Mass Transport and Shear Stress within the Carotid Artery Bifurcation: A Study of Flow Effects on Atherosclerosis

Riley Gorder

A thesis submitted in partial fulfillment
of the requirements for the degree of

Master of Science in Mechanical Engineering

University of Washington

2010

Program Authorized to Offer Degree: Mechanical Engineering

University of Washington
Graduate School

This is to certify that I have examined this copy of a master's thesis by

Riley Gorder

and have found that it is complete and satisfactory in all respects,
and that any and all revisions required by the final
examining committee have been made.

Committee Members:

____
Alberto Aliseda

____
James Riley

____
Nathan Sniadecki

Date: ______________________________

Signature______________________________

Date__________________________________

TABLE OF CONTENTS

Page

[1] This derivation is based on the heat transfer counterpart found in Bird et. al. [2]

LIST OF FIGURES

[2] Picture from http://www.bartleby.com/107/illus513.html

LIST OF TABLES

ACKNOWLEDGMENTS

The author wishes to express sincere appreciation to his advisor, Professor Alberto Aliseda, whose work ethic and energy was an inspiration. Thanks is also given to Dan Leotta for providing the patient scans used in this thesis and to Kirk Beach, Ph.D., M.D for the helpful conversations about the endothelium. Lastly, the author wishes to thank his labmates, whose help and support meant much more to him than they know.

Chapter 1
INTRODUCTION

Someone has a stroke every 40 seconds in the United States and every four minutes someone dies from one. Stroke is the third highest cause of death in the US, behind heart disease and cancer, and places a huge cost on the heath care system with an estimate of $74 billion in 2010 [6]. Gaining a better understanding of the causes of stroke has the potential to save lives and significantly reduce the burden on the health care system.

1.1 Types of Stroke

The National Institutes of Health define stroke as an interruption of the blood supply to any part of the brain [25]. There are two types of stroke, hemorrhagic and ischemic. Hemorrhagic stroke occurs when blood vessels in the brain become weak and rupture, causing loss of blood into the brain. Ischemic stroke occurs when blood flow is stopped due to a blood clot. The clot may be formed locally by a thrombosis (thrombotic stroke), a narrowing of the artery wall, or a clot upstream in the circulation may break off and occlude a smaller artery downstream near to or in the brain (embolic stroke). 87% of strokes are ischemic. A common origin for embolic blood clots is the carotid artery bifurcation. Atherosclerosis affects this region more frequently than anywhere else in the vasculature, except perhaps the coronary arteries. Atherosclerotic plaque ruptures and produces blocking in a smaller diameter vessel in ther cerebral circulation.

1.2 The Carotid Arteries

The carotid arteries are a paired set of arteries, one for each side of the body, that supply blood to the head and neck. The arterial structure consists of a common carotid artery (CCA), internal carotid artery (ICA), and external carotid artery (ECA). The CCA branches

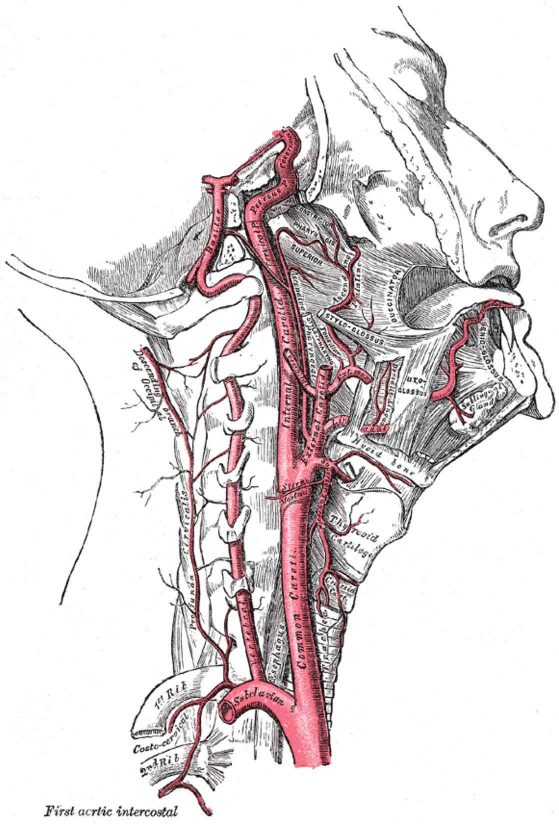


Figure 1.1: Sketch of carotid artery bifurcation and branches [1]

into the ICA and ECA at the carotid artery bifurcation, see Figure 1.2. The ICA has a localized dilation at its carotid artery bifurcation (CAB) origin. This dilation is known as the carotid sinus or bulb. The combination of bifurcation, curvature, pulsatility, and sudden change in arterial diameter causes the flow to be highly complex with recirculation regions and secondary flows. It is in areas of complex or disturbed flow that atherosclerosis is prominent in the vasculature.

1.3 Atherosclerosis

Atherosclerosis is the thickening and hardening of the arterial wall due to build up of fatty materials. It starts with the deposition of lipids into the arterial wall and progresses through a complex inflammatory response involving white blood cells (WBC) and smooth muscle cells (SMC). Low density lipoproteins (LDL) enter the intima through the endothelium. Macrophages and T-lymphocytes, types of white blood cells, enter to wall to *neutralize* the oxidized LDL. Macrophages and LDL combine to form foam cells, named as such for their large size and lipid content. Groups of foam cells form fatty streaks, a first indicator of atherosclerosis. SMCs migrate from the media and form a barrier between the plaque region and the arterial lumen. The SMCs form a fibrous cap around the lesion and, as they die, the cap hardens and becomes calcified. As the fibrous cap grows, the artery wall thickens and the lumen narrows. With time, the foam cells within the fibrous cap die from lack of nutrients and a necrotic core develops. The fibrous cap may rupture with time due to exposure of hemodynamic stresses or decreased cell health. This can lead to a thrombosis, a clotting of the artery at the rupture site, or an embolism, a clotting further downstream.

1.4 Literature Review

1.4.1 Flow Characterization

Early work completed by McDonald, Womersley and others [30] in the 1950s provided knowledge of pressure waveforms and provided evidence of flow reversal within the circulation of rabbits and dogs. Womersley, in his 1955 paper [30], introduced an analytical solution to the velocity profile of fully developed flow in a circular tube when the pressure gradient is pulsatile. The Womersley profile, as it is now known, is used extensively to generate suitable boundary conditions for use in computational simulations. Caro et al. [8] hypothesized that atherosclerosis is found in regions of low wall shear stress (WSS), a hypothesis that was later confirmed from in vitro examinations of CAB plaques. Experimental studies in blown glass models of the CAB were conducted in the 1980s by Ku, Giddens, Zarins and others [14, 15, 32]. These studies used averaged geometries measured from autopsy or bi-planar angiogram. Averaged geometries allow for general flow features common to most CABs to be identified and also for easier construction of experimental models. Steady and pulsatile inlet conditions were used, and comparisons of shear stress statistics to intimal wall thickness were made. The general flow found in the CAB was characterized by permanently separated flow in the carotid sinus region, with the detachment point moving through the cardiac cycle, and secondary flows caused by the bifurcation. Secondary flows can also be caused by curvature in the arterial geometry which early CAB studies did not include. High WSS was found at the apex of the bifurcation where the flow from the CCA impinges and forms secondary vorticies in the ECA and ICA. Low and oscillating WSS was found in the carotid sinus, which is caused by the recirculation and moving attachment and detachment points. Intimal thickening was found to be positively correlated with regions of low and oscillating shear, further reinforcing Caro's hypothesis. Ku [14] introduced the Oscillating Shear Index (OSI), a measure of the degree to which shear stress changes direction at a given point. It is still commonly used today. High values of OSI are found in connection with low values of WSS.

As computational resources grew and medical imaging techniques improved, use of computational fluid dynamics grew rapidly in cardiovascular simulations. Patient specific geometries and realistic pressure waveforms were studied by Milner and Steinman [20] using geometries reconstructed from MRI imaging. The large variability of CAB geometry and thus flow characteristics was studied by Lee et al. [16] and large variations in regions with low WSS and high OSI were found. Various geometry factors were investigated for correlations to low WSS and OSI. Early averaged geometry studies were limited to small parameter spaces and focused on bifurcation angle, planarity, and ICA:CCA cross sectional area ratio. In his work, Lee suggested using the proximal area ratio, the ratio between the carotid sinus and the CCA diameters, as it achieved a higher correlation to low WSS and high OSI than the geometry factors used in earlier bifurcation studies. Another strength in Lee's study was the parameter space investigated was obtained from patient specific, not idealized, averaged geometries.

Early computational mass transfer studies focused on idealized geometries like those found the early CAB experimental studies. Two dimensional models were initially used to minimize the computational, as the low diffusion coefficients typical for chemical species in the blood lead to sharp gradients where mass transfer occurs [11]. These models can only capture the detachment found at the carotid sinus and not the secondary flows caused by the bifurcation. Early 3D models used symmetry conditions, only solving half the geometry, and used steady inlet conditions. The study by Ma et al. [24] found a thickened mass boundary layer in the carotid sinus, caused by separation in the region. The location of minimum mass transfer did not match that of minimum shear stress, although both were highly correlated with intimal thickness.

1.4.2 Cell Response

The endothelial cell response to hemodynamic forces plays an important role in understanding atherosclerosis. The effect of shear stress on endothelial cell function is still not fully understood. Work by Levesque and Nerem [19] found that endothelial cells are highly aligned and elongated when exposed to directional shear stress. It has been shown that in areas of recirculation, or low and oscillating shear stress, endothelial cells lose their alignment and organization [9]. This has been found to lead to higher cell apoptosis. Tricot [28] measured a seven fold increase in cell apoptosis in the downstream section of an arterial narrowing by atherosclerotic plaque. Endothelial cells in "disturbed" flow regions, such as those in the CAB, lack organization of the cytoskeleton and intercellular junctions leading to increased macromolecular permeability [9]. This can lead to increased mass transfer of LDL into the arterial wall.

Reactive oxygen species (ROS) are another key factor in atherosclerosis. Nitric oxide (NO) is involved in vascular tone, regulation of blood pressure, inhibits adhesion of platelets and WBCs, and other protective responses towards atherosclerosis [22, 29]. NO also oxidizes LDL which allows the LDL to enter the intima. NO is produced in part by endothelial cells. In a review by Boo [7], two mechanisms for NO production by the endothelial cells are introduced. A burst production is found in sharp increases in shear stress, and there is a low sustained production for constant shear stress. This suggests that regions of blood flow that are separated throughout the cardiac cycle, such as flow in the carotid sinus, have a deficiency of NO. The role of NO and other ROS is still not fully understood. Clearly the mass transfer of species such as ROS, LDL, and WBC play a key role in atherosclerosis, but more quantitative relations to flow statistics are required.

1.5 Objectives of this Thesis

The main objectives of this thesis are listed below:

- Develop methodology to reconstruct patient-specific arterial geometries.
- Develop numerical methods to resolve the steep mass transfer at the arterial walls.
- Compare results from shear stress and scalar transport and explore potential links to atherosclerosis.

Chapter 2
GEOMETRY RECONSTRUCTION

Reconstruction of arterial geometry from medical images must be completed before CFD simulations or flow phantom studies can be conducted. An overview of geometry construction methods are given in this chapter and the specific workflow used in this work is described in detail.

2.1 Methods

Generating 3D geometries from medical images requires three basic steps: data acquisition, image segmentation, and reconstruction of the 3D geometry. The data acquisition step consists of obtaining medical images from the various technologies available. Ultrasound (US) was the imaging technology used in this thesis.

The last two steps are implemented in a way that is dependent on the method used for construction of the 3D geometry. Two main methods are in use: lofting and deformable parametric models. Lofted geometries are created from segmented 2D cross section contours by sweeping a surface through guide splines where tangency conditions are met [20]. The guide splines are longitidinal contours defining the arterial lumen extracted from the medical image data. This can be accomplished through forming b-splines from operator-chosen points [13, 20] or through active contours called *snakes*[31]. Snakes are dynamic curves that can move and change shape according to specified forces defined from the snake itself (typically curvature or elasticity) or from the medical image data (gradient of a potential function). This dynamic nature allows for an automated segmentation of the lumen geometry. Deformable parametric models allow for the geometry process to be nearly fully automated. The 3D parametric models are akin to the 2D counterpart, snakes, and are often called balloons. Although the process is almost fully automated, the use of 3D parametric models require a much longer development time for creating the geometry creation tool, whereas lofting can be implemented using commercial products such as SolidWorks with little upfront development time. For this reason lofting was the chosen method for the work presented in this thesis.

2.2 Pre-Processing the Medical Imaging Data

The geometries created for this work used data from the Center for Industrial and Medical Ultrasound (CIMU) at the Applied Physics Lab (UW). Patient scans were obtained by Dr. Dan Leotta and collaborators at CIMU with a magnetic 3D registration [18, 17]. The data received contained three dimensional coordinates for discrete points that delineate the segmented contours of the arterial lumen. This data was imported into MATLAB to pre-process before being exported into SolidWorks for solid model generation. The workflow of the MATLAB pre-processing is outlined below. Appendix A contains the scripts used. The procedure is described in more detail below.

1. Import 3D coordinates

2. Calculate centroids of the CCA contours

3. Translate data so the first CCA contour is centered at the origin

4. Estimate the centerline of the CCA

5. Rotate data so that the CCA centerline is aligned with the x-axis of the coordinate system

6. Calculate hydraulic diameter of CCA inlet

7. Create Flow extension contour

8. Export new coordinates for the lumen cross section to SolidWorks

2.2.1 Centerline Estimation

In order to align the CCA with the x-axis, an estimate of the CCA centerline must be made. This was accomplished using cubic spline interpolation, which produces a smooth (in first and second derivatives sense) and continuous curve. MATLAB has pre-built functions for implementing cubic splines, but the general framework is described here:

Cubic splines are piecewise polynomials with the form [3]

$$S_i = a_i + b_i(\hat{x} - x_i) + c_i(\hat{x} - x_i)^2 + d_i(\hat{x} - x_i)^3 \qquad \text{for } x \in [x_i, x_{i+1}] \tag{2.1}$$

For a spline with n centroids there are $n-1$ cubic polynomials that have a total of $4n-4$ unknowns. Continuity constraints account for $2n-2$ unknowns:

$$S_i(x_i) = a_i \tag{2.2a}$$

$$S_i(x_{i+1}) = S_{i+1}(x_{i+1}) = a_{i+1} \tag{2.2b}$$

Smoothness in the first and second derivatives for the interior curves acount for another $2n-4$ unknowns:

$$S_i'(x_{i+1}) = S_{i+1}'(x_{i+1}) \tag{2.3a}$$

$$S_i''(x_{i+1}) = S_{i+1}''(x_{i+1}) \tag{2.3b}$$

The last two constraints are handled in MATLAB with the “Not-a-Knot” constraint. This constraint is applied at the first two or last two cubic curves and matches the third derivatives of the paired polynomials:

$$S_1'''(x_2) = S_2'''(x_2) \tag{2.4a}$$

$$S_{n-2}'''(x_{n-2}) = S_{n-1}'''(x_{n-2}) \tag{2.4b}$$

2.2.2 Rotation Matrix

With the estimate of the CCA centerline calculated, the geometry data can be rotated to align the CCA with the x-axis. In order to rotate the geometry data a rotation matrix must be constructed. Equation 2.5 is a generalized rotation matrix

$$R = \begin{bmatrix} u_1^2 + (1 - u_1^2)c & u_1u_2(1-c) - u_3s & u_1u_3(1-c) + u_2s \\ u_1u_2(1-c) + u_3s & u_2^2 + (1 - u_2^2)c & u_2u_3(1-c) + u_1s \\ u_1u_3(1-c) - u_2s & u_2u_3(1-c) + u_1s & u_3^2 + (1 - u_3^2)c \end{bmatrix} \tag{2.5}$$

where

$$c = cos\theta \tag{2.6a}$$

$$s = sin\theta \tag{2.6b}$$

and $\mathbf{u} = (u_1, u_2, u_3)$ is the axis of rotation and θ is the angle of rotation. In the case of aligning the CCA with the x-axis $\mathbf{u}$ is determined by the cross product between the slope of the centerline estimate at the first CCA contour, $\overrightarrow{n}$, and the unit vector along the x-axis, $\overrightarrow{x}$. The angle is determined by using the four-quadrant inverse tangent, *atan2* function. This allows for accurate calculation of the angle of rotation even when the two vectors are nearly parallel.

2.2.3 Hydraulic Diameter and Flow Extensions

The coordinates of the points from the original segmented images were defined on a Cartesian coordinate system far away from the origin. The contours were translated and rotated to have the first contour of the common carotid artery (CCA) centered on the origin and the centerlines of the geometry were estimated to allow for the CCA to be aligned with the x-axis. The hydraulic diameter of the CCA was calculated to create flow extension with a circular inlet. This allows for an easier description of the inlet boundary conditions and also allows the flow to develop to the patient specific geometry. The hydraulic diameter is defined in Equation 2.7 and the MATLAB script used is found in Appendix A

$$D_h = \frac{4A}{P} \tag{2.7}$$

where A is the cross-sectional area and P is the wetted perimeter.

The flow extensions created are three to five hydraulic diameters in length. This length was chosen to allow for a smooth transition to the irregular cross-section of the arterial lumen during the lofting process. The flow extensions also allow the flow to develop to the patient-specific geometry. The lofting process was completed in SolidWorks and is discussed in the next section.

2.3 Solid Model Generation with SolidWorks

Once the pre-processing in MATLAB is completed, the data is exported to SolidWorks for reconstruction of the arterial geometry. The geometry data is imported into SolidWorks through a macro file generated by the MATLAB script *ultrasound2.mat* (found in Appendix A. The macro imports each arterial contour individually using the *Curve through XYZ Points* feature (see Figure 2.1). Next, the three arterial branches (CCA, ECA, and ICA) are formed by lofting surfaces through the curves (see Figure 2.2). Depending on the dataset, certain curves may be left out of the loft in order to reduce wiggle in the geometry that is not anatomically correct. This wiggle is caused by errors in the original data acquisition and is attributed to slight movements by the patient during the scan.

With the arterial branches lofted, the task of reconstructing the bifurcation region is started. This is the area that represents the highest level of uncertainty in the reconstruction. This is unfortunate as this is also the region of highest interest in terms of the hemodynamic factors. This uncertainty is due to the orientation of the medical scans. The scans are mainly perpendicular to the axial direction of the arterial geometry. While this provides a good description of the geometry in the arterial branches, the apex of the bifurcation has almost no detail. A few scans that are parallel to the axial directions at the bifurcation would

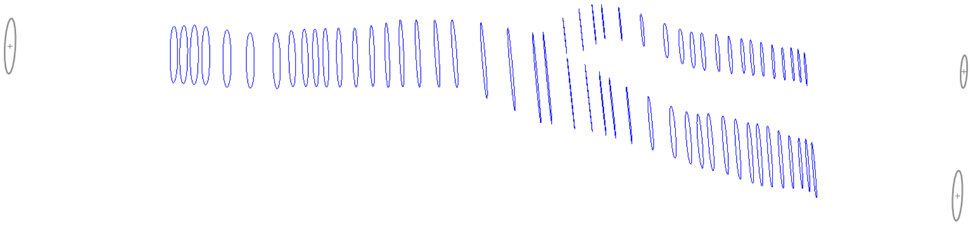


Figure 2.1: Contours imported into SolidWorks from MATLAB

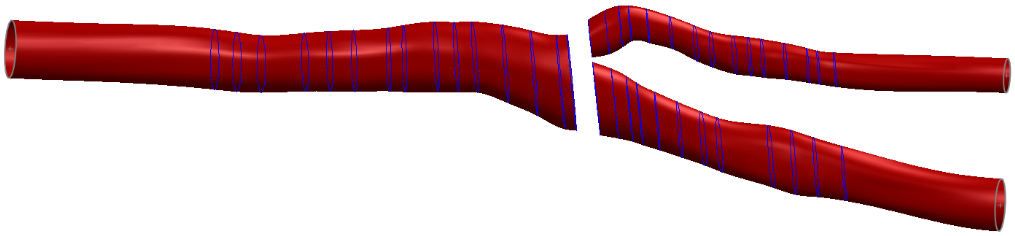


Figure 2.2: Lofted arterial branches after removing patient *wiggle*

provide the information necessary to have higher confidence in the reconstruction of the bifurcation region.

