. Eng. Comput. 46 (2008) 1069–1083. doi:10.1007/s11517-008-0390-3.
[81] F. Rengier, A. Mehndiratta, H. von Tengg-Kobligk, C.M. Zechmann, R. Unterhinninghofen, H.-U. Kauczor, F.L. Giesel, 3D printing based on imaging data: review of medical applications, Int. J. Comput. Assist. Radiol. Surg. 5 (2010) 335–341. doi:10.1007/s11548-010-0476-x.
[82] I. Sazonov, S.Y. Yeo, R.L.T. Bevan, X. Xie, R. van Loon, P. Nithiarasu, Modelling pipeline for subject-specific arterial blood flow—A review, Int. J. Numer. Methods Biomed. Eng. 27 (2011) 1868–1910. doi:10.1002/cnm.1446.
[83] A. Sheidaei, S.C. Hunley, S. Zeinali-Davarani, L.G. Raguin, S. Baek, Simulation of abdominal aortic aneurysm growth with updating hemodynamic loads using a realistic geometry, Med. Eng. Phys. 33 (2011) 80–88. doi:10.1016/j.medengphy.2010.09.012.
[84] A. Wittek, N.M. Grosland, G.R. Joldes, V. Magnotta, K. Miller, From Finite Element Meshes to Clouds of Points: A Review of Methods for Generation of Computational Biomechanics Models for Patient-Specific Applications, Ann. Biomed. Eng. 44 (2015) 3–15. doi:10.1007/s10439-015-1469-2.
[85] K. Pekkan, B. Whited, K. Kanter, S. Sharma, D. de Zelicourt, K. Sundareswaran, D. Frakes, J. Rossignac, A.P. Yoganathan, Patient-specific surgical planning and hemodynamic computational fluid dynamics optimization through free-form haptic anatomy editing tool (SURGEM), Med. Biol. Eng. Comput. 46 (2008) 1139–1152. doi:10.1007/s11517-008-0377-0.
[86] J.R. Rossignac, K. Pekkan, B. Whited, K. Kanter, S. Sharma, A.P. Yoganathan, Surgem: Next Generation CAD Tools for Interactive Patient-Specific Surgical Planning and Hemodynamic Analysis, 2006. https://smartech.gatech.edu/handle/1853/13133 (accessed April 17, 2016).
[87] A. Lopez-Perez, R. Sebastian, J.M. Ferrero, Three-dimensional cardiac computational modelling: methods, features and applications, Biomed. Eng. OnLine. 14 (2015) 1–31. doi:10.1186/s12938-015-0033-5.
[88] I. Llamas, A. Powell, J.R. Rossignac, C.D. Shaw, Bender: A Virtual Ribbon for Deforming 3D Shapes in Biomedical and Styling Applications, 2004. https://smartech.gatech.edu/handle/1853/3734 (accessed April 21, 2016).
[89] SOLIDWORKS Solutions for Life Sciences Professionals, (n.d.). http://www.solidworks.com/sw/industries/life-sciences-overview-industries.htm (accessed April 19, 2016).
[90] Research with Geomagic Sensable, (n.d.). http://www.geomagic.com/en/industries/research (accessed April 19, 2016).
[91] Materialise 3-matic, (n.d.). http://biomedical.materialise.com/3-matic (accessed April 19, 2016).
[92] BioCAD, (n.d.). http://www.biomodel.com/biocad.html (accessed April 19, 2016).
[93] Zygote, (n.d.). https://www.zygote.com/ (accessed April 19, 2016).
[94] Zygote: Featured Solid 3D CAD Models, (n.d.). https://www.zygote.com/cad-models (accessed April 19, 2016).

[95] N. Golovanov, Geometric Modeling: The mathematics of shapes, Reprint edition, CreateSpace Independent Publishing Platform, 2014.
[96] starlab-mcfskel, (n.d.). https://github.com/ataiya/starlab-mcfskel (accessed August 28, 2015).
[97] vmtk - the Vascular Modelling Toolkit, (n.d.). http://www.vmtk.org/ (accessed April 20, 2016).
[98] Materialise Mimics Innovation Suite, (n.d.). http://biomedical.materialise.com/mis (accessed May 5, 2016).
[99] Grasshopper - algorithmic modeling for Rhino, (n.d.). http://www.grasshopper3d.com/ (accessed June 26, 2017).
