

PATIENT-SPECIFIC COMPUTATIONAL  
MODELLING OF EMBOLIC STROKE

Thesis submitted for the degree of

Doctor of Philosophy

at the University of Leicester

by

David Antony Marshall

Department of Cardiovascular Sciences

University of Leicester

September 2017

The work on which this thesis is based is my own independent work, except  
where acknowledged.

David Marshall

September 2017

# Abstract

Embolic stroke occurs when arterial debris or thrombus detaches from the insides of the diseased arteries and moves through the bloodstream to block arteries supplying the brain.

Stroke is a major clinical problem but few researchers have attempted to model embolus transport. Existing methods available for modelling stroke have numerous strengths and limitations.

This aim of this thesis is to investigate approaches to modelling the transport of emboli through the cerebral vasculature with a view to performing patient-specific simulations toward a better understanding of the pathophysiology of stroke.

This thesis is divided into three sections. In section one I investigate flow models of the cerebral circulation, reviewing anatomy and computational blood flow modelling, before creating a 0D model of the circle of Willis, produce a flow phantom of the cerebral vasculature, segment and analyse patient MR angiograms, and apply the 0D model to patient anatomy.

In section two I discuss and review computational modelling of embolus transport, adapt a fractal embolus trajectory model and an embolus sizing algorithm to patient-specific data using measurements from MR angiograms, then discuss the results of patient measurements and the embolus trajectory simulations.

In section three I discuss applying a fluid dynamics method called smoothed-particle hydrodynamics (SPH) to modelling emboli in the blood, I review SPH models of blood flow, present feasibility studies for applying SPH to emboli, and finally create an in-house code for simulating emboli, which is applied to embolus migration in a tube.

Dedicated to my loving, supportive and endlessly patient partner George,  
without whom this thesis would still be an empty document.

# Acknowledgements

I will start by thanking my supervisor Dr Emma Chung for finding funding for me to start this PhD, and supporting me through the years. I would like to thank Dr Jim Hague, Dr Mark Horsfield and Prof. Walter Dehnen for their supervision and advice.

I want to thank my loving family for their help and support, my parents Sheryl and Tony, my sister Lisa, and my grandparents Rita, Lily, Derek and Charlie.

The early years of this PhD would have been far less enjoyable without my office buddies Nikil, Clément, Jamie and Preeya. I would also like to thank Bharti for her words of wisdom, Caroline for enjoyable conversation when I most needed it, Kumar for his excellent pool tips, Justyna for being a constant source of positivity, and Sarah, Nazia, Lizzie, Mike, Baris, Kunal, Sara, Angela and João for being all-round wonderful people who I am very glad to have met.

I would, once again, like to thank George for her support, motivation, and extensive planning, I couldn't have done it without you.

# Publications arising from this thesis

## Work published in peer-reviewed journals

Chung, E. M. L., Banahan, C., Patel, N., Janus, J., **Marshall, D.**, Horsfield, M. A., Rousseau, C., Keelan, J., Evans, D. H., Hague, J. P. (2015). Size distribution of air bubbles entering the brain during cardiac surgery. PLoS ONE, 10(4), 1–11. <http://doi.org/10.1371/journal.pone.0122166>

## Conference and Meeting abstracts

**Marshall, D.**, Chung, E., Hague, J., & Horsfield, M. (2013). Computational simulation of the movement of solids and gasses in the cerebral arteries [Abstract]. Cerebrovascular Diseases (Abstracts of the 18th Meeting of the European Society of Neurosonology and Cerebral Hemodynamics (ESNCH) and the 3rd Meeting of the Cerebral Autoregulation Network (CARNet)), 35(Suppl 2), 46. <http://doi.org/10.1159/000351748>

# Table of Contents

Abstract.....	3
Acknowledgements.....	5
Publications arising from this thesis.....	6
Table of Contents.....	7
Section 1. Person specific models of flow in the major cerebral arteries based on MRI data.....	12
1.1 Introduction.....	12
1.2 Anatomy and physiology of the cerebral vasculature .....	13
1.2.1 Cerebral vasculature and the circle of Willis.....	13
1.2.2 The brain's dependence on blood supply.....	15
1.2.3 Autoregulation.....	16
1.2.4 Microvasculature .....	16
1.3 Computational Modelling of Blood Flow .....	18
1.3.1 Blood flow simulation.....	18
1.3.2 Simulating the cerebral vessels.....	19
1.3.3 0D and 1D models .....	20
1.3.4 2D models.....	22
1.3.5 3D models.....	23
1.3.6 Multi-scale models .....	24
1.4 Literature review of computational simulation of the cerebral vessels.	
27	
1.4.1 0D Models.....	27