The medical images used for this thesis only included scans perpendicular to the axial flow. To form an estimate of the bifurcation region the following workflow was used:

1. Generate *face-curves* on the arterial branches parallel to the axial direction of the flow. See Figure 2.3

2. Create guide curves in the bifurcation region by connecting *face-curves* with 3D-curves matching tangency and, where possible, curvature. See Figure 2.4

3. Loft 3 surfaces connecting the 3 arterial branches. See Figure 2.5

    - CCA to ICA

    - CCA to ECA

- ICA to ECA (apex of bifurcation)

4. Finish geometry reconstruction by filling in remaining gaps in geometry using *surface-fill* feature. See Figures 2.6 and 2.7

The highest uncertainty in the workflow comes from the generation of the guide curves used to create the bifurcation apex. These curves connect *face-curves* of the ICA and ECA branches. While tangency conditions can be applied at the connection of the *face-curves* and guide curves, a curvature condition would at times create unrealistic shapes. On account of this, the curvature of the guide curves are typically defined by hand and this has a large impact on the overall shape of the bifurcation apex. It is recommended for future work to include scan data in the plane of the bifurcation so there is less uncertainty in the geometry reconstruction of the bifurcation apex.

2.4 Future Work

The workflow described was designed to allow for quick geometry reconstruction while using off-the-self modeling tools (SolidWorks). The process typically takes one to three hours per geometry depending on the complexity of the bifurcation region. This time constraint quickly becomes a problem for studies requiring a large number of patient scans. It is recommended to move towards a fully automated process. Parametric models are a good candidate for this development. Similar models are currently being used by Steinman and Antiga [4, 16, 5] through their open-source Vascular Modeling Toolkit (http://www.vmtk.org/). Their process currently works directly with the DICOM images from medical imaging systems, which were not available for this work. Parametric models using already segmented images, such as the data used in this thesis, could be created using methods similar to work done by Cohen [10].

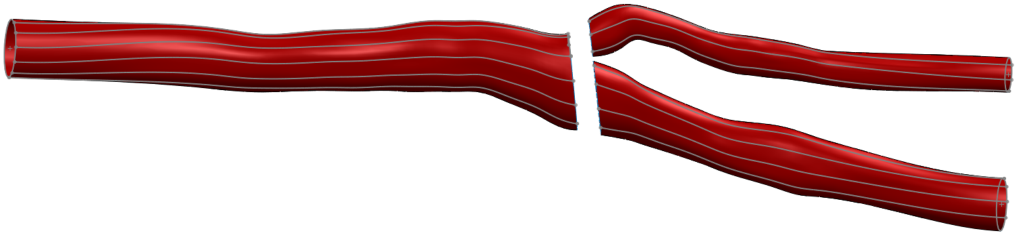


Figure 2.3: Generating face curves on arterial branches

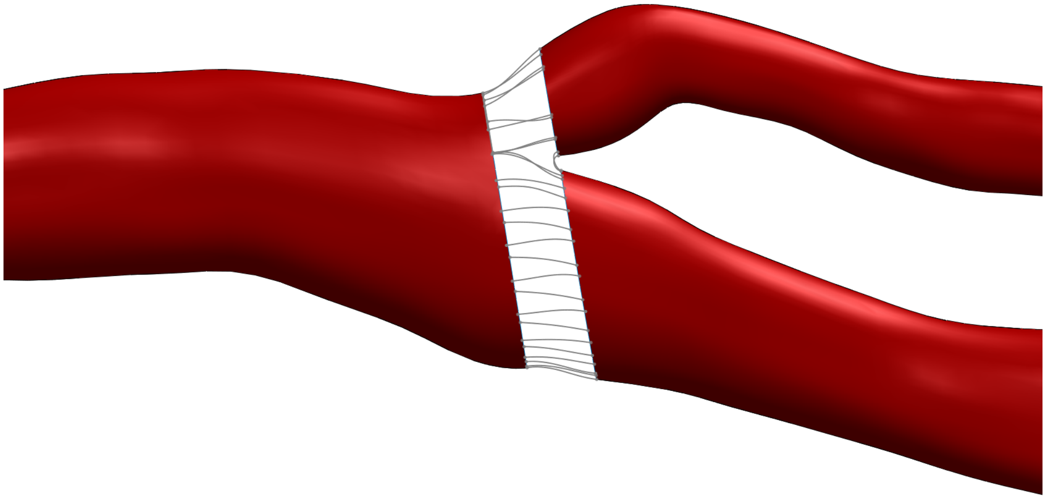


Figure 2.4: 3D guide curves used to connect arterial branches. The guide curves have tangency conditions with the face curves on the arterial branches

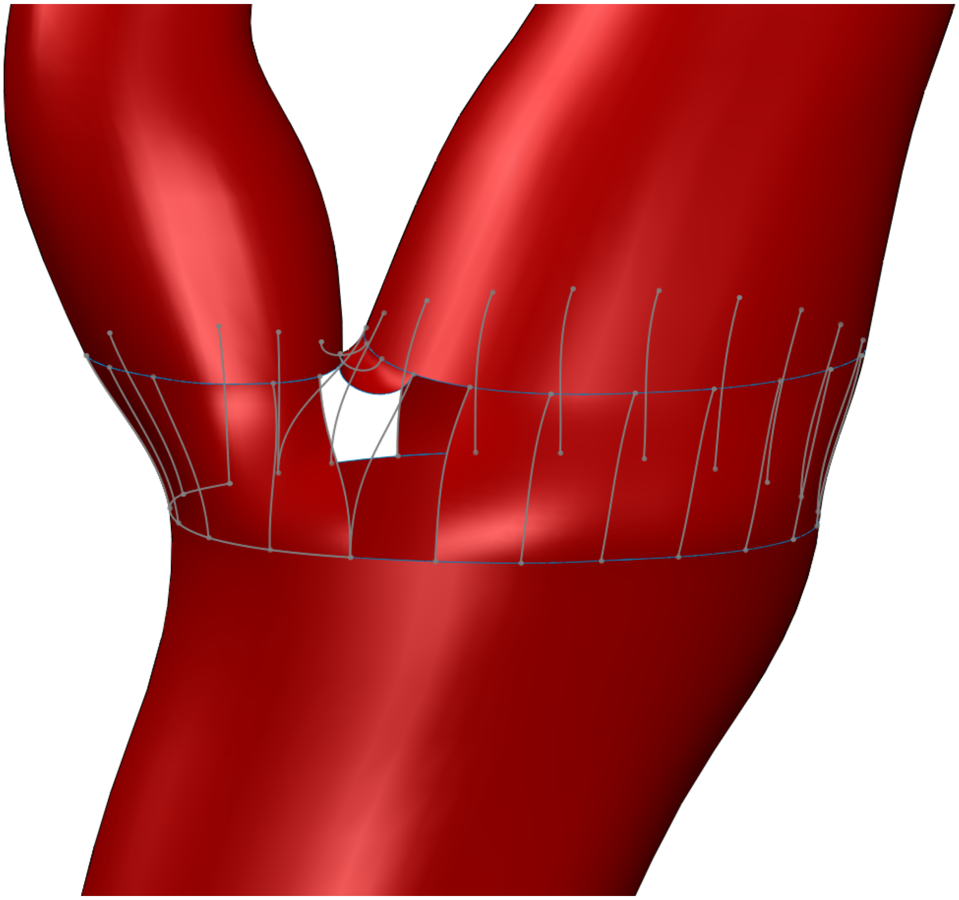


Figure 2.5: Surface Lofts connecting arterial branches

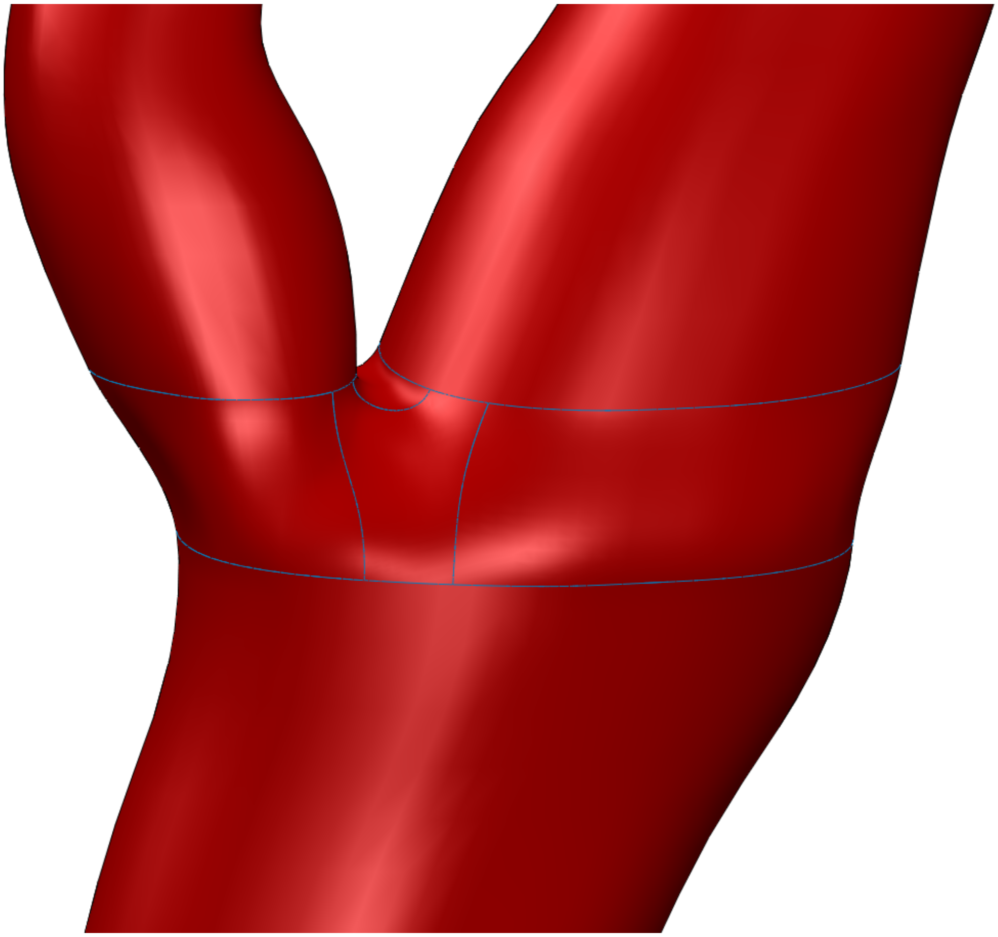


Figure 2.6: The *surface fill* feature fills the areas not lofted by the guide curves

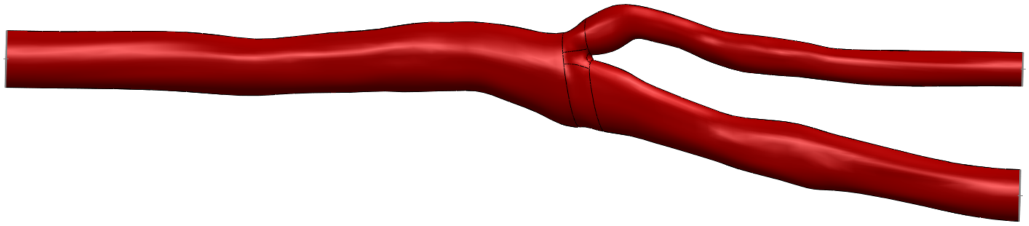


Figure 2.7: Finished geometry reconstruction

Chapter 3

VALIDATION OF THE NUMERICAL SOLUTION TO THE MASS TRANSFER PROBLEM

The high Peclet numbers found in cardiovascular flows makes the study of the mass transfer in large arteries and veins challenging and computationally expensive. The mass boundary layers develop much more slowly than its velocity counterpart which can lead to sharp gradients in the scalar fields that require a much finer mesh to resolve. This has led many past studies to be under-resolved, which causes the solution to have a lower effective Peclet number and underestimate the effects of flow features on the concentration field. In order to maximize the computational resources available for this thesis, a validation study was completed to determine the grid resolution required to resolve the mass transfer near the arterial wall.

A simplified geometry was investigated because analytic solutions can be obtained for use as a benchmark for the computational results. Two wall conditions were explored, constant wall flux and constant wall concentration.

3.1 Constant Wall Flux

The validation study compares the results of analytic and computational solutions to a simplified geometry representative of arterial flow. The analytic solution is used as a benchmark for determining the accuracy of the computational simulation.

3.1.1 Simplified Geometry

The mesh resolution in the radial direction was validated using the simplified case of flow through a circular straight pipe with constant cross section and a fully-developed velocity profile. The scalar profile is uniform for $z < 0$ and there is no mass flux ($\frac{d\phi}{dr} = 0$) at the walls. For $z > 0$ there is a constant scalar flux q_1. These simplifications allow for analytic

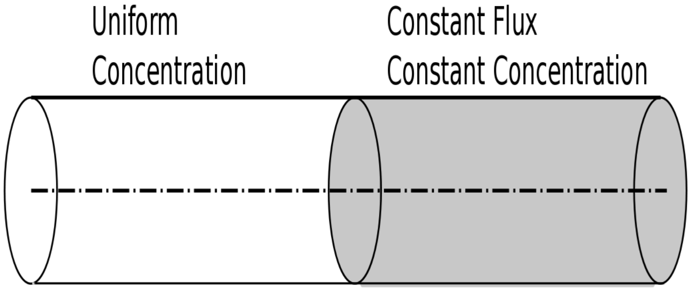


Figure 3.1: Cylinder geometry for scalar validation study

solutions to be calculated and used as the benchmark for comparison to CFD simulations conducted with various grid spacings using FLUENT. Figure 3.1.1 outlines the simplified geometry used in the benchmark problem.

3.1.2 Governing Equations

The constitutive equations are the Navier-Stokes equations, continuity, and the convection-diffusion equation for scalar transport. Assuming axisymmetric, fully-developed flow in a pipe, the Navier-Stokes and continuity is reduced to Poiseuille flow:

$$v_z = v_{max}\left[1 - \left(\frac{r}{R}\right)^2\right] \tag{3.1}$$

The scalar-transport equations takes the following form in cylindrical coordinates:

$$v_z\frac{\partial \phi}{\partial z} = D\left[\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial \phi}{\partial r}\right) + \frac{\partial^2 \phi}{\partial z^2}\right] \tag{3.2}$$

where r is the radial distance from the centerline, z is the axial distance measured from the step in wall flux, ϕ is the scalar concentration, and D is the scalar diffusivity.

3.1.3 Analytical Solution[1]

Typically, for high Peclet number flows, the scalar diffusion in the z-direction is much smaller than the convective transfer. This is exactly true when scalar transport becomes fully-developed. Assumming negligible axial scalar diffusion and plugging in equations 3.1 for the velocity field simplifies the scalar equation to:

$$v_{max}\left[1-\left(\frac{r}{R}\right)^2\right]\frac{\partial\phi}{\partial z}=\frac{D}{r}\frac{\partial}{\partial r}\left(r\frac{\partial\phi}{\partial r}\right) \tag{3.3}$$

with the following boundary conditions:

$$r = 0 \qquad \phi = \text{finite} \tag{3.4a}$$

$$r = R \qquad -D\rho\frac{\partial\phi}{\partial r} = q_1 \text{ (constant)} \tag{3.4b}$$

$$z = 0 \qquad \phi = \phi_0 \text{ (for all r)} \tag{3.4c}$$

It is helpful to non-dimensionalize equations 3.3 and 3.4. The following non-dimensional variables are used:

$$\Theta=\frac{\phi-\phi_0}{\frac{q_1R}{\rho D}} \tag{3.5a}$$

$$\xi=\frac{r}{R} \tag{3.5b}$$

$$\zeta=\frac{zD}{v_{max}R^2}=\left[\frac{z}{R}\right]\frac{1}{Pe} \tag{3.5c}$$

where Pe is the Peclet number and is defined as $Pe = Re \cdot Sc$ or the Reynold's number multiplied by the Schmidt number. Plugging equation 3.5 into equation 3.3 yields:

$$(1-\xi^2)\frac{\partial\Theta}{\partial\zeta}=\frac{1}{\xi}\frac{\partial}{\partial\xi}\left(\xi\frac{\partial\Theta}{\partial\xi}\right) \tag{3.6}$$

with the following boundary conditions:

[1] This derivation is based on the heat transfer counterpart found in Bird et. al. [2]

$$\xi = 0 \qquad \Theta = \text{finite} \tag{3.7a}$$

$$\xi = 1 \qquad -\frac{\partial \Theta}{\partial \xi} = 1 \tag{3.7b}$$

$$\zeta = 0 \qquad \Theta = 0 \tag{3.7c}$$

The profile of the scalar concentration far downstream will become fully developed and can be expected to have the following form:

$$\Theta_\infty = C_0\zeta + \psi(\xi) \tag{3.8}$$

where C_0 is a constant.

Substituting equation 3.8 into 3.6 yields:

$$\frac{1}{\xi}\frac{\partial}{\partial \xi}\left(\xi\frac{\partial \psi}{\partial \xi}\right) = C_0\left(1-\xi^2\right) \tag{3.9}$$

Integrating twice with respect to ξ and plugging in the boundary conditions to solve for the constants of integration yields the fully-developed solution:

$$\Theta_\infty(\xi,\zeta) = \frac{7}{24} - 4\zeta - \xi^2 + \frac{\xi^4}{4} \tag{3.10}$$

While the fully-developed solution is useful for problems concerning scalar transport in long straight pipes, the arterial system is changing direction too often for this solution to be useful. Instead, the entry length solution of the problem described above is used to validate the scalar transport.