[100] Z. Starosolski, C.A. Villamizar, D. Rendon, M.J. Paldino, D.M. Milewicz, K.B. Ghaghada, A.V. Annapragada, Ultra High-Resolution In vivo Computed Tomography Imaging of Mouse Cerebrovasculature Using a Long Circulating Blood Pool Contrast Agent, Sci Rep. 5 (2015). doi:10.1038/srep10178.
[101] A. Fedorov, R. Beichel, J. Kalpathy-Cramer, J. Finet, J.-C. Fillion-Robin, S. Pujol, C. Bauer, D. Jennings, F. Fennessy, M. Sonka, J. Buatti, S. Aylward, J.V. Miller, S. Pieper, R. Kikinis, 3D Slicer as an image computing platform for the Quantitative Imaging Network, Magn. Reson. Imaging. 30 (2012) 1323–1341. doi:10.1016/j.mri.2012.05.001.
[102] V. Caselles, R. Kimmel, G. Sapiro, Geodesic Active Contours, Int. J. Comput. Vis. 22 (1997) 61–79. doi:10.1023/A:1007979827043.
[103] E. Parzy, S. Miraux, F. Jean-Michel, E. Thiaudière, In vivo quantification of blood velocity in mouse carotid and pulmonary arteries by ECG-triggered 3D time-resolved magnetic resonance angiography, NMR Biomed. 22 (2009) 532–537.
[104] D.E. LaCourse, ed., Handbook of solid modeling, McGraw-Hill, New York, 1995.
[105] J. Corney, T. Lim, 3D modeling with ACIS, 2nd ed., Saxe-Coburg, Stirling, 2001.
[106] VTK: Class List, (n.d.). http://www.vtk.org/doc/release/7.0/html/annotated.html (accessed April 20, 2016).
[107] I. Larrabide, M.-C. Villa-Uriol, R. Cárdenes, V. Barbarito, L. Carotenuto, A.J. Geers, H.G. Morales, J.M. Pozo, M.D. Mazzeo, H. Bogunović, P. Omedas, C. Riccobene, J.M. Macho, A.F. Frangi, AngioLab—A software tool for morphological analysis and endovascular treatment planning of intracranial aneurysms, Comput. Methods Programs Biomed. 108 (2012) 806–819. doi:10.1016/j.cmpb.2012.05.006.
[108] Russian National 3D Kernel, Isicad.Net. (n.d.). http://isicad.net/articles.php?article_num=15189 (accessed April 21, 2016).
[109] L. Antiga, M. Piccinelli, L. Botti, B. Ene-Iordache, A. Remuzzi, D.A. Steinman, An image-based modeling framework for patient-specific computational hemodynamics, Med. Biol. Eng. Comput. 46 (2008) 1097–1112. doi:10.1007/s11517-008-0420-1.
[110] D. Tang, C. Yang, J. Zheng, P.K. Woodard, G.A. Sicard, J.E. Saffitz, C. Yuan, 3D MRI-Based Multicomponent FSI Models for Atherosclerotic Plaques, Ann. Biomed. Eng. 32 (2004) 947–960. doi:10.1023/B:ABME.0000032457.10191.e0.
[111] S. Moore, T. David, J.G. Chase, J. Arnold, J. Fink, 3D models of blood flow in the cerebral vasculature, J. Biomech. 39 (2006) 1454–1463. doi:10.1016/j.jbiomech.2005.04.005.
[112] M.R. Kaazempur-Mofrad, A.G. Isasi, H.F. Younis, R.C. Chan, D.P. Hinton, G. Sukhova, G.M. LaMuraglia, R.T. Lee, R.D. Kamm, Characterization of the Atherosclerotic Carotid Bifurcation Using MRI, Finite Element Modeling, and Histology, Ann. Biomed. Eng. 32 (2004) 932–946. doi:10.1023/B:ABME.0000032456.16097.e0.
[113] J.R. Cebral, M.A. Castro, O. Soto, R. Löhner, N. Alperin, Blood-flow models of the circle of Willis from magnetic resonance data, J. Eng. Math. 47 (2003) 369–386. doi:10.1023/B:ENGI.0000007977.02652.02.
[114] J.B. Thomas, J.S. Milner, B.K. Rutt, D.A. Steinman, Reproducibility of Image-Based Computational Fluid Dynamics Models of the Human Carotid Bifurcation, Ann. Biomed. Eng. 31 (2003) 132–141. doi:10.1114/1.1540102.