1.4.2	1D and 2D Models.....	29
1.4.3	3D Models.....	36
1.4.4	Conclusion.....	42
1.5	A generalised 0D fluid model of the cerebral vasculature .....	44
1.5.1	Theory .....	44
1.5.2	Generalisation of the cerebral vasculature.....	45
1.5.3	Model implementation .....	47
1.5.4	Example results.....	47
1.5.5	Conclusion.....	48
1.6	Creating a phantom of the cerebral vasculature for validation of the 0D model.....	49
1.6.1	Introduction.....	49
1.6.2	Acquisition of MRA from a healthy volunteer .....	49
1.6.3	Segmentation and processing of the cerebral MRA .....	50
1.6.4	Phantom fabrication.....	53
1.6.5	Integration of the phantom into a flow rig .....	54
1.6.6	Validation of the 0D model using the flow rig by comparing flow in efferent vessels.....	57
1.6.7	Imaging the flow rig using MRI.....	62
1.6.8	Discussion.....	65
1.7	Patient specific segmentation and analysis of anatomy.....	66
1.7.1	Segmentation and geometric analysis using VMTK .....	66
1.7.2	Acquisition of patient data.....	73
1.7.3	Segmenting MR angiograms for 47 patients .....	74

1.7.4	Categorising anatomical variants.....	75
1.7.5	Discussion.....	80
1.8	0D modelling of patient-specific cerebral blood flow .....	81
1.8.1	Calculating flow in the cerebral vasculature for a single patient ...	81
1.8.2	Modelling the effect of carotid stenosis on patient-specific cerebral blood flow.....	83
1.8.3	Discussion of 0D modelling of a patient. ....	83
1.9	Discussion .....	86
Section 2.	Development of Patient Specific Simulations of Embolisation.....	88
2.1	Introduction.....	88
2.1.1	Models of embolus transport.....	89
2.1.2	Non-invasive detection and sizing of cerebral emboli using ultrasound.....	92
2.2	Literature review of computational modelling of cerebral emboli .....	93
2.2.1	The stenosed carotid .....	93
2.2.2	The carotid bifurcation .....	94
2.2.3	The aortic arch.....	96
2.2.4	The cerebral arteries .....	97
2.2.5	Fractal tree models .....	101
2.2.6	Literature discussion.....	104
2.2.7	Literature conclusion .....	108
2.3	Adaptation of a fractal model and embolus sizing algorithm to patient specific intra-operative TCD monitoring.....	109
2.4	Methods.....	112

2.4.1	Introduction.....	112
2.4.2	Extraction of MCA diameters from patient MRA.....	112
2.4.3	Producing patient-specific embolisation simulations .....	114
2.5	Results.....	115
2.5.1	Measurement of MCA diameters for 10 patients using 3D Slicer	115
2.5.2	Measurement of MCA diameters for 48 patients using VMTK....	115
2.5.3	Example embolus propagation simulation results .....	117
2.6	Discussion .....	121
Section 3.	Application of SPH to modelling of emboli in blood flow.....	124
3.1	Introduction .....	124
3.2	Introduction to SPH.....	128
3.2.1	SPH applied to blood flow .....	129
3.2.2	SPH applied to bodies in pipe flow .....	134
3.2.3	Using SPH to simulate emboli in blood flow .....	135
3.3	SPH methodology .....	137
3.3.1	SPH methodology theory .....	137
3.3.2	Use of DualSPHysics for embolus simulation.....	145
3.3.3	Feasibility study of implementation of SPH code for embolus applications .....	146
3.3.4	Development of an in-house SPH code for embolus simulation..	147
3.4	Results.....	156
3.4.1	Introduction.....	156
3.4.2	Results from DualSPHysics simulation of embolus.....	156
3.4.3	Results from feasibility study of SPH code implementation .....	158

3.4.4	Results from in-house SPH code for embolus simulation.....	160
3.4.5	Discussion of DualSPHysics and preliminary in-house code feasibility studies .....	167
3.4.6	Discussion of in-house SPH code for simulating emboli in the flow 168	
	Conclusion.....	172
	Appendix A. Vasculature processing scripts. ....	174
	convert.py .....	174
	segmentation.py.....	176
	clipmesh.py .....	178
	geometry.py.....	180
	displaygeometry.py .....	182
	branchdiameters.py.....	185
	Appendix B. Batch embolus propagation simulation scripts. ....	188
	Bibliography .....	191