The exact solution of the scalar profile can be expected to have the following form:

$$\Theta(\xi,\zeta) = \Theta_\infty(\xi,\zeta) - \Theta_d(\xi,\zeta) \tag{3.11}$$

Θ_d must decay to zero as $\zeta \to \infty$ in order for equation 3.11 to hold true. When solving for Θ_d, it must satisfy equation 3.9 and the boundary conditions are now:

$$\xi = 0 \qquad \frac{\partial \Theta_d}{\partial \xi} = 0 \tag{3.12a}$$

$$\xi = 1 \qquad \frac{\partial \Theta_d}{\partial \xi} = 0 \tag{3.12b}$$

$$\zeta = 0 \qquad \Theta_d = \Theta_\infty(\xi, 0) \tag{3.12c}$$

The solution of $\Theta_d(\xi, \zeta)$ is expected to have the following form:

$$\Theta_d(\xi, \zeta) = \Xi(\xi) Z(\zeta) \tag{3.13}$$

Plugging this into equation 3.9 and separating variables, results in the following two ordinary differential equations:

$$\frac{\partial Z}{\partial \zeta} = -c^2 Z \tag{3.14a}$$

$$\frac{1}{\xi}\frac{\partial}{\partial \xi}\left(\xi \frac{\partial \Xi}{\partial \xi}\right) + c^2 \left(1 - \xi^2\right) \Xi = 0 \tag{3.14b}$$

Equation 3.14a is a Sturm-Liouville problem on account of the boundary conditions in equation 3.12. Thus, we know that an infinite number of eigenvalues c_i and corresponding eigenfunctions Ξ_i exist. The solution must have the following form:

$$\Theta = \Theta_\infty(\xi, \zeta) - \sum_{i=1}^{\infty} A_i e^{-c_i \zeta \Xi_i(\xi)} \tag{3.15}$$

with

$$A_i = \frac{\int_0^1 \Theta_\infty(\xi, 0) \Xi_i(\xi) \left(1 - \xi^2\right) \xi d\xi}{\int_0^1 \Xi_i(\xi)^2 \left(1 - \xi^2\right)} \tag{3.16}$$

The problem is now solving for the eigenvalues and eigenfunctions of equation 3.14a. A shooting method was employed to solve for the first 1000 eigenvalues and eigenfunctions. The details of the shooting method can be found in Appendix B.

The eigenfunction solution is feasible for moderate Peclet numbers (10,000s) but higher Peclet numbers require a much larger number of eigenfunctions to accurately describe the solution near the step in wall flux. A more efficient solution for high Peclet number flows is an asymptotic solution for short distances near the step in scalar flux at the wall. The following assumptions are made:

1. The effect of curvature may be neglected and the problem is treated as scalar transport over a flat wall.

2. A new axial variable is defined, $s = R - r$, with domain (0,∞)

3. The velocity profile is treated as linear. The slope is given by the true slope at the wall:
   $v_z(s) = v_0 \frac{s}{R}$ where $v_0 = v_{max} 2 \frac{r}{R}$

The result of these assumptions is a function for Θ that contains gamma functions. The derivation of the result can be found in Bird et al. [2] and the resulting solution is shown in Equation 3.17.

$$\Theta = \sqrt[3]{\frac{9}{2}\zeta}\left[\frac{e^{-\chi^3}}{\Gamma\left(\frac{2}{3}\right)} - \chi\left(1 - \frac{\Gamma\left(\frac{2}{3}, \chi^3\right)}{\Gamma\left(\frac{2}{3}\right)}\right)\right] \tag{3.17}$$

where

$$\chi = \frac{1 - \xi}{\sqrt[3]{\frac{9}{2}\zeta}} \tag{3.18}$$

$\Gamma\left(\frac{2}{3}\right)$ is a complete gamma function; $\Gamma\left(\frac{2}{3}, \chi^3\right)$ is an incomplete gamma function; ξ and ζ are the same as in the eigenvalue solution.

3.1.4 Computational Simulation

The computational solution was conducted using 2D-axisymmetric and 3D geometries within FLUENT. The simulation was set to match the fluid properties and boundary conditions of the analytical solutions. The simulations were started on a coarse mesh and repeated

Table 3.1: List of mesh sizes used in 2D validation study

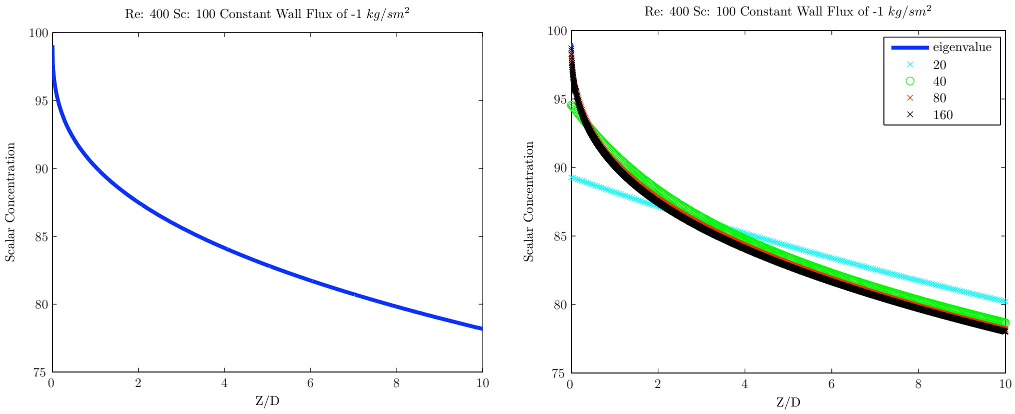
Mesh	**Cell Size** $\left(\dfrac{\Delta r}{D}\right)$
1	0.05
2	0.025
3	0.0125
4	0.00625

on subsequently refined meshes until the result of the simulation matched the analytical solution. Two-dimensional simulations were used to determine the radial resolution of the mesh required to resolve the scalar transport. Three-dimensional simulations were then conducted to verify the 2D findings and to develop scripts for use in the patient-specific geometry simulation.

3.1.5 Results

The 2D simulation consisted of four mesh resolutions. The mesh resolution was non-dimensionalized by the diameter of the tube. Figure 3.2a shows the analytic profile for the eigenvalue solution for a Peclet number of $Pe = 4 \cdot 10^4$. There is a sharp gradient in the scalar concentration near the step in scalar flux at the wall. Figure 3.2b shows the comparison of the four mesh sizes, listed in Table 3.1 to the eigenvalue solution. The coarse mesh of $\dfrac{\Delta r}{D} = 0.05$ is unable to capture the sharp gradients near the step in flux through the wall, leading to 10% error for the scalar concentration. The two finest resolutions, $\dfrac{\Delta r}{D} = 0.0125$ or 0.00625 accurately capture the sharp gradient and only have 2% error near the step in flux. The good agreement between the 2D simulation and the analytical results validated FLUENT's scalar transport equation used in the patient-specific simulations.

3D simulations were validated using higher Peclet numbers, $Pe = 1.2 \cdot 10^6$. The results are shown in Figure 3.3 comparing the eigenvalue solution, the asymptotic gamma function solution, and two computational mesh sizes. The two mesh sizes were $\dfrac{\Delta r}{D} = 0.003$ and 0.0003. The mass transfer results have been non-dimensionalized by using the



(a) Eigenvalue solution

(b) 2D simulations

Figure 3.2: Comparing eigenvalue and 2D numerical solutions for $Re = 400$ and $Sc = 100$

Sherwood number:

$$Sh = \frac{N_s 2R}{\rho D\left(w_s - w_\infty\right)} \tag{3.19}$$

where N_s is the species wall flux, R is the cylinder radius, w_s is the species concentration at the wall, and w_∞ is the species concentration at the inlet. The Sherwood number asymptotes to infinity as the step in scalar flux at the wall is approached, where $w_s = w_\infty$. Note that for the higher Peclet number, the eigenvalue solution has an inadequate number of eigenvalues and eigenfunctions. The asymptotic gamma function solution is a better metric to use for comparison here. Both mesh resolutions compared well to the analytic solution with less than 10% error near the step in flux. The mesh resolution used for the patient-specific geometries was $\dfrac{\Delta r}{D} = 0.003$. This will provide highly resolved scalar fields for $Sc = 100$ in the range of Reynolds numbers seen in the CAB. The scalar fields for species with $Sc = 1000$ will be potentially under-resolved, especially at the bifurcation apex, but using a radial resolution of $\dfrac{\Delta r}{D} = 0.003$ already pushes the limits of the computational resources available.

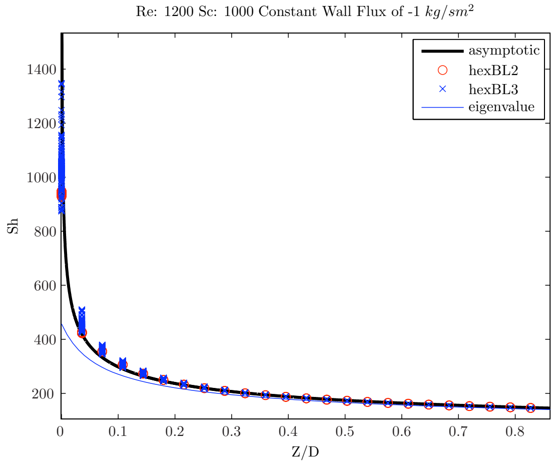


Figure 3.3: Comparison of wall concentrations for 3D simulations; eigenvalue solution; and asymptotic solution

Chapter 4

NUMERICAL METHODS

4.1 Meshing

The numerical discretization process was a key part of this thesis. As mentioned earlier, the high Peclet numbers found in cardiovascular flows lead to steep species concentration gradients. High resolution in the radial direction near the arterial wall is required to resolve the mass transfer accurately. To keep the mesh cell count at a manageable level, various strategies are implemented. High aspect ratios in the arterial branches away from the bifurcation, a fast growing boundary layer, and surface mesh resolution studies were used. The following workflow was used for the meshing process:

1. Import arterial geometry from SolidWorks into GAMBIT

2. Break the domain into four parts: 3 arterial branches and a bifurcation region

3. Generate a surface mesh of the geometry

4. Generate 3D meshes for the arterial branches

5. Export mesh into TGrid

6. Generate boundary / prism layer in bifurcation region

7. Import back into GAMBIT

8. Generate 3D mesh of remaining interior of bifurcation region

9. Set up boundary conditions and export mesh to FLUENT

4.1.1 Meshing Tools

The process involves two meshing programs: GAMBIT and TGrid. GAMBIT is used for generating surface meshes from the SolidWorks geometry and creating volume meshes for the arterial branches. TGrid is used to create the prismatic boundary layer in the bifurcation region. GAMBIT is a geometry creation and mesh generation tool for both 2D and 3D geometries. Although it can mesh boundary layers easily for a simpler geometry, it can be challenging to generate structured boundary layers for complex geometries like an arterial bifurcation. Structured meshes are meshes that are made solely of hexagonal and prismatic cells instead of tetrahedral cells. Structured meshes benefit from better numerical properties. This includes less numerical diffusion and lower cell counts than unstructured meshes for the same resolution. TGrid is a mesh generation tool specifically designed for growing prism layers from surface meshes and can handle complex geometries. TGrid is limited in that it cannot create geometry or surface meshes on its own so an additional surface meshing tool such as Gambit is always required. TGrid also allows for a user to manually clean up the quality of the mesh. Specifically, it allows the user to select specific node points and move them to improve various mesh quality quantities, such as skewness. This step is completed after a 3D mesh has been generated and there are highly skewed elements still remaining after generation.

4.1.2 Breaking Up the Domain

After importing the geometry from SolidWorks the domain is broken up into four parts: the three arterial branches (CCA, ICA, and ECA) and a bifurcation region as shown in Figure 4.1. The break-lines were made perpendicular to the axial direction to minimize skewness of elements near the arterial wall. The bifurcation region is sized such that it protrudes into each of the branches three diameters. The cubic spline estimation of centerlines, described in Section 2.2.1 was used to estimate the normal vectors used to create the break-lines.

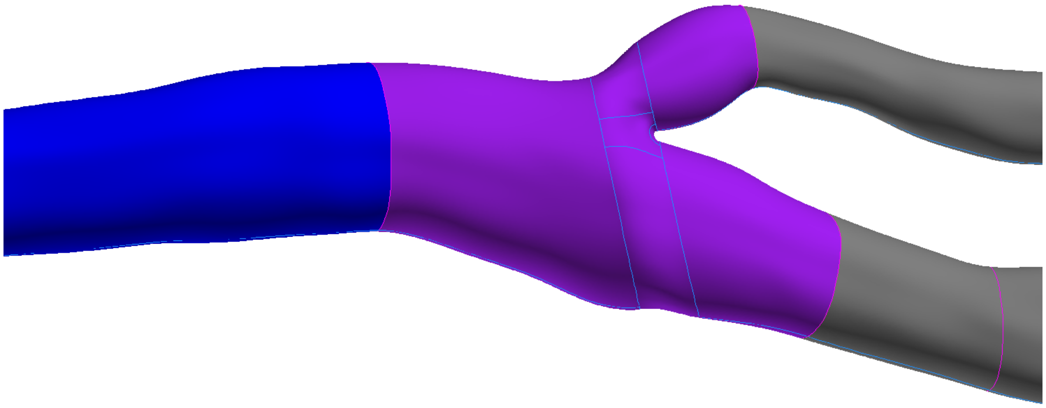


Figure 4.1: The four regions the geometry was broken into: CCA, ICA, ECA, and Bifurcation regions

4.1.3 Surface Meshing

With the domain split into branches and a bifurcation, the surface of each region must be meshed. The three branches were meshed with an aspect ratio (AR) of 1:1 near the bifurcation region and slowly grown with a size function to an AR of 5:1, see Figures 4.2 and 4.3. The AR is described as the ratio between the axial length and circumferential length of an element (axial:circumferential). The bifurcation region was meshed with rectangular elements near the arterial branches and a band of triangular elements in the complex region of the bifurcation apex. Figures 4.3 and 4.4 show this in more detail. The AR is 1:1 throughout the bifurcation region. This was done for two reasons. First, the bifurcation is where the most complex, secondary flows originate and care was taken to capture the mass transport and shear in high detail in this region. Second, the interior mesh of the bifurcation region consists of mostly tetrahedral elements (not including the prism boundary layer created by TGrid) and low surface aspect ratios are required to have low skewness in the volume mesh.

The circumferential resolution was chosen after conducting a short surface resolution study. The study compared two meshes of the carotid geometry using a steady flow rate matching the Reynolds number used at peak systole (Re = 900) for this simulation. The

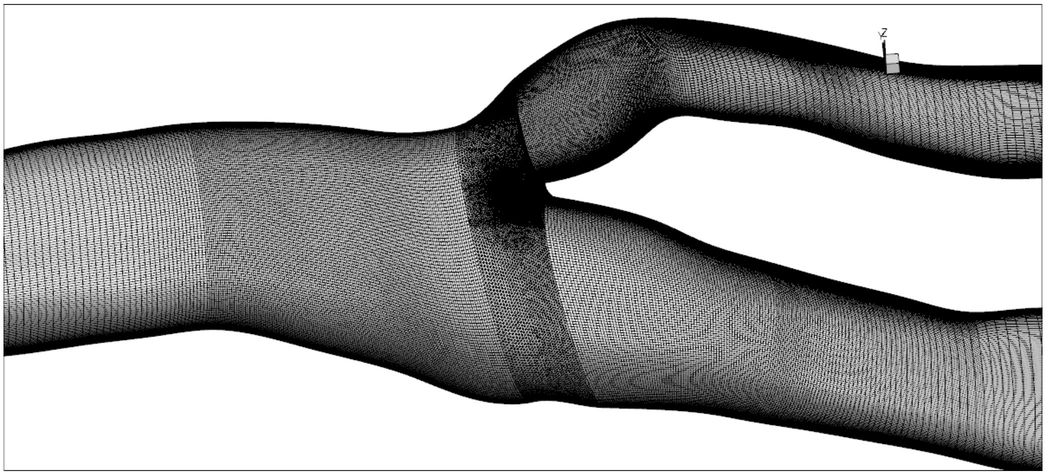


Figure 4.2: Surface mesh near the bifurcation region

Table 4.1: Parameters for boundary layer size and growth

first cell height	$\frac{\Delta r}{D} = 0.003$
Growth Rate	1.18
# of Rows	10

two resolutions compared had 250 or 350 points discretizing the circumferential direction. The two axial resolutions used the methodology described earlier and the radial resolution was identical in the two cases. A comparison of maxima and average quantities of scalar concentration and flux through the wall showed a percent difference of 5% or less for the two meshes. The circumferential resolution of 250 points was chosen to keep the cell count at a manageable level for the transient simulation. This reduced the number of elements in the final volume by half.

4.1.4 Volume Meshing

The results from the validation study are used in meshing the volumes of the arterial branches. A boundary layer is defined using the parameters shown in Table 4.1

The inlets and outlets are meshed using the *pave* algorithm. A size function is used to

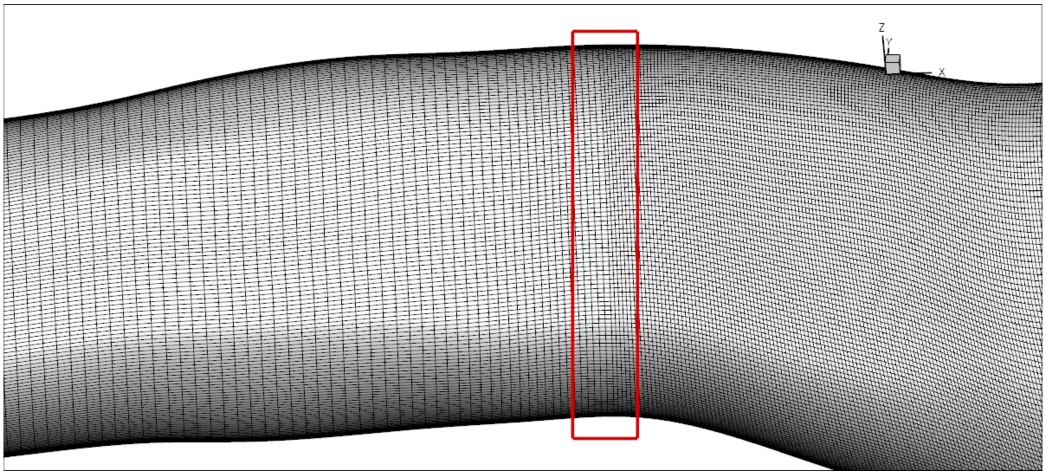


Figure 4.3: Surface mesh of CCA and bifurcation region. Red box highlights where the CCA and bifurcation meet and the 1:1 apsect ratio begins for the bifurcation region

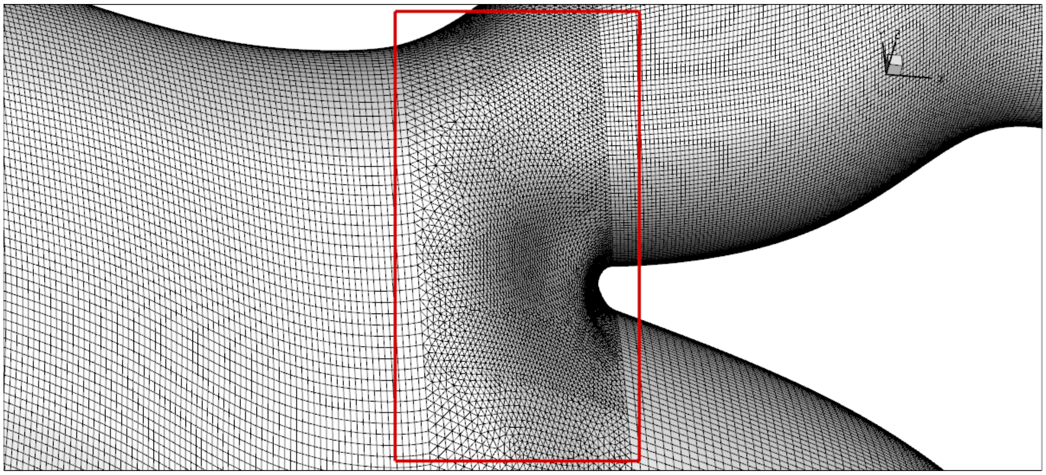


Figure 4.4: Surface mesh in the bifurcation region. Red box is highlighting the region where there is triangular elements in the surface mesh.

continue growing the size of the elements after the boundary layer elements, although in a less structured way. The ICA mesh is shown in Figure 4.5. The branch volumes are then meshed using the *Cooper* scheme. This scheme treats the grid as a cylinder and then maps it onto the volume. The arterial branches are an ideal geometry for this algorithm.