[115] R. Torii, M. Oshima, T. Kobayashi, K. Takagi, T.E. Tezduyar, Fluid–structure interaction modeling of blood flow and cerebral aneurysm: Significance of artery and aneurysm shapes, Comput. Methods Appl. Mech. Eng. 198 (2009) 3613–3621. doi:10.1016/j.cma.2008.08.020.

[116] E. Boutsianis, H. Dave, T. Frauenfelder, D. Poulikakos, S. Wildermuth, M. Turina, Y. Ventikos, G. Zund, Computational simulation of intracoronary flow based on real coronary geometry, Eur. J. Cardiothorac. Surg. 26 (2004) 248–256. doi:10.1016/j.ejcts.2004.02.041.
[117] F. Auricchio, M. Conti, M. De Beule, G. De Santis, B. Verhegghe, Carotid artery stenting simulation: From patient-specific images to finite element analysis, Med. Eng. Phys. 33 (2011) 281–289. doi:10.1016/j.medengphy.2010.10.011.
[118] T. Frauenfelder, M. Lotfey, T. Boehm, S. Wildermuth, Computational Fluid Dynamics: Hemodynamic Changes in Abdominal Aortic Aneurysm After Stent-Graft Implantation, Cardiovasc. Intervent. Radiol. 29 (2006) 613–623. doi:10.1007/s00270-005-0227-5.
[119] C.A. Conti, A. Della Corte, E. Votta, L. Del Viscovo, C. Bancone, L.S. De Santo, A. Redaelli, Biomechanical implications of the congenital bicuspid aortic valve: A finite element study of aortic root function from in vivo data, J. Thorac. Cardiovasc. Surg. 140 (2010) 890–896.e2. doi:10.1016/j.jtcvs.2010.01.016.
[120] S. Appanaboyina, F. Mut, R. Löhner, C. Putman, J. Cebral, Simulation of intracranial aneurysm stenting: Techniques and challenges, Comput. Methods Appl. Mech. Eng. 198 (2009) 3567–3582. doi:10.1016/j.cma.2009.01.017.
[121] L. Grinberg, E. Cheever, T. Anor, J.R. Madsen, G.E. Karniadakis, Modeling Blood Flow Circulation in Intracranial Arterial Networks: A Comparative 3D/1D Simulation Study, Ann. Biomed. Eng. 39 (2010) 297–309. doi:10.1007/s10439-010-0132-1.
[122] K.M. Tse, P. Chiu, H.P. Lee, P. Ho, Investigation of hemodynamics in the development of dissecting aneurysm within patient-specific dissecting aneurismal aortas using computational fluid dynamics (CFD) simulations, J. Biomech. 44 (2011) 827–836. doi:10.1016/j.jbiomech.2010.12.014.
[123] S. Morlacchi, S.G. Colleoni, R. Cárdenes, C. Chiastra, J.L. Diez, I. Larrabide, F. Migliavacca, Patient-specific simulations of stenting procedures in coronary bifurcations: Two clinical cases, Med. Eng. Phys. 35 (2013) 1272–1281. doi:10.1016/j.medengphy.2013.01.007.
[124] G. De Santis, M. De Beule, K. Van Canneyt, P. Segers, P. Verdonck, B. Verhegghe, Full-hexahedral structured meshing for image-based computational vascular modeling, Med. Eng. Phys. 33 (2011) 1318–1325. doi:10.1016/j.medengphy.2011.06.007.
[125] Q. Wang, E. Sirois, W. Sun, Patient-specific modeling of biomechanical interaction in transcatheter aortic valve deployment, J. Biomech. 45 (2012) 1965–1971. doi:10.1016/j.jbiomech.2012.05.008.
[126] D. Gallo, G.D. Santis, F. Negri, D. Tresoldi, R. Ponzini, D. Massai, M.A. Deriu, P. Segers, B. Verhegghe, G. Rizzo, U. Morbiducci, On the Use of In Vivo Measured Flow Rates as Boundary Conditions for Image-Based Hemodynamic Models of the Human Aorta: Implications for Indicators of Abnormal Flow, Ann. Biomed. Eng. 40 (2011) 729–741. doi:10.1007/s10439-011-0431-1.
[127] Autodesk Alias, (n.d.). http://www.autodesk.com/products/alias-products/overview (accessed April 21, 2016).
[128] Autodesk Fusion 360, (n.d.). http://www.autodesk.com/products/fusion-360/overview (accessed April 21, 2016).
[129] Autodesk Inventor, (n.d.). http://www.autodesk.com/products/inventor/overview (accessed April 21, 2016).