# **Section 1. Person specific models of flow in the major cerebral arteries based on MRI data.**

## **1.1 Introduction**

Constructing 3D meshes from medical imaging data is key to the development of anatomically and physiologically realistic in vitro and in silico phantoms of the cerebral circulation. To do this, a pipeline of tools must be used to segment the imaging data, combine the 2D layers into a 3D mesh, optimise the mesh for flow modelling, and if required, extract geometric information about the anatomy. By extracting geometric properties of vessels from imaging data it becomes possible to determine the length, radius, curvature and tapering of vessels, which can be incorporated into simplified (0D and 1D models). It is also possible to examine characteristics of the vascular tree such as bifurcation components and tortuosity as well as investigating the incidence of anatomical variations in the arteries supplying the brain and capacity for collateral flow. Anatomical assessment of the circle of Willis can then be used to highlight potential risk factors for embolisation and haemodynamic insufficiency in the presence of vascular obstruction.

Conventional techniques for producing 3D meshes involve segmentation of image slices into anatomical structures of interest, then layering these segmented images and joining the edges. The mesh can then be refined using smoothing algorithms to realistically mimic physical structures.

## **1.2 Anatomy and physiology of the cerebral vasculature**

### **1.2.1 Cerebral vasculature and the circle of Willis**

The arteries of the brain are interesting to study due to their complex and variable anatomy. As shown in Figure 1.1, the brain is supplied by the carotid and vertebral arteries, which communicate at the base of the brain via an arterial anastomosis (joining of arteries) known as the circle of Willis, see Figure 1.1. However, anatomical variations in the circle of Willis are extremely common: (Krabbe-Hartkamp et al., 1998) found only 42% of subjects had a complete circle of Willis. Common variants include absent anterior or posterior communicating arteries, supply of both anterior cerebral arteries from a single internal carotid artery, and a posterior cerebral artery extending from the internal carotid artery rather than the basilar artery, see Figure 1.2. These variations are of particular interest for patients with stenosis (narrowing) or occlusion of the carotid or vertebral arteries, as the anatomy of the circle of Willis will partly determine whether adequate perfusion can be maintained through recruitment of collateral pathways and/or vasodilation for maintenance of blood flow.

Section 1

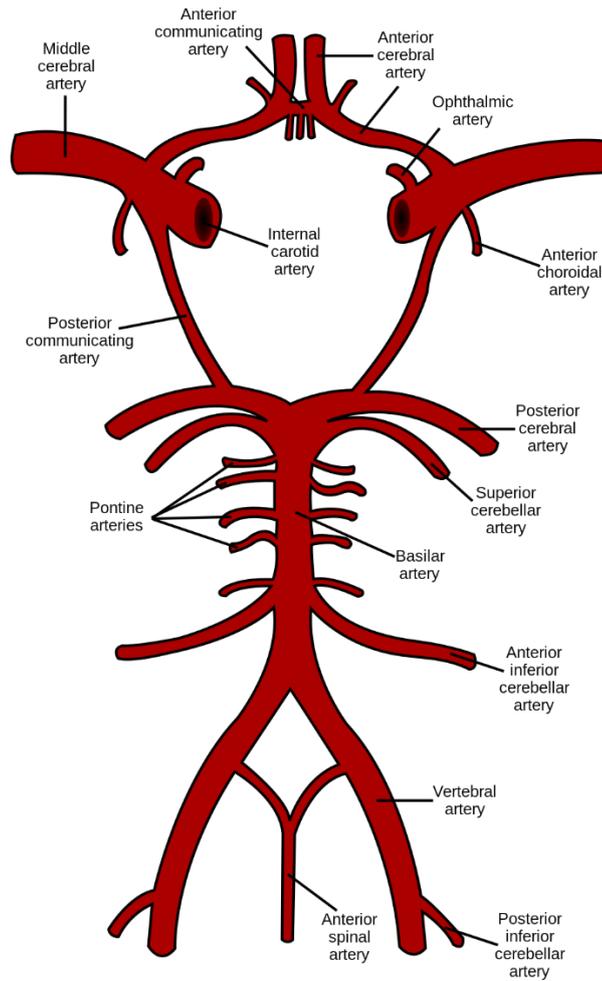


Figure 1.1 Anatomy of the circle of Willis. From source: Wikimedia Commons (Public domain, accessed 2017-09-24).