The mesh is exported to TGrid to generate the boundary layer in the bifurcation region. The parameters are the same as in the branches, detailed in Table 4.1. The mesh is then imported back into Gambit for meshing of the remainder of the bifurcation region with tetrahedral elements. A size-function with a growth rate of 1.18, the same rate used for the boundary layer, is used to continue growing the elements away from the arterial wall. After this step is completed the arterial geometry has a complete 3D mesh. The final mesh contained 7 million elements with most located in the bifurcation region. The mesh quality is good considering the high aspect ratio due to the fine radial resolution near the wall. 95% of the cells had skewness below 0.5. 28 cells are in the range of 0.8 to 0.9 and considered to have high skew. These cells are located near the wall where the bifurcation region connects with the branches. The high skewness in this region is due to the mesh converting from hexagonal to tetrahedral in both the axial and radial directions. High skewness can potentially increase error in the simulation or cause the simulation to diverge. With so few highly skewed cells however, this was not considered a significant problem.

4.2 FLUENT Setup

The commercial finite volume solver FLUENT, was used to solve continuity, the Navier-Stokes equations, and the advection-diffusion equation for scalar transport. For incompressible flow continuity is:

$$\nabla \cdot \mathbf{v} = 0 \tag{4.1}$$

The Navier-Stokes equations take the form:

$$\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v} = -\nabla \frac{p}{\rho} + \nu \nabla^2 \mathbf{v} \tag{4.2}$$

where $\mathbf{v}$ is the velocity (m/s), p is the static pressure (Pa), ρ is the fluid density

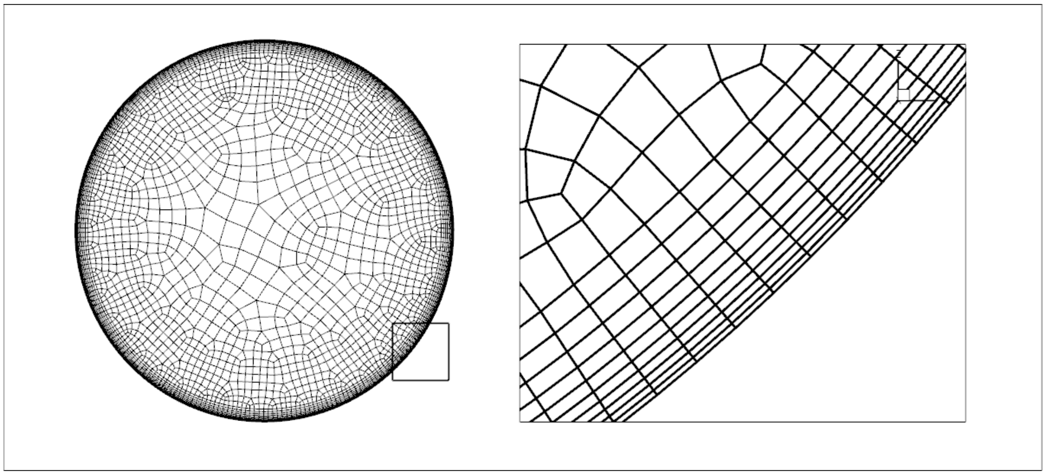


Figure 4.5: Left: surface mesh of the CCA inlet; Right: close up of boundary layer region

(kg/m^3),and ν is the kinematic viscosity (m^2/s). The scalar transport equation takes the form:

$$\frac{\partial \phi_k}{\partial t} + \nabla \mathbf{v} \phi_k = D \nabla^2 \phi_k \tag{4.3}$$

where ϕ_k is the scalar concentration and D is the scalar diffusion coefficient (m^2/s). The spatial discretization uses second order central differencing and the temporal discretization uses a second order implicit method. The PISO method was used for the pressure-velocity coupling. The PISO method is a pressure-based segregated algorithm that uses the predictor-corrector approach for convergence. It also has a correction for cells with high skewness and is recommended for transient problems. More specific details can be found in the FLUENT manual [1].

4.2.1 Boundary Conditions

Accurate and adequate boundary conditions are an important part of any CFD simulation.

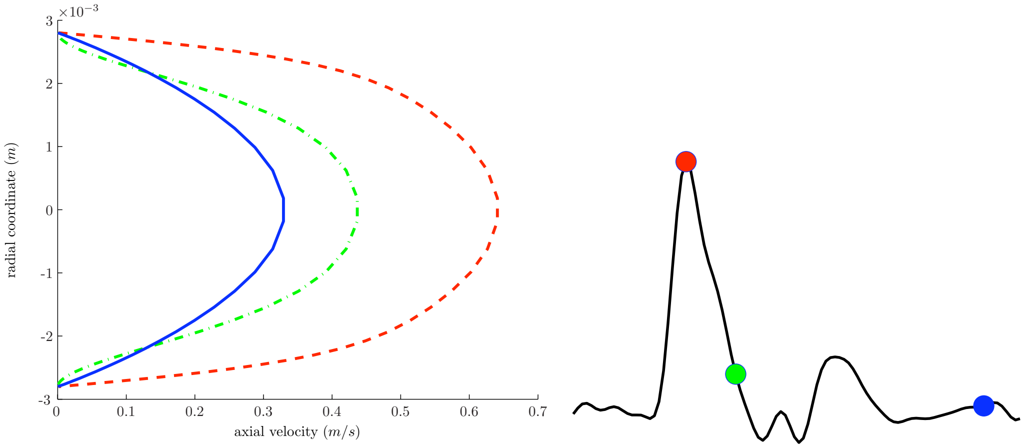
Velcocity Boundary Conditions

The pulsatile, axial flow rates of the CCA can accurately be obtained through medical imaging techniques, however secondary flows that arise from upstream curvature and bifurcations are often an order of magnitude smaller that the axial flow and this makes their measurement difficult. As a result, secondary velocities are often neglected in prescribing velocity inlets and outlets for cardiovascular flows. A study by Moyle et. al. [21] investigated the effect of including secondary velocities to WSS and OSI statistics. In the study, the Womersley velocity profile [30] is prescribed at the inlet and various inlet extensions including a straight tube and helices of various curvature and torsion. It was found that there is about 13% variance in the WSS and OSI statistics when adding the secondary flows to an inlet compared to just a Womersley profile. Although this may seem high, it was also shown that there is greater variability from the geometry uncertainty, about 48%. The CCA branch length and flow extensions used in this thesis are comparable to the work done by Moyle. This will make the error, introduced by neglecting the secondary flows, of the same order of that caused by the geometric reconstruction. The profile used was a 15 term representation, and Figure 4.6 shows the axial velocity profile at peak systole, end of systole, and at rest during diastole. Figure 4.6 also shows the volume flowrate over the cardiac cycle.

For this thesis, there was no data for the CCA inlets or the ICA and ECA outlets from the specific patient scan. A typical CCA flow rate waveform was used, scaled to provide the peak and mean Reynolds numbers, as well as the desired Womersley number. The ICA and ECA outlets were prescribed for a flowsplit of 80:20 and 60:40, the same used from other studies by Ku et. al. [15]. For future studies, if the outlet flowrates are known, the time-dependent RC model developed by Grinberg and Karniadakis [12] is recommended.

Scalar Boundary Conditions

The effect of WSS and species concentration on wall permeability and mass transfer rates is currently known on a qualitative level. Profiles of species concentration are almost impossible to obtain, although bulk concentrations can be measured. On account of these



(a) Womersely solutions

(b)

Figure 4.6: Axial Velocity for CCA inlet at specified times during the cardiac cycle

limitations, simplified scalar transport boundary conditions are used. A constant concentration of 100 was used at the CCA inlet and a no diffusive flux condition was prescribed at the ICA and ECA outlets. Three boundary conditions were implemented at the arterial wall. A no-flux boundary condition was used for a "flush test" to estimate residence times and highlight areas of recirculation. A constant wall flux of $\pm 1\frac{1}{m^2}$ and constant wall concentration of 0 were used to look at mass transfer patterns of NO and other ROSs, oxygen, LDL, etc. The simplified boundary conditions used in this thesis allow us to identify regions where the mass-transfer is inhibited or enhanced by the flow effects.

4.2.2 Time Step Convergence

10,000 time steps per cardiac cycle were used in the simulations. Larger time step sizes attemped caused the solution to diverge during systole. This is likely due to the fine resolution in the boundary layer near the apex of the bifurcation, where the highest velocities of the CCA impinge. This causes the CFL condition to be highest in this region. The CFL is defined as:

$$CFL = \frac{v_i \Delta t}{\Delta x_i} \tag{4.4}$$

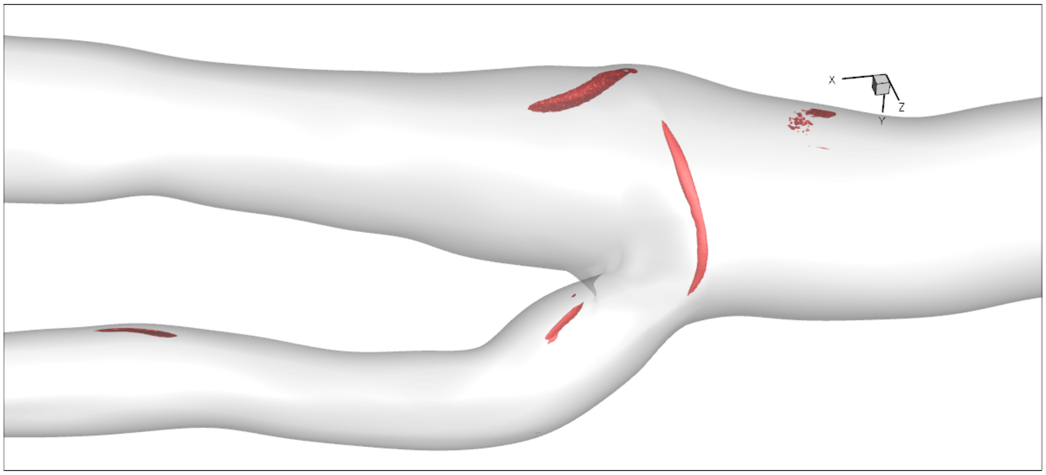


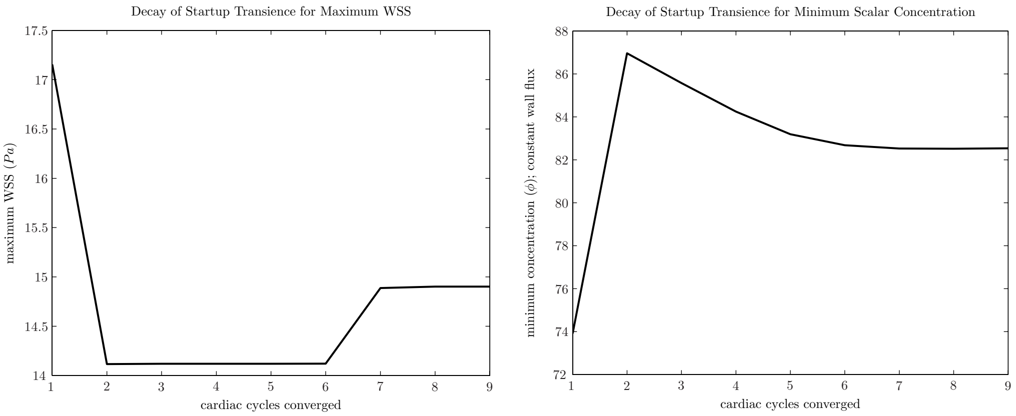
Figure 4.7: Iso-surface of RMS difference (1% error) between scalar results for difference timestep sizes

where v_i is the velocity (m/s) in the i^{th} direction, Δt is the time step size (s), and Δx_i is the length of the finite volume cell in the i^{th} direction. For the 10,000 time step case, the maximum CFL was 27 during peak systole and the average CFL was around 10.

A timestep convergence study was completed to ensure that the scalar transport was fully resolved in time. Three timestep sizes were investigated: 10,000, 20,000, and 40,000 timesteps over the cardiac cycle. Each case was run until it reached peak systole and the RMS scalar difference between the various timestep was computed. The maximum error was found to be 3% in the region of the bifurcation when the 10,000 and 40,000 timestep cases were compared. Figure 4.7 shows the iso-surface of 1% error when comparing the 10,000 and 40,000 case. The error is mostly located in the bifurcation region, and only a small fraction of the computational domain is affected. With small and localized error, the timestep of 10,000 per cardiac cycle was chosen for use in the simulations.

4.2.3 Startup Transience

The flow field was initialized using a steady flow solution matching the Reynolds number for the start of the cardiac flowrate waveform ($Re = 323$), see Figure 4.6. The time dependent simulation was then iterated until the startup transience had damped away. The maximum



(a) Decay of Maximum WSS

(b) Decay of Minimum Scalar Concentration

Figure 4.8: Plots showing decay of startup transience for maximum WSS and minimum scalar concentration along the arterial wall

WSS and minimum scalar concentration for constant wall flux were tracked at the end of each cardiac cycle. Figure 4.8 shows the results of the startup decay. The startup transience was determined to be cleared after six cardiac cycles. The data was then changed to save 100 timesteps per cardiac cycle instead of just 1. It was then noticed that the maximum WSS jumped between the sixth and seventh cycle, while the scalar concentration remained unchanged. The difference of the WSS was computed for the entire arterial surface and found to be less than 0.2 *Pa* everywhere except at the apex of the bifurcation, as shown in Figure 4.9. The localized difference in the apex of the bifurcation will not influence the analysis as the main regions of interest are in the carotid sinus.

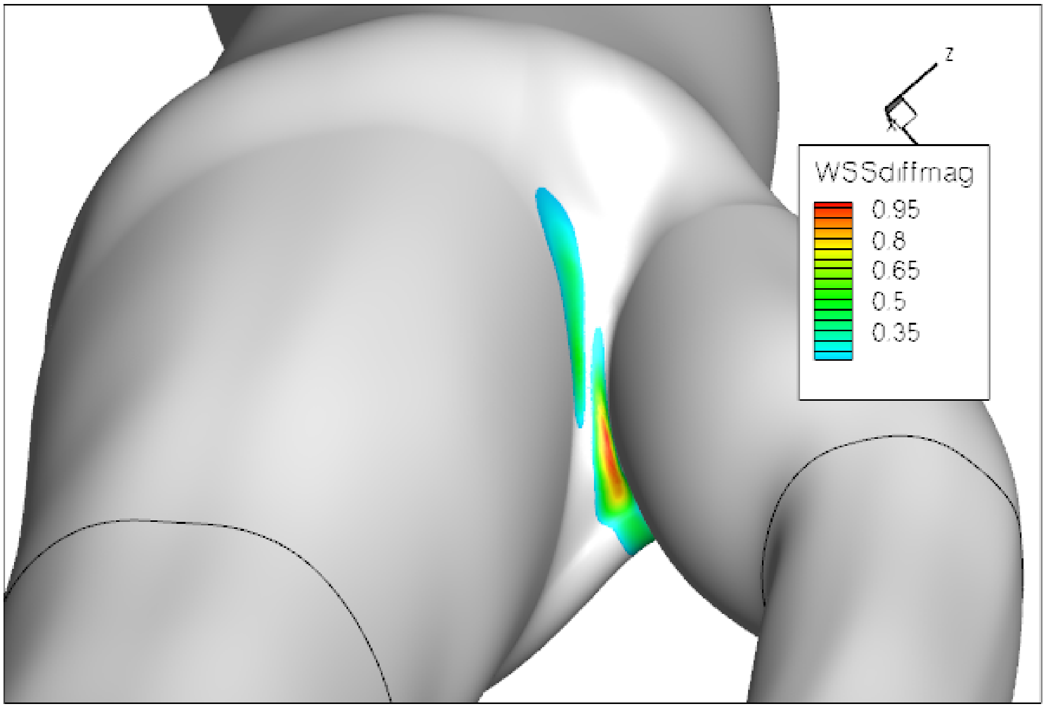


Figure 4.9: RMS difference of WSS between end of 6th cycle and 7th cycle; Most of domain has error less than 0.2 Pa (2 $dynes/cm^2$)

Chapter 5
RESULTS

This section covers the results from analysis of the momentum and scalar transport in the carotid artery bifurcation numerical model described in the previous chapter. Typical velocity profiles found in the CAB are described and results from the patient-specific geometry used in this thesis are given. Shear stress statistics are calculated with results from specific times in the cardiac cycle as well as cycle-averaged values. The scalar transport is analyzed using three different tests. The first case imposed constant flux out of the arterial wall and measures the resulting surface concentration. This can be used as an approximation for species that are consumed at the wall (i.e. O_2 or ROS) or if the direction of the flux is inverted, NO generation at the wall. The second case imposes a constant scalar concentration at the wall and measures the resulting flux at the surface. The last case is a “flush” test, which initializes the domain with constant scalar value and captures the dynamics of how the CAB geometry is cleared out. There is no scalar flux at the walls, only a zero concentration value is set at the CCA inlet. This test captures regions of long fluid residence times and regions of recirculation. Lastly, comparisons between patterns in the shear and scalar transport are made and a simple thresholding technique is implemented. This highlights differences between the two mechanisms for flow-induced effects on atherosclerosis origin and progression.

5.1 Velocity Profiles

The velocity profiles found in branching geometries can be qualitatively described easily [23]. The flow is split into two streams at the apex of the bifurcation, where new boundary layers must develop on the inner walls of the branches. The localized high pressure region caused by the stagnation point at the bifurcation produces secondary velocities similar to Dean flow. The peak velocity profiles are skewed towards the inner wall and separation regions are possible depending on the geometry.

The velocity profiles in the carotid artery bifurcation region for two different flow splits, 80:20 and 60:40, are shown in Figure 5.1. The results agree with the expected qualitative predictions and show that the size of the separated region in the carotid sinus is not sensitive to the flow split.

5.2 Shear Stress

The WSS seen in the CAB has large temporal and spatial variability. The maximum WSS occurs at the bifurcation apex, where the flow from the CCA impinges. The lowest WSS is found in the carotid sinus, where the flow separates and forms a recirculation bubble. The maximum WSS was 52 Pa during peak systole, while the lowest was 0.0012 Pa at the end of diastole.