[130] BRL-CAD: Open Source Solid Modeling, (n.d.). http://brlcad.org/ (accessed April 21, 2016).
[131] nanoCAD, (n.d.). http://nanocad.com/ (accessed April 21, 2016).
[132] NX: Siemens PLM Software, (n.d.). https://www.plm.automation.siemens.com/en_us/products/nx/ (accessed April 21, 2016).
[133] OPEN CASCADE, (n.d.). http://www.opencascade.com/ (accessed April 21, 2016).
[134] PTC Creo, (n.d.). http://www.ptc.com/cad/creo (accessed April 21, 2016).
[135] Solid Edge: Siemens PLM Software, (n.d.). https://www.plm.automation.siemens.com/en_us/products/solid-edge/ (accessed April 21, 2016).
[136] pyFormex: “Unlike traditional CAD systems, pyFormex uses a powerful (Python based) scripting language as the basic user input method, making it very well suited for automated and repeated (parametric) design procedures.,” (n.d.). http://www.nongnu.org/pyformex/ (accessed April 20, 2016).

[137] S. Prakash, C.R. Ethier, Requirements for mesh resolution in 3D computational hemodynamics, J. Biomech. Eng. 123 (2001) 134–144.
[138] M.A. Castro, M.C.A. Olivares, C.M. Putman, J.R. Cebral, Unsteady wall shear stress analysis from image-based computational fluid dynamic aneurysm models under Newtonian and Casson rheological models, Med. Biol. Eng. Comput. 52 (2014) 827–839. doi:10.1007/s11517-014-1189-z.
[139] R. Cárdenes, J.L. Díez, N. Duchateau, A. Pashaei, A.F. Frangi, Model generation of coronary artery bifurcations from CTA and single plane angiography, Med. Phys. 40 (2013) 013701. doi:10.1118/1.4769118.
[140] C.M. Haggerty, M. Restrepo, E. Tang, D.A. de Zélicourt, K.S. Sundareswaran, L. Mirabella, J. Bethel, K.K. Whitehead, M.A. Fogel, A.P. Yoganathan, Fontan hemodynamics from 100 patient-specific cardiac magnetic resonance studies: A computational fluid dynamics analysis, J. Thorac. Cardiovasc. Surg. 148 (2014) 1481–1489. doi:10.1016/j.jtcvs.2013.11.060.
[141] M.V. Caruso, V. Gramigna, M. Rossi, G.F. Serraino, A. Renzulli, G. Fragomeni, A computational fluid dynamics comparison between different outflow graft anastomosis locations of Left Ventricular Assist Device (LVAD) in a patient-specific aortic model, Int. J. Numer. Methods Biomed. Eng. 31 (2015) n/a-n/a. doi:10.1002/cnm.2700.
[142] C.G. DeGroff, Modeling the Fontan Circulation: Where We Are and Where We Need to Go, Pediatr. Cardiol. 29 (2007) 3–12. doi:10.1007/s00246-007-9104-0.
[143] D.C. Barber, E. Oubel, A.F. Frangi, D.R. Hose, Efficient computational fluid dynamics mesh generation by image registration, Med. Image Anal. 11 (2007) 648–662. doi:10.1016/j.media.2007.06.011.
[144] Y. Zhang, Y. Bazilevs, S. Goswami, C.L. Bajaj, T.J.R. Hughes, Patient-specific vascular NURBS modeling for isogeometric analysis of blood flow, Comput. Methods Appl. Mech. Eng. 196 (2007) 2943–2959. doi:10.1016/j.cma.2007.02.009.
[145] M.A. Scott, M.J. Borden, C.V. Verhoosel, T.W. Sederberg, T.J.R. Hughes, Isogeometric finite element data structures based on Bézier extraction of T-splines, Int. J. Numer. Methods Eng. 88 (2011) 126–156. doi:10.1002/nme.3167.
[146] Deepesh Toshniwal, Hendrik Speleers, Rene R. Hiemstra, Thomas J. R. Hughes, Multi-Degree C^k Smooth Polar Splines: A Framework for Design and Analysis, (2016). https://www.ices.utexas.edu/media/reports/2016/1617.pdf.
[147] K.C. Wang, R.W. Dutton, C.A. Taylor, Improving geometric model construction for blood flow modeling, IEEE Eng. Med. Biol. Mag. 18 (1999) 33–39. doi:10.1109/51.805142.