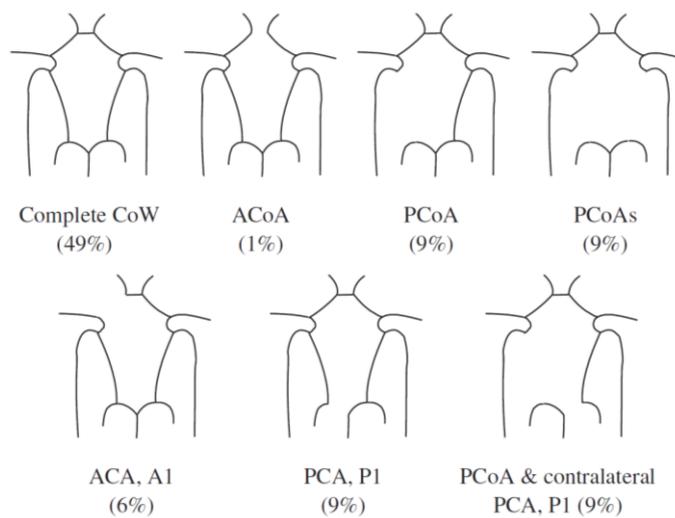


Figure 1.2 Population prevalence of major anatomic variations of the circle of Willis. Image from (Alastruey et al., 2007), prevalence data from (Lippert and Pabst, 1985).

## *Section 1*

### **1.2.2 The brain's dependence on blood supply**

Cells in the body require energy to carry out functions to keep our tissues and organs alive. The energy comes from the metabolism of substrates such as glucose, fatty acids and in some cases (e.g. starvation) protein. Metabolism can be either aerobic; using oxygen, or anaerobic; with no oxygen.

The cells and tissues of the brain are unique in that they are almost entirely dependent on aerobic metabolism to provide energy for cellular functions. The only substrate that can be metabolised aerobically is glucose. Therefore the brain must have a constant supply of oxygen and glucose delivered to its tissues by the blood vessels. (Young et al., 2008 pp276-277)

Depriving the brain of oxygen and glucose by interrupting the blood supply, (e.g. due to a cerebral embolism in stroke patients), will impair aerobic metabolism and the cells will have insufficient energy to survive or carry out cellular processes (such as altering portions of the cell membrane to allow electrical signals to be transmitted).

Depriving the brain of oxygen and glucose for as little as twenty seconds will cause a person to lose consciousness. If deprivation of oxygen and glucose continues for four to five minutes, the cells making up the brain tissue will begin to become damaged, or even die. At the point of cell death in the brain tissue, any damage becomes irreversible as neurological tissue has a limited capacity for regeneration. (Young et al., 2008 p275)

The circle of Willis provides collateral flow in case of occlusion of the carotid, vertebral or basilar arteries. However, due to the high variability in anatomy and cerebral autoregulation between individuals, ability to tolerate occlusion and maintain adequate brain perfusion is also variable (Schomer et al., 1994).

## *Section 1*

### **1.2.3 Autoregulation**

Cerebral Autoregulation aims to maintain adequate blood flow to brain tissue despite changes in blood pressure. It functions by adjusting the diameter of blood vessels to alter vascular resistance. While autoregulation is observed in the blood supply to most organs; the brain, heart and kidneys utilise this process the most, with muscles showing moderate autoregulation and the skin almost none.

In the case of tissue ischemia, autoregulation seeks to compensate flow to that tissue. Autoregulation has a limited capacity for compensation (Moore et al., 2006), if blood pressure at the vessels supplying the tissue increases or decreases to outside of the limits of autoregulation, then tissue perfusion will not be regulated.

### **1.2.4 Microvasculature**

Blood vessels repeatedly bifurcate as we travel downstream, with each new vessel smaller than the last, until the capillary bed is reached, which is a mesh of vessels in which blood exchanges water, carbon dioxide, oxygen, and nutrients with the surrounding tissue. The size of daughter vessels branching from their parent are asymmetric at all depths of the vascular tree, and may increase in size compared with the parent. The deeply bifurcating vascular tree displays fractal properties, with vessels at different levels of the tree manifesting in self-similar branching patterns. (Zamir, 1999) This structure has been extensively studied using X-rays, light microscopy with staining of vessels, the 'Microfil' technique which involves perfusing vessels with a liquid polymer before solidifying, and scanning electron microscopy of resin casts (Kassab et al., 1993). Kassab et al. imaged detailed casts of the coronary arteries of pigs with an inverted light microscope, to provide morphometric data at all vessel 'orders'. They found that, on average, orders of arteries reduce according to a constant diameter ratio in agreement with Horton's law,

## Section 1

$$\log_{10} L_n = a + bn \quad (1.1)$$

Where  $a$  and  $b$  are empirical constants,  $L_n$  is the length of a vessel element at order  $n$ . Structures following Horton's law are known as fractal, which is why we refer to the microvasculature as a fractal system, or tree.