5.2.1 Time Averaged Wall Shear Stress

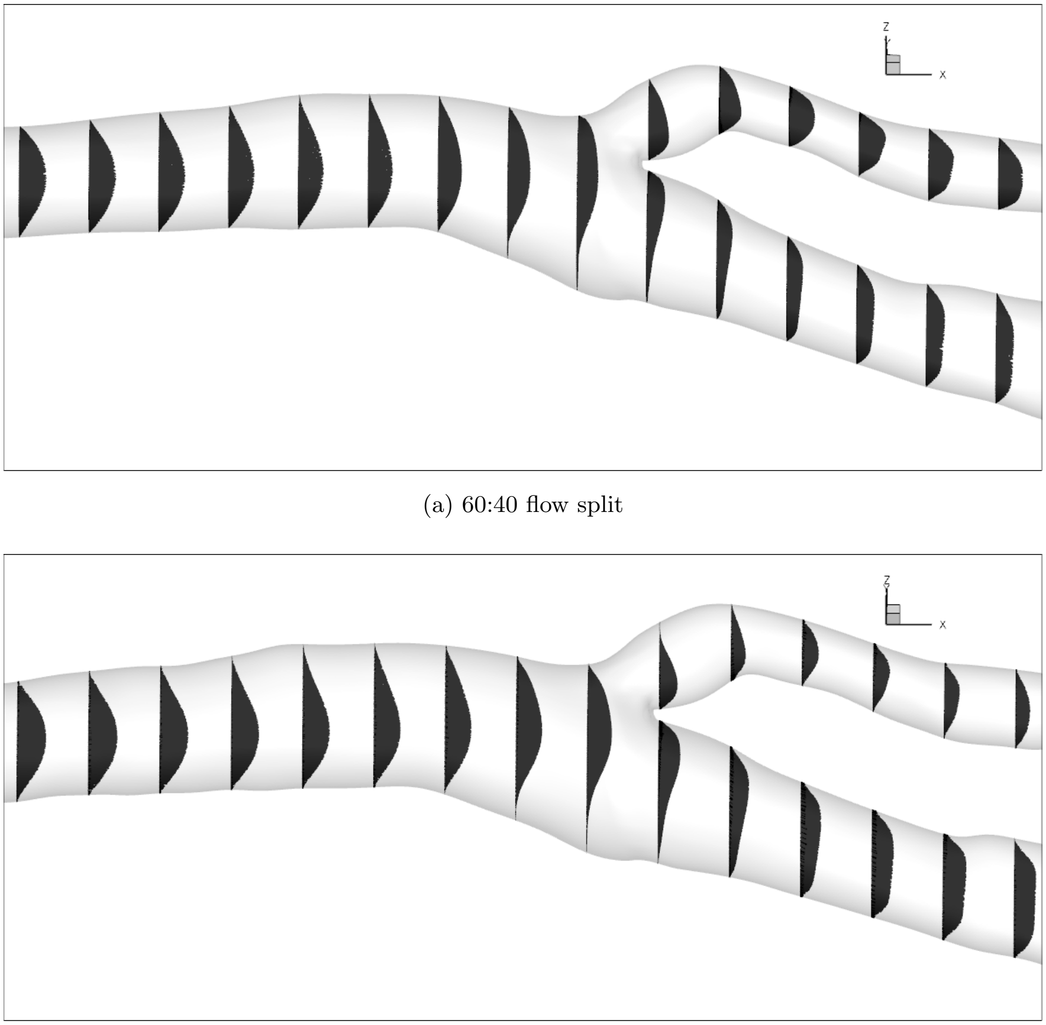
The large timescales associated with atherosclerosis suggest that phase-average values of WSS during the cardiac cycle could be important in understanding the interplay between shear and disease origin and progression. One such WSS statistic is the time-averaged magnitude of wall shear stress (TAWSS), which is defined as:

$$\hat{\tau} = \frac{1}{T}\int_0^T |\tau_i| dt \tag{5.1}$$

where T is a multiple of the cardiac period and $|\tau_i|$ is the magnitude of WSS at a specific point in time. The TAWSS is shown in Figure 5.2. The maximum is again at the bifurcation apex and has a value of 17 Pa. The minimum TAWSS is in the carotid sinus and has a value of 0.16 Pa. Figure 5.3 shows the region where the TAWSS is below 0.25 Pa. Only the carotid sinus region has TAWSS values in this range.

5.2.2 Oscillatory Shear Index

The oscillatory shear index (OSI), introduced by Ku et al. [14], is a WSS-related quantity that measures how often the WSS vector changes direction over the cardiac cycle. The



(a) 60:40 flow split

(b) 80:20 flow split

Figure 5.1: Velocity profiles during the resting period following diastole for two different flow splits

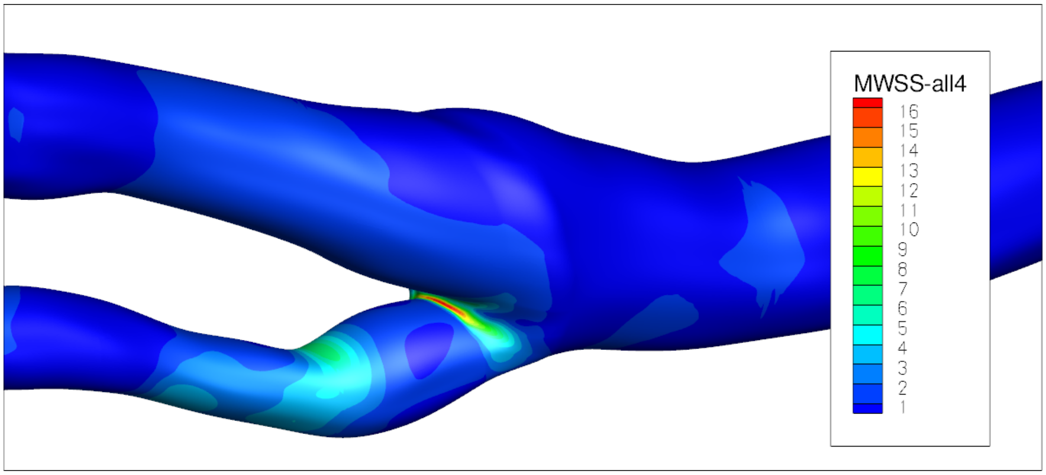


Figure 5.2: Time averaged wall shear stress (Pa)

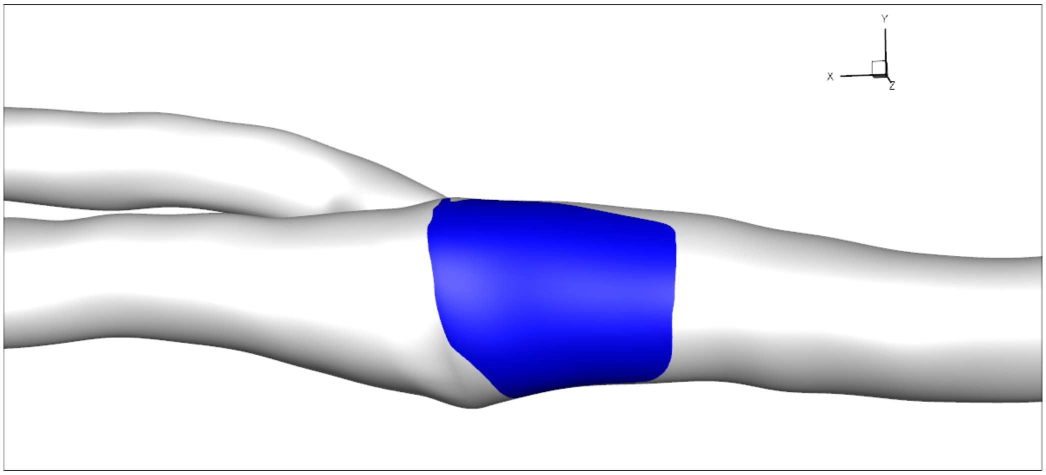


Figure 5.3: Region where TAWSS is below 0.25 Pa (2.5 $dynes/cm^2$)

formulation used in this thesis is the same used by Talor et al. [27]:

$$\text{OSI} = \frac{1}{2}\left(1 - \frac{\bar{\tau}}{\hat{\tau}}\right) \tag{5.2}$$

where $\hat{\tau}$ is the TAWSS described earlier and $\bar{\tau}$ is given by:

$$\bar{\tau} = |\frac{1}{T}\int_0^T \tau_i dt| \tag{5.3}$$

The OSI can vary from 0, when the WSS vector is always aligned in the same direction, and 0.5, which occurs when the WSS has a zero average over the cardiac cycle, for example with a sinusoidal time dependency.

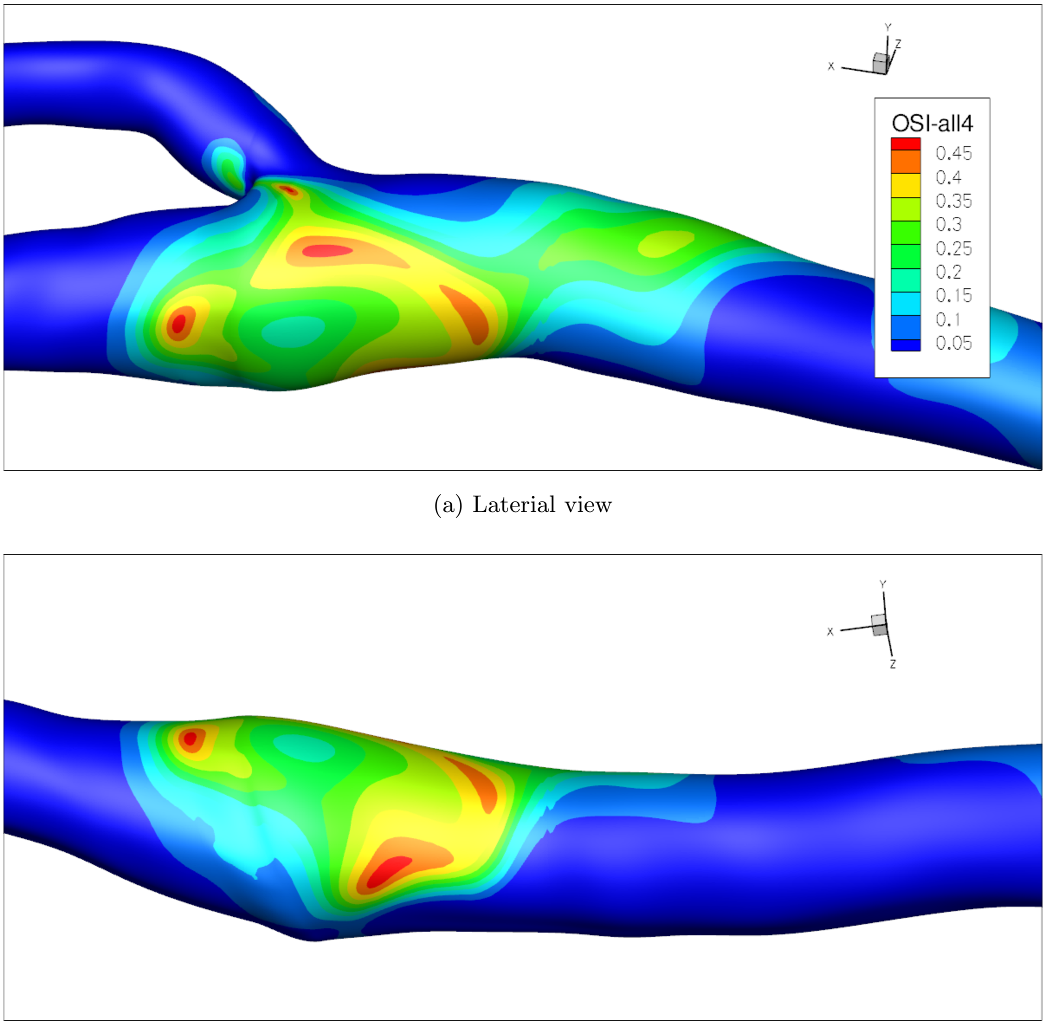
The OSI calculated over four cardiac cycles is shown in Figure 5.4. It is highest in the region of the carotid sinus were the flow is separated throughout the cardiac cycle. The highest values, where OSI > 0.49, are located at the regions of flow detachment and reattachment. The OSI is lowest in the arterial branches, where the WSS vector is mainly aligned with the axial flow direction.

The OSI correlates well with regions of low WSS. The region of TAWSS < 0.25 Pa, shown in Figure 5.3, matches the region of moderate to high OSI. A key difference can be noted, as the OSI is high at the region of reattachment where the TAWSS is above 2.5 Pa.

5.3 Scalar Transport

5.3.1 Constant Wall Flux

The surface species concentration for the case with constant flux out of the fluid domain is shown in Figure 5.5. This condition is an approximation for transport of species that are consumed uniformly by the endothelial layer or are transported at a constant rate into the arterial wall. In the region of the carotid sinus there is a deficiency of 20 concentration units with respect to the value at the core. This is due to the separation in the sinus region, which slows transport to the wall. Low transport to or away from the arterial walls in the carotid sinus have the potential to create regions of scalar concentrations detrimental to endothelial cell health. Inhibited mass transfer prevents species from entering the sinus region. The lowest regions of scalar concentration highly correlate with the regions of high OSI except



(a) Lateral view

(b) Posterior view

Figure 5.4: OSI averaged over four cardiac cycles

at the reattachment region. This high correlation between mass transfer and OSI patterns has the potential to allow for qualitative mass transfer studies to be completed using coarser discretization that can correctly resolve the flow but would under-resolve the high Schmidt number species transport. Care must be taken however, due to the OSI being high at regions of both flow detachment and reattachment while the mass transfer is inhibited or enhanced in these regions respectively.

The scalar concentration along the arterial wall does not change significantly throughout the cardiac cycle. This is shown in Figure 5.6, which shows the scalar concentration at three different times within in the cardiac cycle. This occurs even though there can be large changes in the species concentration in the core of the flow during the cardiac cycle, as seen in Figure 5.8. This is largely due to the low diffusion time scales associated with species in blood. An estimate of the diffusion response time, t_{diff}, can be given by:

$$t_{diff} = \frac{h_{MBL}^2}{D} \tag{5.4}$$

where h_{MBL} is the height of the mass boundary layer (MBL) and D is the species diffusivity in blood. The thickened MBL in Figure 5.7b has a response time of $t_{diff} \approx 16s$. For $t_{diff} >> T$ (the period of the cardiac cycle) the scalar transport occurs too slowly through the boundary layer and the changes in bulk concentration are averaged over the cardiac cycle. Thus, the surface concentration does not respond to the large changes in the bulk flow occuring at a “short” period or “fast” frequency. The slow diffusion times potentially allow for quantitative mass transfer studies to be completed with steady simulations. The pulsatility of the cardiac cycle has a large effect on the shear stress statistics, but there is little variation within the mass transfer. Conducting steady simulations significantly reduces the computational requirements for conducting mass transfer studies. It must be noted however, that the parameter range explored in this thesis is small and a larger range of Sc and Re numbers must be explored before the use of steady state simulations can be definitively advised.

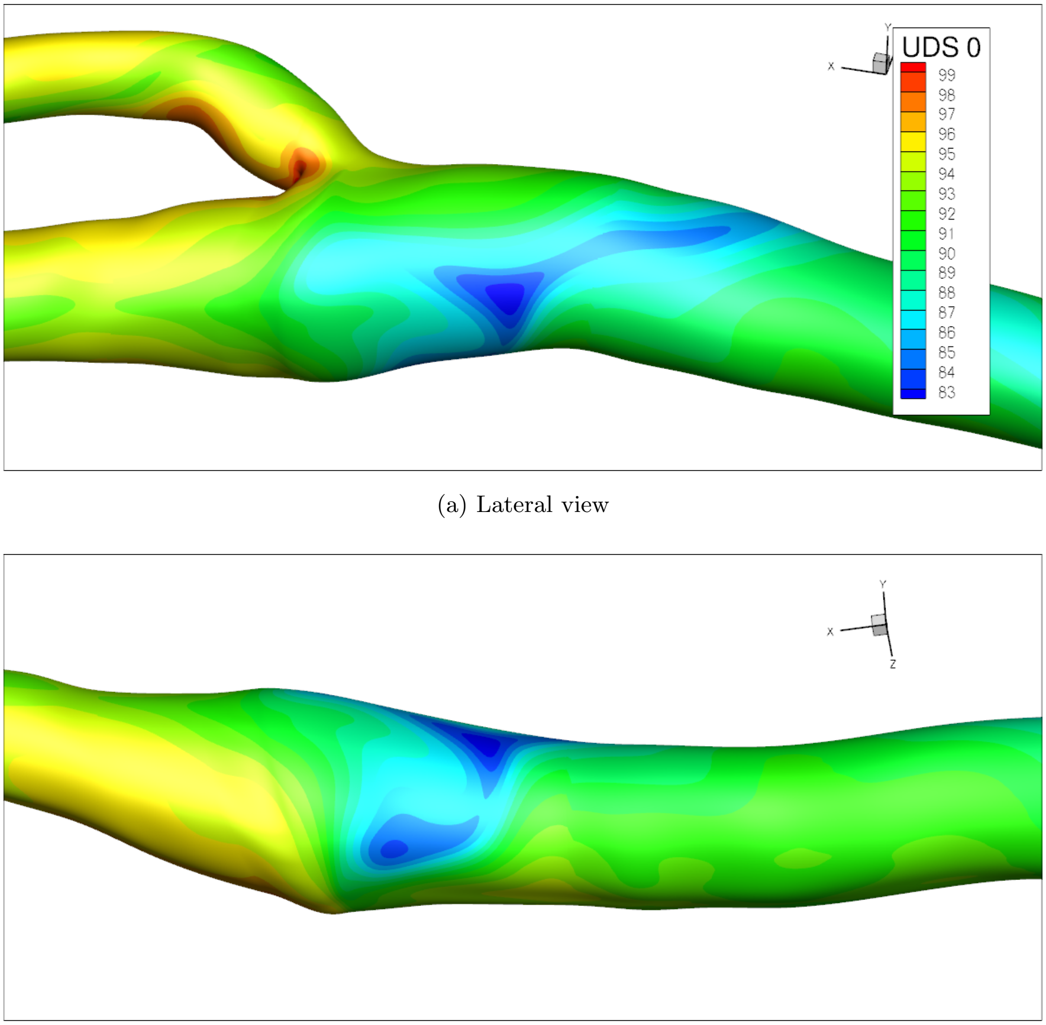
The thickened MBL shown in Figure 5.7b is caused by the detachment in the carotid sinus region, and the vorticity generated by the separation, seen in Figure 5.7c, pulls chemical species away from the wall.

5.3.2 Constant Wall Concentration

The constant concentration case can be used as a first approximation for species whose concentration is regulated by the physiology of the arterial wall or for mass transfer processes where the wall-side resistance is negligible. The results are skewed by the flow impinging at the apex of the bifurcation and the flux at the apex is significantly higher than the rest of the domain (40-50x). Figure 5.9 shows the species flux out of the fluid domain, into the arterial wall. Note that the patterns seen in low WSS and high OSI and low species concentration are qualitatively similar. The agreement between mass transfer and OSI breaks down again at the point of flow reattachment, but correlate better than with low TAWSS upstream in the CCA. The regions of high OSI and low species flux both have a meandering “tail” that protrudes upstream of the carotid sinus.

5.3.3 Flush Test

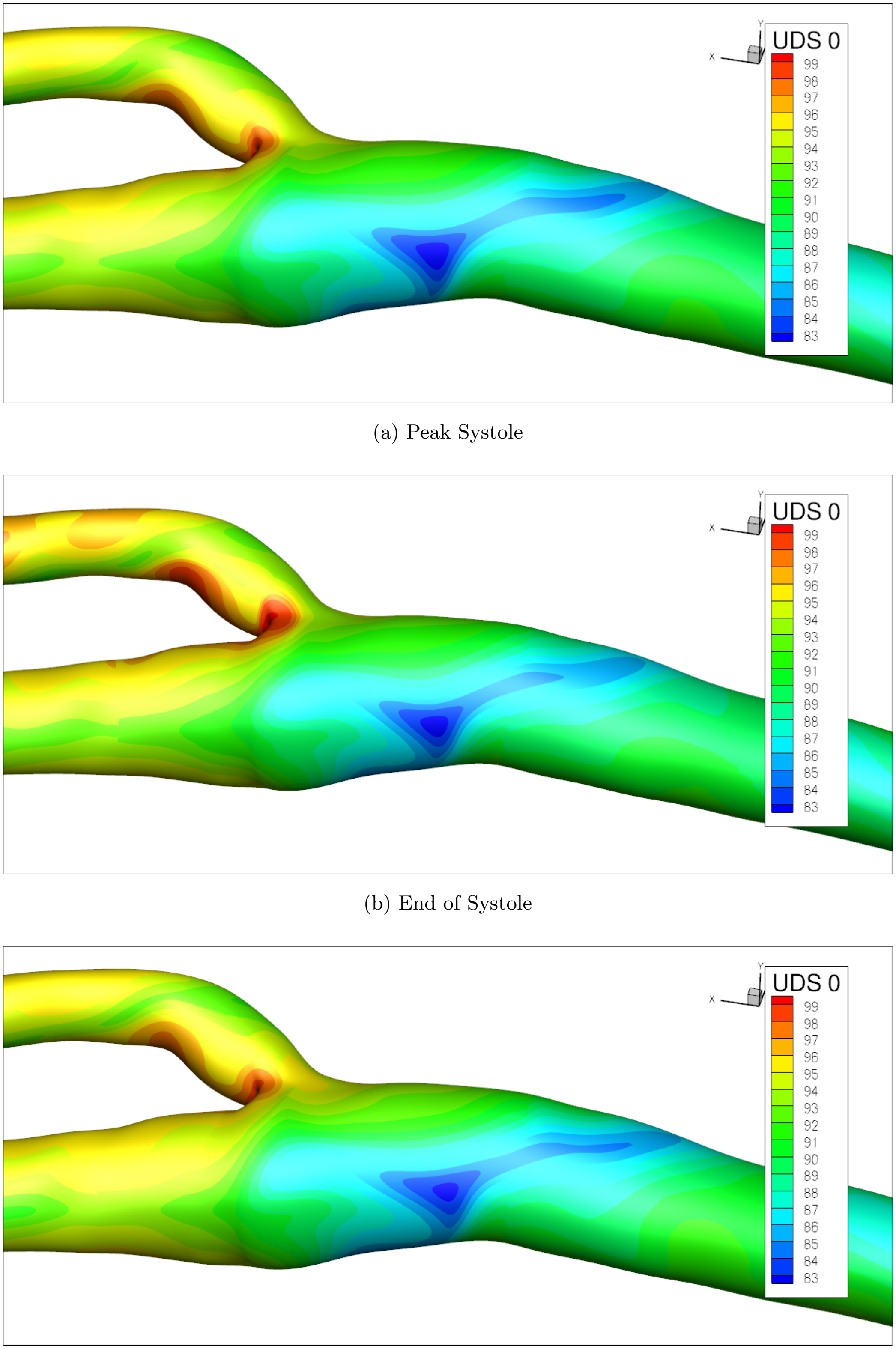
The last mass transfer simulation was a “flush test”. The test is useful for identifying regions of long residence times and high recirculation. The test was initialized with a species concentration of 100 for the entire domain and the inlet was prescribed a uniform profile of 0. The process of how the CAB was “flushed” clean was observed. Figure 5.10 show different points in time for the test. The CCA is slowly flushed out as the inlet condition of 0 works its way through the boundary layer. The domain interior is flushed much more quickly due to the higher velocities. The bifurcation apex is cleared after one cardiac cycle due to the impinging flow from the CCA, see Figure 5.10a. The secondary vortices formed by the bifurcation and pulsatile flow quickly clear the ECA and ICA wall regions. The region upstream of the carotid sinus is cleared after the end of the fifth systole, see Figure 5.11. Most importantly, the carotid sinus is still not cleared after six systoles, see Figure 5.11. A very slow process by which the vortex generated at the bifurcation during deceleration after systole entrains a “whiff” of high concentration fluid from the recirculation bubble in the carotid sinus is the only mechanism for convective exchange between the core flow and the



(a) Lateral view

(b) Posterior view

Figure 5.5: Scalar concentration at arterial wall; constant flux out of fluid

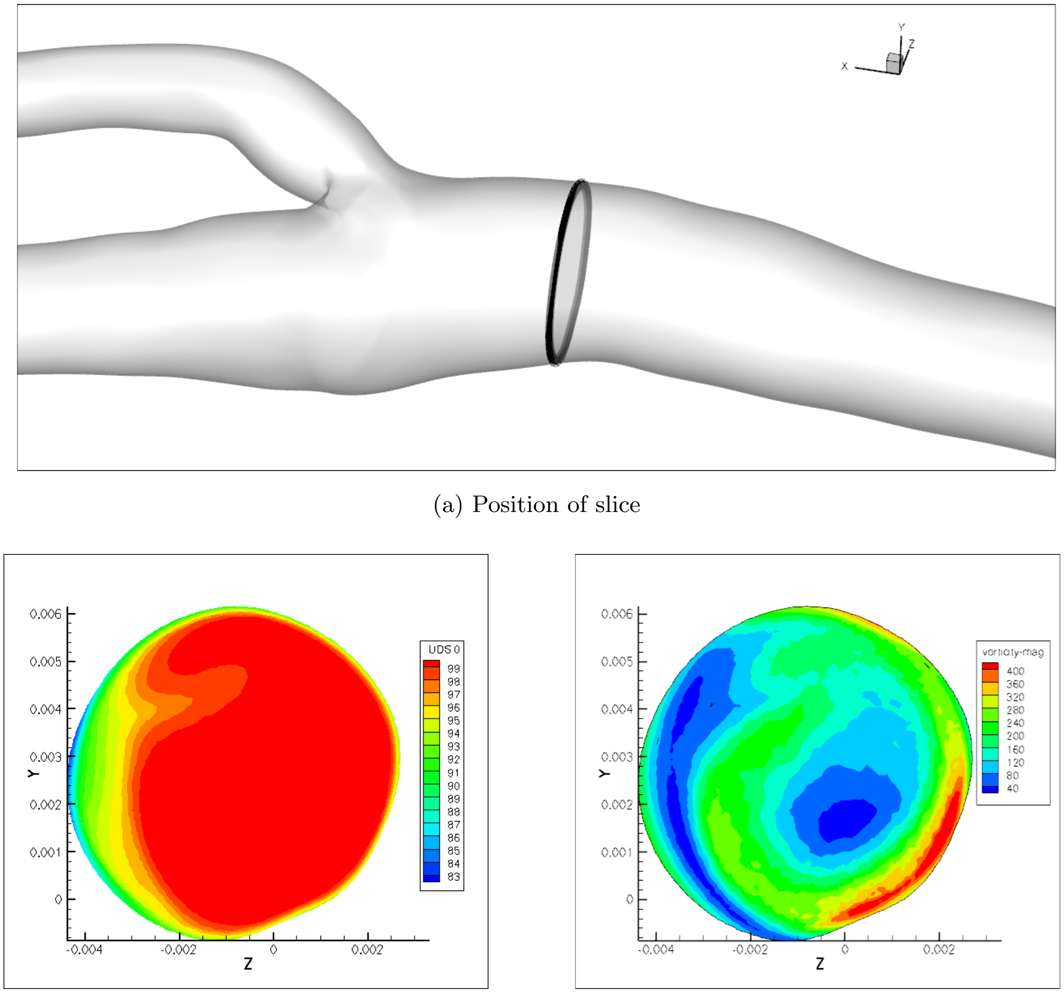


(a) Peak Systole

(b) End of Systole

(c) Rest

Figure 5.6: Scalar concentration at arterial wall; Comparison between different times within the cardiac cycle

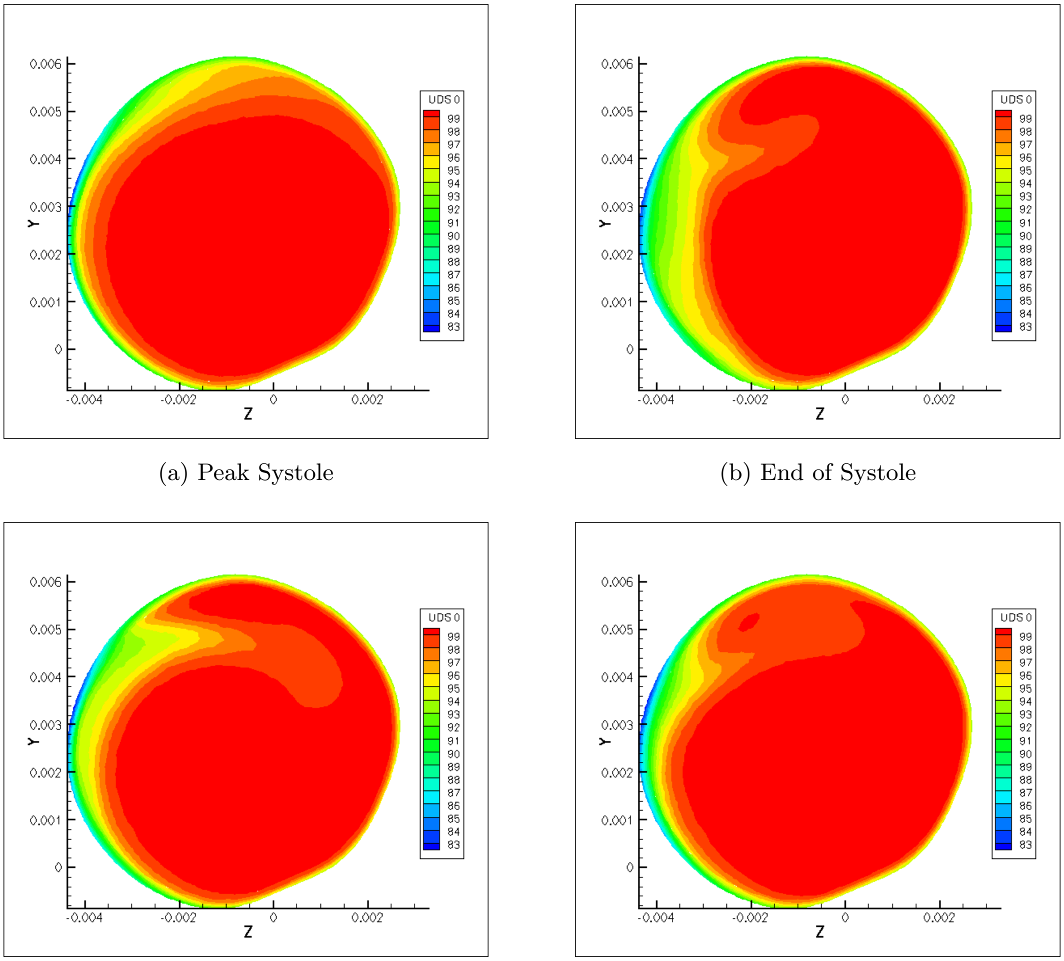


(a) Position of slice

(b) Scalar concentration

(c) Vorticity magnitude

Figure 5.7: Comparison of scalar concentration and vorticity magnitude in carotid sinus



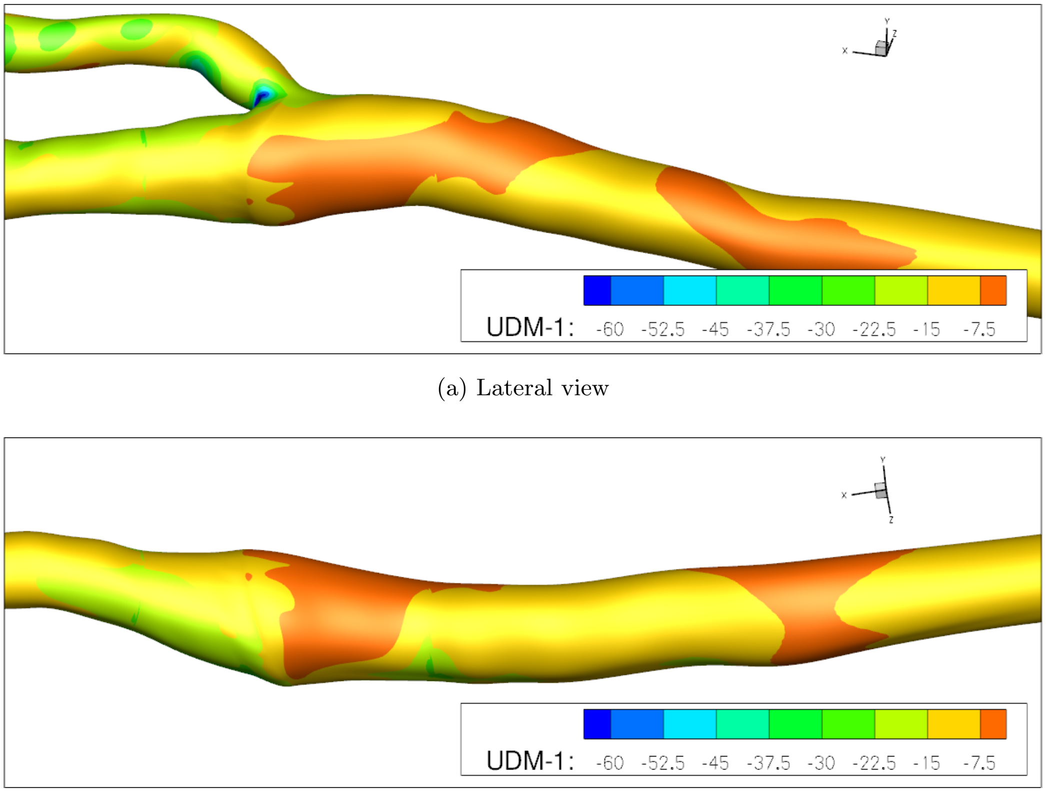
(a) Peak Systole

(b) End of Systole

(c) Peak Diastole

(d) Rest

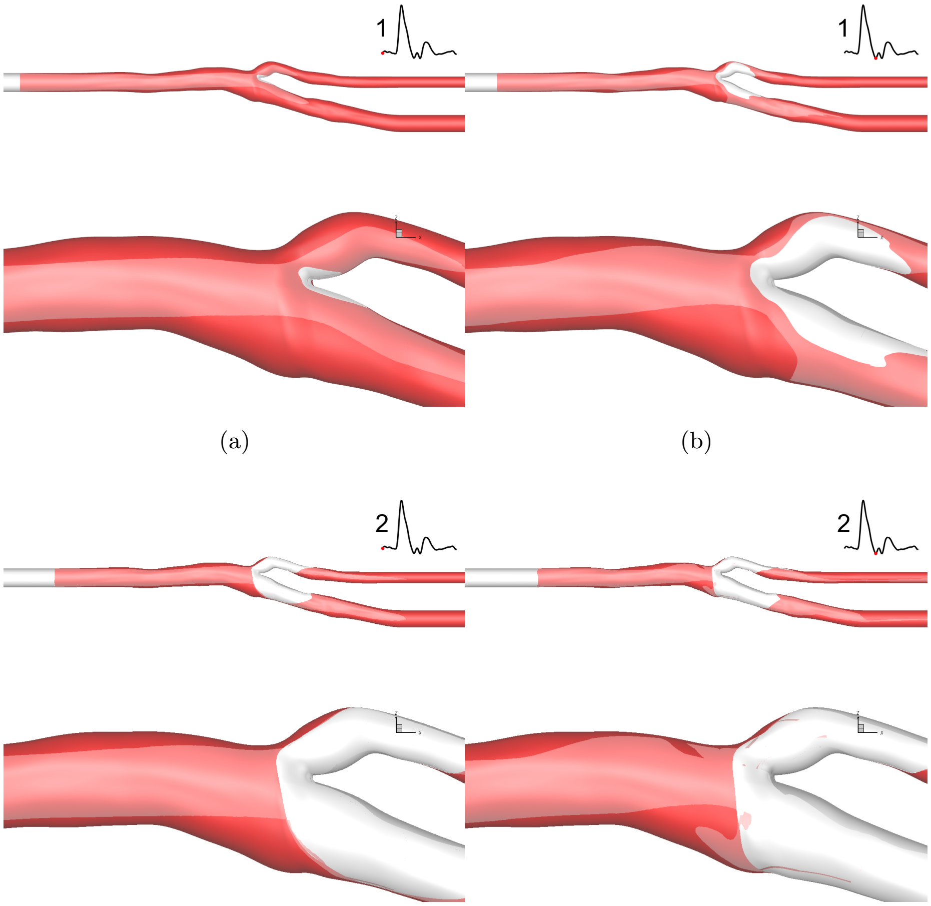
Figure 5.8: Scalar concentration in carotid sinus for various times within cardiac cycle


Y
Z
X
UDM-1: -60 -52.5 -45 -37.5 -30 -22.5 -15 -7.5
Y
X
Z
UDM-1: -60 -52.5 -45 -37.5 -30 -22.5 -15 -7.5

(a) Lateral view

(b) Posterior view

Figure 5.9: Scalar flux at arterial wall; constant concentration at arterial wall



(a)

(b)

(c)

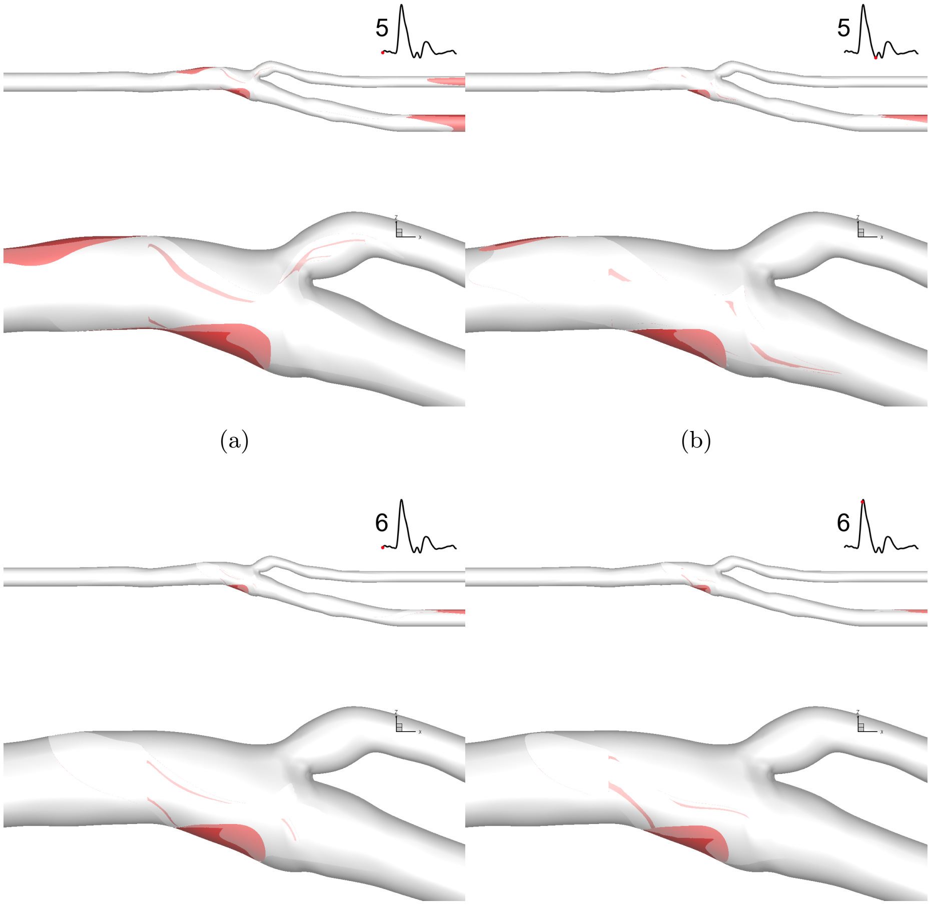
(d)

Figure 5.10: Flushing of carotid artery over six cardiac cycles

sinus. The time of fluid residence in that region is, therefore, orders of magnitude longer than the rest of the arterial lumen. This has implications towards atherosclerosis in that species within the carotid sinus have orders of magnitude longer times to enter the arterial wall, or to create adverse environments for endothelial cells.