The purpose of a microvascular structure containing increasing numbers of vessels of decreasing size is twofold. Firstly, small vessels collectively have a large surface area, which is important for transporting nutrients between the blood and tissue. Secondly, resistance to flow increases as vessel diameter and length reduce (see section 1.3.3), meaning the blood must travel at a lower flow rate in capillaries compared with arteries (which is also good for material transport). Since net flow must be constant at all levels of the vasculature, this results in a structure with an increasing number of vessels as their scale reduces, achieved through successive bifurcations. The ideal relationship between parent and daughter vessel radii at a bifurcation is given by Murray's law (Murray, 1926; Sherman, 1981),

$$r_0^3 = r_1^3 + r_2^3$$

where  $r_0$  is the radius of the parent vessel and  $r_1$  and  $r_2$  are the radii of the daughter vessels. At the limit of laminar flow, vessels following Murray's law, which minimises the work need to pump blood through the vasculature.

## **1.3 Computational Modelling of Blood Flow**

### **1.3.1 Blood flow simulation**

Computational simulation of the vasculature can be a powerful tool for studying the behaviour of regions of the circulatory system involved in disease. Previous research has focussed largely on the carotid arteries, coronary arteries, the aorta, and the cerebral vasculature (Taylor and Figueroa, 2009). These sites are associated with pathologies such as atherosclerosis, embolism, and aneurysm. Simulations allow these diseases to be studied non-invasively and can reveal information about the vessel and blood mechanics that would not otherwise be observable.

Progress in computational simulations of blood flow began with circuit analogue models that treat each vessel section in a network as a combination of electrical components, such as resistors and capacitors (Westerhof et al., 1969). These "lumped parameter models" are useful for looking at gross flow behaviour, but do not provide information on small scale flow patterns and interactions within vessels.

Models simulating flow within 3D geometries provide more information about flow patterns and wall interactions, but are not presently used to simulate complete systems due to the huge computational cost involved. These models usually involve generation of 3D meshes produced from medical images such as MR or CT angiography. Simulations are patient specific which is useful for surgical planning and studying anatomical risk factors.

To produce physiologically accurate results it is often beneficial to couple 3D simulations to a lumped parameter model. In this way, detail is revealed where it is needed, without requiring unreasonable computing power. The 3D model can be further coupled to more detailed models simulating localised behaviour

## *Section 1*

such as molecular interactions at the lumen. These simulations are referred to as 'multiscale models'.

Recently, increasing research has focussed on modelling micro-scale cellular and molecular interactions, usually at the lumen, using methods such as molecular dynamics and dissipative particle dynamics (Gizzi et al., 2011; Grinberg et al., 2012). Micro-scale simulations are usually performed on a small region, chosen based on the results of, and coupled to, 3D flow simulations. These models can again then be coupled to 1D models describing flow in the remainder of the vasculature.

### **1.3.2 Simulating the cerebral vessels**

Modelling cerebral blood flow can be more difficult than for other areas of the body:

- There are imaging constraints; since the vessels of interest are within the skull they are more difficult to image using ultrasound.
- Flow depends heavily on cerebral autoregulation, the mechanism by which the body ensures adequate perfusion of tissue.
- Anatomical variation amongst the population means that findings may not be generally applicable.

Some models include autoregulation as a modifier of peripheral resistances, but it is often neglected. Models that include autoregulation use a relation that modifies the peripheral resistance according to flow at a rate which depends on a characteristic time constant (Moorhead et al., 2004).

Measurement of in vivo flow velocities is useful for applying boundary conditions and validating model results. The techniques commonly used for cerebral flow measurement are TCD ultrasound and MRA. TCD offers high temporal resolution, but low spatial resolution. It uses the Doppler shift in a pulsed ultrasound signal to determine blood velocity, and can do so with high

## Section 1

accuracy as long as the angle of insonation is known. Patients require a temporal bone window for the ultrasound beam to penetrate, otherwise TCD monitoring is not possible. MRA offers good spatial resolution, but low temporal resolution. Various MRA sequences are available depending on the desired information. Time-of-flight angiography only detects fluid motion perpendicular to the image slice, so unless the flow is in a single direction, slices must be captured in multiple planes and combined. Phase-contrast angiography detects the phase shift in the blood's transverse magnetisation as it moves through a magnetic field gradient, which is then used to estimate blood velocity. MRI contrast agents can also be injected into the blood, but this is avoided if possible, since image quality achieved without contrast is often sufficient.

### 1.3.3 0D and 1D models

0D and 1D reduced models use parallels between the physics of fluid flow in a tube and the physics of electrical circuits. The well-known Hagen-Poiseuille equation which describes the drop in pressure  $\Delta P$  as fluid travels through a pipe of length  $L$  and radius  $r$  is

$$\Delta P = \frac{8\mu L Q}{\pi r^4} \quad (1.2)$$

where  $\mu$  is the fluid viscosity and  $Q$  is the volume flow rate through the tube.

Compare this to Ohm's law for an electrical circuit

$$\Delta V = IR \quad (1.3)$$

where  $\Delta V$  is the potential difference between two points in a circuit,  $I$  is the electrical current, and  $R$  is the resistance between the points. It is immediately apparent in this analogy that potential difference represents a pressure difference and current represents flow. Therefore, the resistance of a tube is

## Section 1

$$R = \frac{8\mu L}{\pi r^4} \quad (1.4)$$

This analogy is an approximation and only valid for a theoretical straight tube with no entrance length effects. It does not consider factors such as tapering and curvature of the tube or stretching of the vessel wall induced by pulsatile flow, which will alter vessel resistance (Shi et al., 2011). Despite this, the analogy is revealing, with the relation  $R \propto 1/r^4$  describing the extreme dependence of vessel resistance on radius. It is due to this that the body is able to regulate blood flow by up to 40% through altering the radii of arteries (Moore et al., 2005).

Due to this electric-hydraulic analogy many of the rules governing the behaviour of electrical circuits can be adopted. Kirchoff's laws state that the sum of currents entering a junction is equal to the sum of currents leaving. This is equivalent to mass conservation. The voltage law states that the sum of the potential differences across each component along a closed path will be equal to the total potential difference across the path, this is the same for pressures. Applying these rules leads to the possibility of adding the resistances of individual vessel sections in series and in parallel, creating a full circuit representation of the vessels of interest.

0D models use these rules to solve for steady state flows in a system of vessels. The application of mass conservation at each bifurcation produces a set of linear equations. Summing the pressures ( $QR$  terms) across each route through the circuit creates another set of equations. We can then solve this set of  $n$  linear equations for the flows through  $n$  vessels of the system. As this method is only applicable to steady state flow, the effects of pulsatile flow and therefore the time dependence of the flow in each vessel is not calculable. In the vascular system, pulse waves reflect from successive bifurcations, which results in an alteration of mean flow in vessels. Despite these weaknesses, the linear nature

## Section 1

of 0D models allows analytical solutions that can give valuable insights into the gross behaviour of the system (Hillen et al., 1988; Cassot et al., 2000).

1D models introduce positional dependence of variables along the length of vessels, such that pulse wave propagation can be modelled. Models can be validated against *in vivo* data, and used to improve blood pressure measurement by assessing the pressure waveform shape in the brachial or radial artery and relating that to central aortic pressure (Chen et al., 1997). 1D models solve the continuity and Navier-Stokes equations, and, if the model includes them, the equations of equilibrium for an elastic, or viscoelastic, vessel wall (Shi et al., 2011).

While 0D models are now less popular due to their limitations, 1D models are still frequently used and are under continuous development (Matthys et al., 2007). Both 0D and 1D models are now more commonly used to supply boundary conditions to 3D models, to allow detailed simulation in a particular area without the computational cost of simulating the whole network in 3D.

### 1.3.4 2D models

A 2D flow model simulates flow in a single plane between line boundaries representing the vessel wall. Simulations are usually of a single bifurcation (Baaijens et al., 1993; Abolfazli et al., 2014), or of a small section of the vasculature, such as a single vessel occluded by atherosclerotic plaque. (Tian et al., 2013). The geometry of the vessels may be extracted from ultrasound images (of, for instance, a stenosed carotid artery), but are commonly generated without imaging data as idealised cases for the problem of interest.

The equations describing flow in 2D differ from those in 3D, and since real vessels are 3D, results from 2D models will not always be the same as in 3D. Despite this, the simplified nature of 2D models allows analysis which is not possible in 3D, and important results can be obtained that are generalisable to

## *Section 1*

real problems. Since 2D simulations require significantly fewer computations than 3D models at the same resolution, they can be run in less time, or in greater detail. For this reason, as well as reduced complexity in building the model, 2D simulation is sometimes used in preliminary studies that later investigate the same problem in 3D. 2D simulations were popular before sufficient computing power for 3D simulations were available.