5.4 Thresholding

A simple thresholding technique was implemented to highlight discrepancies between the shear and mass transport. Figure 5.12 compared the result of WSS $<$ 0.25 Pa (shown in



(a)

(b)

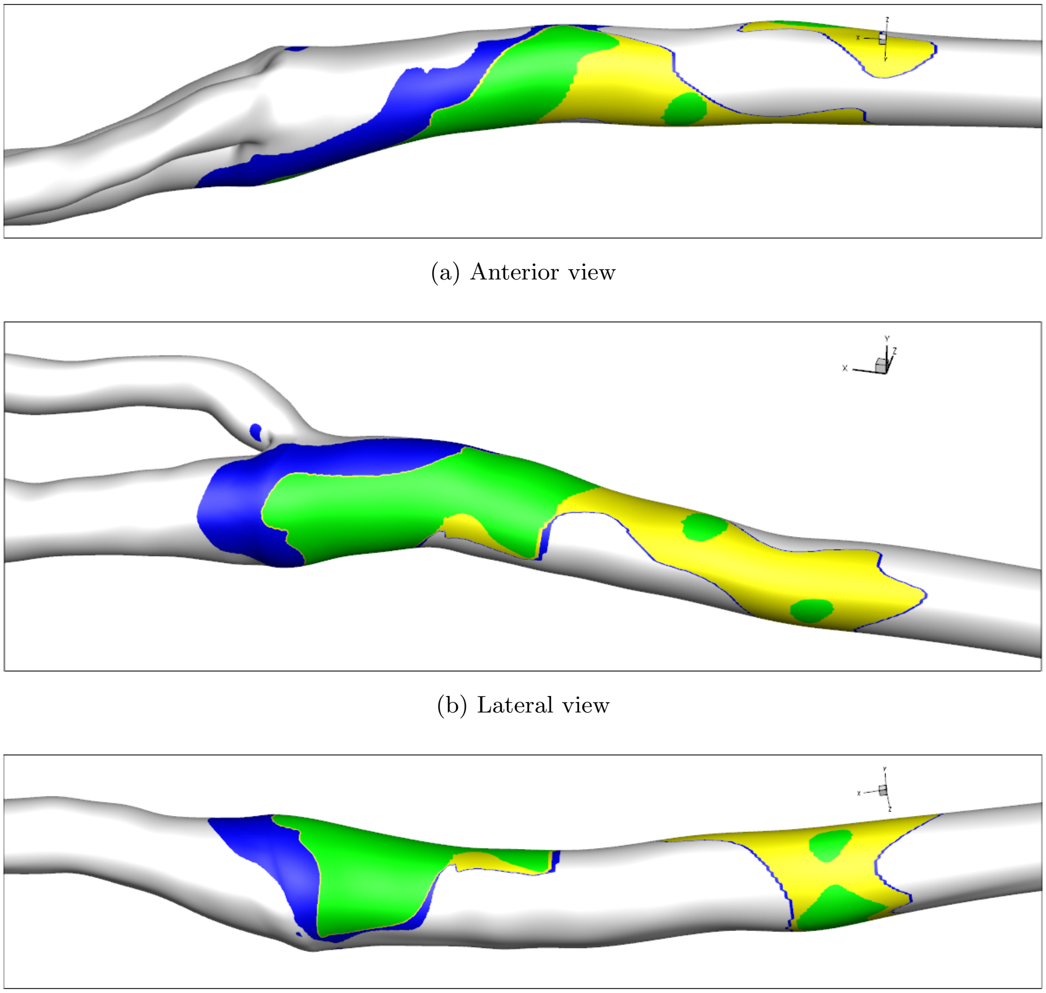
(c)

(d)

Figure 5.11: Continuation of Figure 5.10

blue) and the region of reduced scalar concentration ($\phi < 90$), shown in yellow. The region where the two thresholds overlap is shown in green. There is a large amount of overlap between the two, but some differences can be seen. The inhibited mass transfer starts upstream of the carotid sinus while the low WSS extends further into the ICA.

This technique is highly sensitive to the values used in the cutoff for the various mass transport or shear parameters. However, the analysis is a useful way to combine with future results from cell culture studies that may give more insights on specific thresholds of interest. One example would be the WSS at which endothelial cells become well aligned or values bellow which endothelial permeability is significantly enhanced. This will allow for isolation of regions prone to atherosclerotic processes from the rest of the domain.



(a) Anterior view

(b) Lateral view

(c) Posterior view

Figure 5.12: Threshold of WSS and Scalar concentration; Blue: WSS $< 0.25Pa$; Yellow: $\phi < 90$; Green: The two thresholds overlap

Chapter 6

CONCLUSIONS AND FUTURE WORK RECOMMENDATIONS

The fluid flow and scalar transport within the carotid artery bifurcation was solved for patient specific geometries and realistic velocity waveforms. Simplified scalar transport boundary conditions were used due to the limited quantitative knowledge of the species transport through the arterial walls. The simplified boundary conditions allowed for first order estimates of the flow effects of species transport.

Within the carotid sinus region, high levels of OSI, low TAWSS, and inhibited scalar transport was found. The mass transfer regions correlate highly with the regions of high OSI except at the point of flow reattachment within the internal carotid artery, where the OSI would give a false positive. In future studies, the OSI could be used as an approximation for regions of inhibited mass transfer (as long as the regions of reattachment are not included) instead of resolving the mass transfer gradients near the arterial wall. This would significantly reduce the computational requirements for future qualitative mass transfer studies.

The scalar transport was found to vary little throughout the cardiac cycle, unlike the WSS. This is attributed to the high Schmidt numbers found in cardiovascular flows which leads to long diffusion times. Future quantitative mass transfer studies could implement steady state inflow and outflow conditions, which would reduce computational expense. It must be noted that the parameter range explored in this thesis was limited. A wider range of Reynolds and Schmidt numbers must be explored before making a definitive statement regarding the scalar variation at the wall.

A flush test was completed to highlight long residence times and poor mass transfer between the separated region and the rest of the flow. The main mode of scalar transfer into the sinus region occurred during the end of systole when a vortex would sweep fluid from the core of the flow into the separated region. This, coupled with the long diffusion times, inhibits the mass transfer within the carotid sinus.

6.1 Future Work

There is still a large number of unknowns for the mass transfer process within cardiovascular flows. One of the main limitations of this thesis was the small parameter space investigated. Expanding the parameter space would allow for more concrete statements on the scalar variability at the wall as mentioned earlier and also confirm the methods of reducing computational expense for qualitative and quantitative mass transfer studies. Introducing complex species boundary conditions at the arterial wall is another region that requires more work. How the arterial wall reacts to shear stress and species concentrations potentially has big implications for macromolecular transport into the arterial wall and overall endothelial cell health. One potential suggestion would be to introduce a mass-spring-damper system to allow for the wall flux to change based on WSS or species concentration at the wall.

The species within the blood are constantly reacting with each other. The reaction-rates and source terms are not well defined at the time of writing this thesis, but adding reaction kinetics to mass transfer studies of ROS or other reactive species would improve the accuracy of flow side mass transfer effects.

Larger macromolecules such as LDL could also be modeled using particle tracking instead of scalar transport. The larger molecules have increasing larger Schmidt numbers, causing the numerical requirements to increase. Using particle tracking would reduce the computational resources required although it may require the use of computationally expensive Monte-Carlo techniques. More research is required to determine the computational savings.

BIBLIOGRAPHY

[1] *ANSYS FLUENT 12.0 User's Guide.*

[2] *Transport Phenomena.* John Wiley and Sons, 1960.

[3] *Numerical Methods with MATLAB.* Prentice Hall, 2000.

[4] L Antiga, B Ene-Iordache, L Caverni, G Paolo Cornalba, and A Remuzzi. Geometric reconstruction for computational mesh generation of arterial bifurcations from ct angiography. *Computerized Medical Imaging and Graphics*, 26(4):227–235, 2002.

[5] L Antiga, B Ene-Iordache, and A Remuzzi. Computational geometry for patient-specific reconstruction and meshing of blood vessels from mr and ct angiography. *IEEE Transactions on Medical Imaging*, 22(5):674–684, 2003.

[6] American Heart Association. Heart disease and stroke statistics - 2010 update. Technical report, American Heart Association, 2010.

[7] Yong Boo and Hanjoong Jo. Flow-dependent regulation of endothelial nitric oxide synthase: role of protein kinases. *AJP - Cell Physiology*, 285(3):C499, Sep 2003.

[8] CG Caro, JM Fitz-Gerald, and RC Schroter. Atheroma and arterial wall shear observation, correlation and proposal of a shear dependent mass transfer mechanism for atherogenesis. *Proceedings of the Royal Society of London. Series B, Biological Sciences*, 177(1046):109–133, 1971.

[9] S Chien. Effects of disturbed flow on endothelial cells. *Ann Biomed Eng*, 36(4):554–562, 2008.

[10] L Cohen. Deformable surfaces and parametric models to fit and track 3 d data. *PROC IEEE INT CONF SYST MAN CYBERN*, Jan 1996.

[11] C Ethier. Computational modeling of mass transfer and links to atherosclerosis. *Ann Biomed Eng*, Jan 2002.

[12] Leopold Grinberg and George Em Karniadakis. Outflow boundary conditions for arterial networks with multiple outlets. *Ann Biomed Eng*, 36(9):1496–1514, Jan 2008.

[13] MR Kaazempur-Mofrad, AG Isasi, HF Younis, RC Chan, DP Hinton, G Sukhova, GM LaMuraglia, RT Lee, and RD Kamm. Characterization of the atherosclerotic carotid bifurcation using mri, finite element modeling, and histology. *Ann Biomed Eng*, 32(7):932–946, 2004.

[14] David N Ku, DP Giddens, CK Zarins, and S Glagov. Pulsatile flow and atherosclerosis in the human carotid bifurcation. positive correlation between plaque location and low oscillating shear stress. *Arteriosclerosis, Thrombosis, and Vascular Biology*, 5:293–302, Jul 1985.

[15] DN Ku and DP Giddens. Pulsatile flow in a model carotid bifurcation. *Arteriosclerosis, Thrombosis, and Vascular Biology*, 3(1):31, Jan 1983.

[16] Sang-Wook Lee, Luca Antiga, J Spence, and David Steinman. Geometry of the carotid bifurcation predicts its exposure to disturbed flow. *Stroke*, 39(8):2341, Aug 2008.

[17] D. F. Leotta, Jean F. Primozich, C. M. Lowe, L. N. Karr, Robert O. Bergelin, Kirk W. Beach, and R. Eugene Zierler. Measurement of anastomosis geometry in lower extremity bypass grafts with 3-d ultrasound imaging. *Ultrasound Med Biol*, 31(10):1305–1315, 2005.

[18] Daniel F. Leotta, Jean F. Primozich, Kirk W. Beach, Robert O. Bergelin, and Jr. D. Eugene Strandess. Serial measurement of cross-sectional area in peripheral vein grafts using three-dimensional ultrasound. *Ultrasound Med Biol*, 27(1):61–68, 2001.

[19] M Levesque and R Nerem. The elongation and orientation of cultured endothelial cells in response to shear stress. *Journal of Biomechanical Engineering*, 107, Jan 1985.

[20] J Milner, J Moore, B Rutt, and D Steinman. Hemodynamics of human carotid artery bifurcations: computational studies with models reconstructed from magnetic resonance imaging of normal subjects. *Journal of Vascular Surgery*, Jan 1998.

[21] Keri R Moyle, Luca Antiga, and David A Steinman. Inlet conditions for image-based cfd models of the carotid bifurcation: Is it reasonable to assume fully developed flow? *Journal of Biomechanical Engineering*, 128:371–379, May 2006.

[22] A Mügge. The role of reactive oxygen species in atherosclerosis. *Zeitschrift für Kardiologie*, 87(11):851, 1998.

[23] T.J Pedley, R.C Schroter, and M F Sublow. Flow and pressure drop in systems of repeatedly branching tubes. *J Fluid Mech*, 46(2):365–383, 1971.

[24] M Pingping, L Xianming, and D Ku. Convective mass transfer at the carotid bifurcation. *Journal of biomechanics*, Jan 1997.

[25] Medline Plus. Stroke. http://www.nlm.nih.gov/medlineplus/ency/article/000726.htm, 2010.

[26] R Siegel, E M Sparrow, and T Hallman. Steady laminar heat transfer in a circular tube with presecribed wall heat flux. *Applied Scientific Research*, A7:386–392, Mar 1958.

[27] C Taylor, T Hughes, and C Zarins. Finite element modeling of three-dimensional pulsatile flow in the abdominal aorta: relevance to atherosclerosis. *Ann Biomed Eng*, Jan 1998.

[28] Tedgui Olivier Tricot, Ziad Mallat, Christophe Heymes, Joël Belmin, Guy Lesèche, and Alain. Relation between endothelial cell apoptosis and blood flow direction in human atherosclerotic plaques. *Circulation*, 101:2450–2453, May 2000.

[29] Philip Tsao, Ricardo Buitrago, Jason Chan, and John Cooke. Fluid flow inhibits endothelial adhesiveness: Nitric oxide and transcriptional regulation of vcam-1. *Circulation*, 94(7):1682, Oct 1996.

[30] JR Womersley. Method for the calculation of velocity, rate of flow and viscous drag in arteries when the pressure gradient is known. *The Journal of physiology*, 127(3):553, 1955.

[31] C Xu and J Prince. Snakes, shapes, and gradient vector flow. *IEEE transactions on image processing*, Jan 1998.

[32] C Zarins, D Giddens, and B Bharadvaj. Carotid bifurcation atherosclerosis. quantitative correlation of plaque localization with flow velocity profiles and wall shear stress. *Circ. Res.*, 53, 1983.

Appendix A

MATLAB SCRIPTS

ultrasound3.m

Main geometry pre-processing file. Imports data, estimates centerlines, rotates data, exports to SolidWorks.

```
%
%   Ultrasound Data Analysis File
%       —reads mesh file  "filename.msh"
%       —determines centriod of contours
%       —finds equation for axial vector
%       —determines rotation matrix to place the inlet
%           parrellel with x—axis
%       —shifts contours to place inlet centriod at the origin
%   Required Files:
%       align_x.m
%       centriod.m
%       hydraulic_dia.m
%       plotcountours.m  — if you want to look at the resulting curves
close all;
clear all;
clc;


%% Read mesh file
% — 3D coordinates of contours

files = 3;
cd Carotid_2
fname=(['/sccor_car01b_cca_s.msh';'/sccor_car01b_eca_s.msh';...
    '/sccor_car01b_ica_s.msh']);


for h = 1:files
%     disp('Select mesh file');
%     [mfile, ipath] = uigetfile('*.msh', 'Select mesh file');


    %+++++++++++++++++++++++++++++++++++++++++++++++++++++++
    % read re—sampled contours
%         fname = mfile;


    % file format:
    %   [number of contours]
    %   [image number, number of points]
    %   [x y z]

%         fid = fopen(fname);
        fid = fopen(fname(h,:));
        [n_image(h,1)] = fscanf(fid, '%i', 1);
        im_list = zeros(n_image(h,1),1);

        for i = 1:n_image(h,1)
            [a] = fscanf(fid, '%i %i', 2);
            im_list(i) = a(1);
            n_lines = a(2);

            if i == 1 && h == 1
                poly = zeros(n_lines+1,3,n_image(h,1),files);
            end

            for j = 1:n_lines
                [b] = fscanf(fid, '%f %f %f', 3);
                poly(j,:,i,h) = b';
            end

            poly(n_lines+1,:,i,h) = poly(1,:,i,h);
        end
        fclose(fid);
end
clear fid
clear fname
clear a
clear ans
clear b

cd('..')
%% ++++++++++++++++++++++++++++++++++++++++++++++++++++++


%   Next the geometry will be shifted to have the first curve centered on
%   the origin. It also finds the centriod of each contour and stores them
%   in an array.

centriods = zeros(n_image(h,1),3);
origin = centriod(poly(:,:,1,1));

for h = 1:files
    for i = 1:n_image(h,1)
        for j = 1:n_lines+1
            poly(j,:,i,h) = poly(j,:,i,h) - origin;
        end

        if h == 1
            centriods(i,:) = centriod(poly(:,:,i,1));
        end
    end
end

centriods(1,:) = [0,0,0];

```

```
% Cubic Spline estimation on the centerlines
pp = spline(centriods(:,1),transpose([centriods(:,2), centriods(:,3)]));

% Calculating Derivative of centerline estimation
ppder = pp;
ppder.coefs = ppder.coefs*diag([3 2 1],1);

%xx = centriods(9);
xx = centriods(1);
yy = ppval(pp,xx);
yyder = ppval(ppder,xx);



%   Finding the Rotation Matrix required to align the normal vector of the
%   curve fit at the origin with the x—axis

RM = align_x([1; yyder]);
%%
%   Rotating the Data Set
poly_shift = zeros(n_lines+1,3,n_image(h,1),files);
centriods_shift = zeros(n_image(h,1),3,files);

for h = 1:files
    for i = 1:n_image(h,1)
        poly_shift(:,:,i,h)=transpose(RM*transpose(poly(:,:,i,h)));
        centriods_shift(i,:,h) = centriod(poly_shift(:,:,i,h));
    end
end


centriods_shift(1,:,1) = [0,0,0];

curves = 4;
x = zeros(curves,3,files);
dia = zeros(files,1);
```

```
axial = zeros(3,files);
for h = 1:files
    for i = 1:(curves+1)
        if h ≠ 1
            x(i,1,h) = centriods_shift(i,1,h);
            x(i,2,h) = centriods_shift(i,2,h);
            x(i,3,h) = centriods_shift(i,3,h);
        else
            % slice = n_image(h);
            slice = 5;
            x(i,1,h) = centriods_shift(slice—curves+i—1,1,h);
            x(i,2,h) = centriods_shift(slice—curves+i—1,2,h);
            x(i,3,h) = centriods_shift(slice—curves+i—1,3,h);
        end
    end

    if h == 1
        dia(h) = hydraulic_dia(poly_shift(:,:,1,1),...
            [1,x(:,1,h)\x(:,2,h),x(:,1,h)\x(:,3,h)]);
    else
        dia(h) = hydraulic_dia(poly_shift(:,:,5,h),...
            [1,x(:,1,h)\x(:,2,h),x(:,1,h)\x(:,3,h)]);
    end

    axial(:,h) = [1,x(:,1,h)\x(:,2,h),x(:,1,h)\x(:,3,h)];
end

%*************************************************************

  Exporting to SolidWorks

  Adjust the code to keep number of curves below 30 per macro file

for h = 1:files
    %++++++++++++++++++++++++++++++++++++++++++++++++++++++
    output the data in a way Solidworks can read.
    file=sprintf(['ultrasound_' num2str(h) '.swp']);
    fid2=fopen(file,'w');
%++++++++++++++++++++++++++++++++++++++++++++++++++
    Beginging of Solidworks Macro

    fprintf(fid2,'Dim swApp As Object\n');
    fprintf(fid2,'Dim Part As Object\n');
    fprintf(fid2,'Dim SelMgr As Object\n');
    fprintf(fid2,'Dim boolstatus As Boolean\n');
    fprintf(fid2,'Dim longstatus As Long, longwarnings As Long\n');
    fprintf(fid2,'Dim Feature As Object\n');
    fprintf(fid2,'Sub main()\n\n');
    fprintf(fid2,'Set swApp = Application.SldWorks\n\n');
    fprintf(fid2,'Set Part = swApp.ActiveDoc\n');
    fprintf(fid2,'Set SelMgr = Part.SelectionManager\n');
%++++++++++++++++++++++++++++++++++++++++++++++++++++++



file format:
    [number of contours]
    [image number, number of points]
    [x y z]

for i = 1:n_image(h)

    fprintf(fid2,'Part.InsertCurveFileBegin\n');

    for j = 1:n_lines
        fprintf(fid2,['Part.InsertCurveFilePoint '...
            num2str(poly_shift(j,1,i,h),'%8.4f') ', '...
            num2str(poly_shift(j,2,i,h),'%8.4f') ', '...
            num2str(poly_shift(j,3,i,h),'%8.4f') '\n']);
    end

    fprintf(fid2,['Part.InsertCurveFilePoint '...
        num2str(poly_shift(1,1,i,h),'%8.4f') ', '...
```

```
            num2str(poly_shift(1,2,i,h),'%8.4f') ', '...
            num2str(poly_shift(1,3,i,h),'%8.4f') '\n']);

        fprintf(fid2,'Part.InsertCurveFileEnd\n');

    end

        fprintf(fid2,'End Sub\n');
end

fclose('all');

%***********************************************************
```

align_x.m

File used to align CCA inlet with X-axis.

```
%
% Align with X-axis
%

function R = align_x(A)
%   ALIGN-X
%   ALIGN-X(A) outputs the rotation matrix required to align the inputed
%   vector with the x-axis.  This done by completeing two rotations.  The
%   first aligns the vector on the xy-plane by rotating about the z-xis by
%   the Yaw angle.  The second rotation aligns the vector on the xz-plane
%   by rotating about the y-axs by the Pitch angle.
%
%   The vector, A, must have the size: [3,1]

%   Calculating the Yaw angle and rotating the vector
yaw = 2*pi - atan2(A(2),A(1));
R_z = [cos(yaw),-sin(yaw),0;sin(yaw),cos(yaw),0;0,0,1];
A2 = R_z*A;

```

```matlab
%   Calculating Pitch angle
pitch = atan2(A2(3),A2(1));
R_y = [cos(pitch),0,sin(pitch);0,1,0;-sin(pitch),0,cos(pitch)];

%   Outputing the Rotation matrix
R = R_y*R_z;
end
```

centriod.m

Calculates the centroid of a given data set

```matlab
%
% Centriod Calculation
% - 3D coordinates of contours

function r = centriod(A)
%   CENTRIOD Matrix centriod.
%   CENTRIOD(A) provides the centriod of the points in stored in the
%   matrix.  The centriod is calculated assuming "m" points and "n"
%   dimensions in a m x n matrix.
[m,n] = size(A);

if A(1,:) == A(m,:)
    r = (sum(A) - A(1,:))/(m - 1);
else
    r = sum(A)/m;
end
end
```

circle3.m

Creates a circle in any 3D orientation and size. Used in creating flow extensions.

```
function circ = circle3(center,radius,points,normal, distance)
%
%  circle(v,r,n)
%
%   Creates a circle with specified radius, center and orientation
%
%     c - center (either 2 or 3 dimensional vector)
%     d - normal vector
%     r - radius; default = 1
%     n - number of plot points; default = 20
%

th = linspace(0,2*pi,points);
rr = radius*ones(1,points);
[xx,yy] = pol2cart(th,rr);

normal = normal/norm(normal);
a = [0;0;1];

axis = cross(a,normal);
axis = axis/norm(axis);
if norm(axis) ≠ 0
    angle = atan2(1,dot(a,normal));
    c = dot(a,normal);
    cc = 1-c;
    s = sin(angle);

    R = [axis(1)^2 + (1-axis(1)^2)*c, axis(1)*axis(2)*cc - axis(3)*s,...
        axis(1)*axis(3)*cc + axis(2)*s;
         axis(1)*axis(2)*cc + axis(3)*s, axis(2)^2 + (1-axis(2)^2)*c,...
         axis(2)*axis(3)*cc - axis(1)*s;
         axis(1)*axis(3)*cc - axis(2)*s, axis(2)*axis(3)*cc + axis(1)*s,...
         axis(3)^2 + (1—axis(3)^2)*c];

    circ = R*[xx; yy; zeros(1,points)];
else
    circ = [xx; yy; zeros(1,points)];
end

circ(1,:) = circ(1,:) + center(1) + distance*normal(1);
circ(2,:) = circ(2,:) + center(2) + distance*normal(2);
circ(3,:) = circ(3,:) + center(3) + distance*normal(3);

end
```

hydraulic_dia.m

Calculates the hydraulic diameter of a given contour projected onto a given plane.

```
%
% Calcuates the Hydraulic Diameter
%

function D_h = hydraulic_dia(A,B)
%   HYDRAULIC_DIA
%   HYDRAILIC_DIA(A,B) outputs the hyrdaulic diameter for the given contour
%   data (A).  The vector (B) is the axial vector to calculate the D_h
%   from.  A projection of the contour data onto the axial plane is
%   completed before calculating the area and perimeter to find the D_h.
%
%   The vector, B, must have the size: [3,1]

%   For testing purposes

[m,n] = size(B);
if (m ==1 && n == 3)
    B = B';
end

```

```
[m,n] = size(B);

if (m == 3 && n == 1)

    %   Centering the countour at the Origin
    origin = centriod(A(:,:));
    for j = 1:length(A)
        A(j,:) = A(j,:) - origin;
    end

    %   The Rotation matrix
    R = align_x(B);

    %   Rotating the contour to align with the X-axis
    A_rotated(:,:)=transpose(R*transpose(A(:,:)));

    %   Projecting onto the Y-Z plane
    A_projected(:,:)=A_rotated(:,:)*[0,0,0;0,1,0;0,0,1];

    %   Calculating the Area and Perimeter of the Curve
    area = polyarea(A_projected(:,2),A_projected(:,3));

    points = length(A);
    lengths = zeros(24,1);

    for i = 1: points - 1
        segment = A_projected(i,:) - A_projected(i+1,:);
        lengths(i) = norm(segment);
    end

    perimeter = sum(lengths);

    %   Hydraulic Diameter
    D_h = 4*area/perimeter;
else
    error('Axial Vector,B, must be a 3x1 or 1x3 vector')
```

```
end
```

plotcontours2.m

Used for plotting results of rotation, flow extensions, etc.

```
%   Plot the centriod line and the curves
close all;
hold on;
grid on;
%     for h = 1:files
%         for i = 1:n_image(h,1)
%          plot3(poly(:,1,i,h),poly(:,2,i,h),poly(:,3,i,h),'-r');
%         end
%     end

% Plotting CCA countours

        CCAs = 9;

        for i = 1:n_image(1,1)
            plot3(poly_shift(:,1,i,1),poly_shift(:,2,i,1),...
                poly_shift(:,3,i,1),'c--');
            daspect([1 1 1])
        end

        for i = CCAs:n_image(1,1)
            plot3(poly_shift(:,1,i,1),poly_shift(:,2,i,1),...
                poly_shift(:,3,i,1),'b-');
        end

 % Plotting ECA countours

        ECAe = 4;

        for i = 1:n_image(2,1)
            plot3(poly_shift(:,1,i,2),poly_shift(:,2,i,2),...
                poly_shift(:,3,i,2),'c--');
        end

        for i = 1:ECAe
            plot3(poly_shift(:,1,i,2),poly_shift(:,2,i,2),...
                poly_shift(:,3,i,2),'b-');
        end


% Plotting ICA countours

        ICAe = 12;

        for i = 1:n_image(3,1)
            plot3(poly_shift(:,1,i,3),poly_shift(:,2,i,3),...
                poly_shift(:,3,i,3),'c--');
            daspect([1 1 1])
        end

        for i = 1:ICAe
            plot3(poly_shift(:,1,i,3),poly_shift(:,2,i,3),...
                poly_shift(:,3,i,3),'b-');
            daspect([1 1 1])
        end


%% Calculating Centerlines of CCA and plotting it
CCA_pp = spline(centriods_shift(1:end-2,1,1),...
    transpose([centriods_shift(1:end-2,2,1),...
    centriods_shift(1:end-2,3,1)]));

CCA_ppder = CCA_pp;
CCA_ppder.coefs = CCA_ppder.coefs*diag([3 2 1],1);

CCA_xx = 0:0.01:35;
CCA_yy = ppval(CCA_pp,CCA_xx);
```

```
CCA_yyder =  ppval(CCA_ppder,CCA_xx(1:200:end));

plot3(CCA_xx,CCA_yy(1,:),CCA_yy(2,:));
quiver3(CCA_xx(1:200:end),CCA_yy(1,1:200:end),...
    CCA_yy(2,1:200:end),1,CCA_yyder(1,:),CCA_yyder(2,:),0.5);


% Calculating Centerlines of ECA and plotting it
ECA_pp = spline(centriods_shift(1:end-2,1,2),...
    transpose([centriods_shift(1:end-2,2,2),...
    centriods_shift(1:end-2,3,2)]));

ECA_ppder = ECA_pp;
ECA_ppder.coefs = ECA_ppder.coefs*diag([3 2 1],1);

ECA_xx = 40:0.01:60;
ECA_yy = ppval(ECA_pp,ECA_xx);
ECA_yyder = ppval(ECA_ppder,ECA_xx(1:200:end));

plot3(ECA_xx,ECA_yy(1,:),ECA_yy(2,:));
quiver3(ECA_xx(1:200:end),ECA_yy(1,1:200:end),...
    ECA_yy(2,1:200:end),1,ECA_yyder(1,:),ECA_yyder(2,:),0.5);


% Calculating Centerlines of ICA and plotting it
ICA_pp = spline(centriods_shift(1:end-2,1,3),...
    transpose([centriods_shift(1:end-2,2,3),...
    centriods_shift(1:end-2,3,3)]));
ICA_ppder = ICA_pp;
ICA_ppder.coefs = ICA_ppder.coefs*diag([3 2 1],1);

ICA_xx = 40:0.01:60;
ICA_yy = ppval(ICA_pp,ICA_xx);
ICA_yyder = ppval(ICA_ppder,ICA_xx(1:200:end));

plot3(ICA_xx,ICA_yy(1,:),ICA_yy(2,:));
```

```
quiver3(ICA_xx(1:200:end),ICA_yy(1,1:200:end),...
    ICA_yy(2,1:200:end),1,ICA_yyder(1,:),ICA_yyder(2,:),0.5);


%% Adding Flow extention
center = centriods_shift(CCAs,:,1);
normal = [1 ; ppval(CCA_ppder,center(1))];
dia = hydraulic_dia(poly_shift(:,:,CCAs,1),normal)

CCAinlet = circle3(center ,dia/2 , 24, normal, -dia*3);
plot3(CCAinlet(1,:),CCAinlet(2,:),CCAinlet(3,:));
hydraulic_dia(transpose(CCAinlet),normal)


center = centriods_shift(ECAe,:,2);
normal = [1 ; ppval(ECA_ppder,center(1))];
dia = hydraulic_dia(poly_shift(:,:,ECAe,2),normal)

ECAoutlet = circle3(center ,dia/2 , 24, normal, dia*3);
plot3(ECAoutlet(1,:),ECAoutlet(2,:),ECAoutlet(3,:));
hydraulic_dia(transpose(ECAoutlet),normal)


center = centriods_shift(ICAe,:,3);
normal = [1 ; ppval(ICA_ppder,center(1))];
dia = hydraulic_dia(poly_shift(:,:,ICAe,3),normal)


ICAoutlet = circle3(center ,dia/2 , 24, normal, dia*3);
plot3(ICAoutlet(1,:),ICAoutlet(2,:),ICAoutlet(3,:));
hydraulic_dia(transpose(ICAoutlet),normal)

%% Writing Flow Extentions curves to txt files

dlmwrite('CCAinlet.txt', transpose(CCAinlet)*1000, 'precision', '%.6f', ...
    'delimiter', '\t', 'newline', 'pc')
```

```

dlmwrite('ECAoutlet.txt', transpose(ECAoutlet)*1000, 'precision', '%.6f', ...
    'delimiter', '\t', 'newline', 'pc')

dlmwrite('ICAoutlet.txt', transpose(ICAoutlet)*1000, 'precision', '%.6f', ...
    'delimiter', '\t', 'newline', 'pc')
```

shooting.m

Shooting method used to determine eigenvalues and eigenfunctions in Chapter 3 and Appendix B

```
%%% Shooting Method for solving step in wall flux problem

%%  Initialization

clc
clear all;
close all;

L = 1;
tol = 10^(−8);  %   Tolerance level for shooting algorithm
col = ['y','m','c','r','b','g','k'];    %   colors for eigenfunctions

ΔX = 0.0001;
XSPAN = [ΔX:ΔX:L];   %   Span of Domain
phi_start = 1;

%%  System Parameters

Re = 400;  % Reynolds number based off of mean velocity
Sc = 100;  % Schmidt number
Pe = Re*Sc;% Peclet number

flux = 1;          % [scalar / m^2]
R    = 0.005615;     % [m]
```

```
rho  = 1000;        % [kg/m^3]
Te   = 100;         % entrance scalar value
visc = 2.63e-6;     % kinematic viscosity [m^2/s]

D = visc/Sc;        % scalar diffusivity [m^2/s]
k = D*rho;
Umax = 2*Re*visc/(2*R); % max velocity [m/s]

%% Main Program - Shooting with Secant
n = 10;                       % number of eigenvectors to find
epsl_start = 28;            % Initial guess for the eigenvalue
epsl_storage = zeros(1,n);  % Matrix to store converged eigenvalues
y_storage = zeros(L/ΔX,n);

for modes=1:n

      epsl=epsl_start;
      secant= zeros(2,2);

     for j=1:100
         BC=[1 0];  % Initial Conditions to start the shooting method
         [t,y]=ode45(@(t,y)shoot2(t,y,epsl),XSPAN,BC,10^(-8));

        if abs(y(end,2)) < tol
            norm=trapz(t,y(:,1).*y(:,1));
            epsl_storage(modes) = epsl;
            y(:,:)=y(:,:)/sqrt(norm);
            y_storage(:,modes) = y(:,1);
            break
        end

        if j>1
            secant(1,1) = secant(2,1);
            secant(2,1) = epsl;
            secant(1,2) = secant(2,2);
            secant(2,2) = y(end,2);

            epsl = epsl - (epsl-secant(1,1))/...
                (secant(2,2)-secant(1,2)+ eps)*secant(2,2);

        else
            secant(2,1) = epsl;
            secant(2,2) = y(end,2);
            epsl = ((4*(modes+1)+4/3)^2 - epsl)/10 + epsl;
        end
     end

     epsl_start= (4*(modes+1) + 4/3)^2;
    subplot(2,2,1)
    plot(t,y(:,1)); hold on
    title('Eigenfunctions')
end

%% Calculating Solution
axial_length = 10;     % dimensionless length of axial direction [z/D]
points = 100;
zeta_length = axial_length*2/Pe;
ZSPAN = [0:zeta_length/points:zeta_length];

[Z,X] = meshgrid(ZSPAN,XSPAN);

Theta_inf = -4*Z - X.^2 + 1/4*X.^4 + 7/24;


B = zeros(1, n);
Theta_d = zeros(length(XSPAN),points+1);

for i = 1:n
   B(i) = trapz(XSPAN,(-XSPAN.^2 + 1/4*XSPAN.^4 +7/24).*...
       transpose(y_storage(:,i)).*(1-XSPAN.^2).*XSPAN) / ...
          trapz(XSPAN,transpose(y_storage(:,i).^2).*((1-XSPAN.^2).*XSPAN));

   Theta_d = Theta_d + B(i)*exp(-epsl_storage(i)*Z).*...
       (y_storage(:,i)*ones(1,points+1));
end

Theta = Theta_inf - Theta_d;


T = Theta*flux*R/k + Te;

Zd = Z*Pe/2;


%%  Plotting Solution

subplot(2,2,3)
% contourf(X,Zd,real(T))
pcolor(X,Zd,real(T))
shading interp
title('Temperature Recontruction')


subplot(2,2,4)
plot(Zd(1,:),T(end,:))
title('Wall Temperature')

subplot(2,2,2)
plot(epsl_storage, '-o')
title('eigenvalues')
```

shoot2.m

This file contains the structure of system of ODEs to be solved by the shooting method in *shooting.m*.

```
    %% shoot2.m function


function rhs = shoot2(t,y,epsl)
rhs = [y(2) ; -(epsl*(1-t.^2)*y(1) + 1/t*y(2))];
```

Appendix B
SHOOTING METHOD

A shooting method was used to solve the Sturm-Liouville problem in Chapter 3. Equation 3.14a is reproduced here:

$$\frac{1}{\xi}\frac{\partial}{\partial\xi}\left(\xi\frac{\partial\Xi}{\partial\xi}\right)+c^2\left(1-\xi^2\right)\Xi=0 \tag{B.1a}$$

$$\frac{\partial\Xi}{\partial\xi}=0 \quad \text{for} \quad \xi=0,1 \tag{B.1b}$$

Shooting methods solve boundary value problems like the one described in Equation B.1 by transforming the problem to an initial value problem. This is done transforming the PDE found in Equation B.1 to a system of ODEs. Equation B.2 shows the result of the transformation.

$$\begin{pmatrix}\dot{\Xi}\\ \ddot{\Xi}\end{pmatrix}=\begin{pmatrix}0 & 1\\ -C_n^2(1-\xi^2) & \dfrac{-1}{\xi}\end{pmatrix}\begin{pmatrix}\Xi\\ \dot{\Xi}\end{pmatrix} \tag{B.2}$$

A time-stepping algorithm is then used to advance the problem from one boundary to the next. This provides the eigenfunction, Ξ_m, for a given eigenvalue C_n^2. Implementing shooting methods initial values for Ξ and $\dot{\Xi}$ and guesses for C_n^2 are required. The guess for Ξ is not important as the eigenfunctions calculated are normalized to make $\int_0^1 \Xi_n d\xi_n = 1$. $\dot{\Xi}$ is defined by the boundary conditions given in Equation B.1. The initial guess for C_n^2 was determined using the limiting value of C_n for large n given by $C_n = 4n + \frac{4}{3}$ [26]. A fourth order Runge-Kutta scheme was used to advance through the domain and a secant method was used to quickly converge on the correct value of C_n^2. There is a challenge in the discretization of ξ in that if 0 is included in the solved domain the shooting method will give an error due to a divide by zero error. An easy way to get around this is to use an extremely fine discretization for ξ and set $\dot{\Xi} = 0$ at the first step after zero. This will introduce a small error but it was not significant for the number of eigenvalues solved in this thesis. The scripts used in the shooting method are found in Appendix A under *shooting.m* and *shoot2.m*. The first 1000 eigenvalues where found.