2D simulations have been used to study atherosclerotic plaque formation and rupture (Tian et al., 2013), embolus transport (Abolfazli et al., 2014), and the effect of flow patterns on the constituents of the blood and interactions at the endothelium. Modelling a stenosis can reveal areas of fast, slow and stagnant flow. Models of cellular and sub-cellular interaction can then be combined with the flow simulation (Gholami et al., 2014).

2D flow modelling methods include the finite element method, finite difference method, finite volume method, lattice-Boltzmann method, and smoothed-particle hydrodynamics. Each of these methods require boundary conditions to be defined at the inlet and outlet of the simulation, which may be constant or time-dependant, and can be taken from physiological measurements, or produced by a 0D or 1D model (which may also have physiological measurements as an input).

### **1.3.5 3D models**

3D models simulate volumetric flow between 3D vessel boundaries which are usually extracted from CT or MR images. 3D models can simulate the same problems as 2D models, but are usually more complicated to create and run. Since 3D simulations are computationally expensive, they are best suited to small sections of the vasculature. These simulations are usually run on multiple CPUs in computing clusters, since typical simulations may take tens to thousands of hours of CPU time.

## *Section 1*

3D modelling is used to investigate physiological and pathological blood flow. Pathologies such as atherosclerosis, aneurysm, and embolism are commonly investigated. Simulation is also used to design medical devices such as stents.

To create a 3D fluid simulation, first an image is segmented. Segmentation is the process of extracting structures from imaging data. Segmentation algorithms look for features delineating structural boundaries in all image planes to extract the 3D shape that those 2D boundaries represent. The segmented vessel mesh is then processed depending on the simulation method used. For Eulerian methods such as finite volume, the vessels are processed into a mesh with a structure optimised for the chosen computation method. For Lagrangian methods such as smoothed-particle hydrodynamics, the vessels may be represented by layers of particles. (Adami et al., 2012)

Boundary conditions are applied at the edges of the computation domain, and are sourced in the same manner as for 2D models, using physiological data and/or 0D and 1D models.

Methods used for 2D simulation are also used for 3D simulation (see section 1.3.4). Several commercial 3D fluid dynamics software packages are available and are in widespread use in the engineering industry. Open-source and in-house software implementations are also commonly used. The 3D flow solver may be coupled to a different model when studying moving boundaries such as flexible vessel walls or emboli.

### **1.3.6 Multi-scale models**

Biological systems operate across a wide range of scales. In the vasculature, we range from arteries with diameters of several centimetres or millimetres, down to capillaries at a scale of several micrometres. Simulating all scales varying by more than a factor of 1000 is not possible with a brute-force 3D simulation since the computational cost would be enormous. We also reach resolution limits

## *Section 1*

when studying interactions with components of the blood, and simulating the movement of all red blood cells in even a small vessel is not currently possible.

Simulating these problems usually requires what are known as multi-scale models. These models couple together different methods, or the same method at different resolutions, to give an insight into the effect of macro-scale flow on micro-scale behaviour. (Grinberg et al., 2009)

3D simulations can be used to study large scale vascular networks, which is useful for investigating problems such as oxygen transport and drug delivery. They can also be used for simulating specific regions of the vasculature to improve our understanding of biomechanics and pre-surgical planning. Most simulations are, however, limited to a small number of arteries.

Problems which cannot be solved by standard 3D simulation are how global flows lead to local effects, such as plaque formation and rupture, clotting, and occlusion. These problems involve interactions on several scales, down to sub-cellular interactions. In large scale systems, the bulk flow is coupled to local time-dependant pulsatile flow, which has a significant impact on small-scale interactions. Surgery or pathology may lead to issues such as high or low wall-shear-stress at a distance from the cause, resulting in new pathology at a distal location. Injured blood vessels can grow plaques, which may rupture, causing platelets to clot and eventually result in occlusion. (Grinberg et al., 2009)

To simulate these problems, a combination of methods are needed along with high computing power. 1D, 3D, and microscopic models, which are both space and time dependent, would be required. These include simulation of portions of the vasculature that can be imaged (with, for instance CT or MRI), as well as those that cannot be imaged due to resolution constraints. See Figure 1.3.

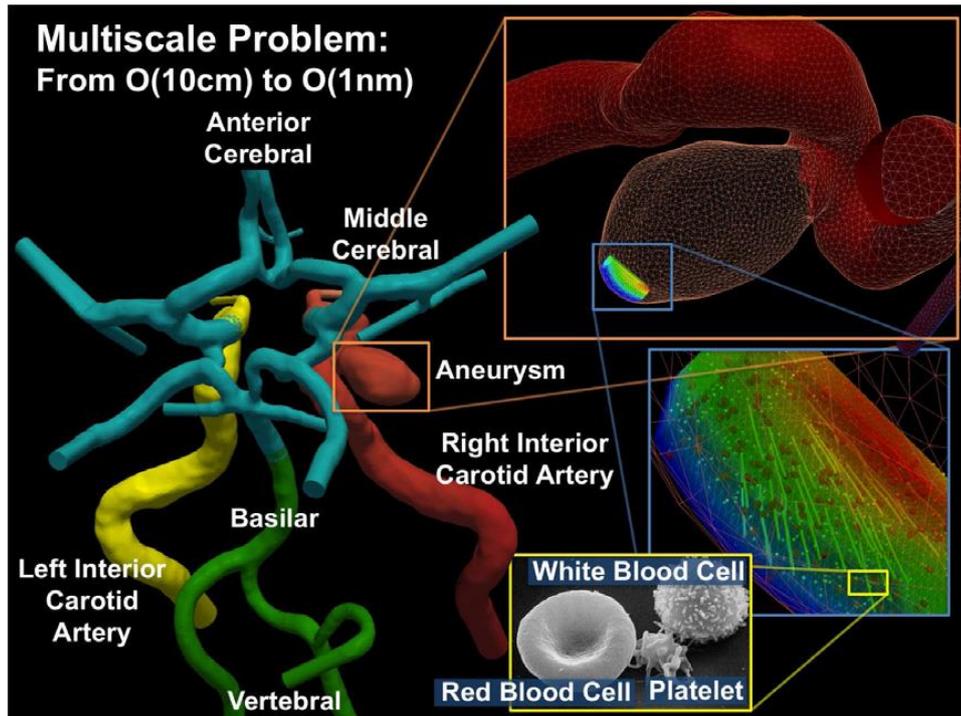


Figure 1.3. Image from Grinberg et al., 2012 showing a multiscale simulation. A region of interest is defined at the aneurysm wall, where a micro-scale simulation is performed to reveal additional detail at a cellular level.

## 1.4 Literature review of computational simulation of the cerebral vessels.

### 1.4.1 0D Models

0D models are simpler to implement than 1D models, so early models of the cerebral blood flow were 0D. These models evolved into 1D models, but were not replaced completely since these simpler models can be easier to analyse and provide insights not possible with 1D models.

An electrical circuit analogue model was first applied to the study of cerebral blood flow by Murray in 1963. Murray (1963) obtained average artery lengths and diameters from autopsies of thirty-five brains fixed in formol-saline and reduced this information to an electrical circuit, as shown in Figure 1.4.

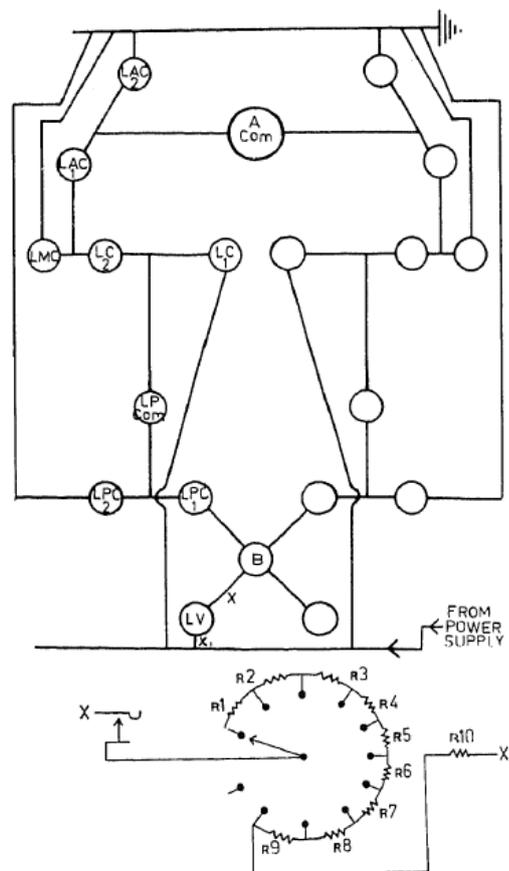


Figure 1.4 An electrical circuit analogue of the major cerebral arteries supplying the brain, from Murray (1963).

## *Section 1*

Results from the circuit were compared with observations from glass and plastic models of the cerebral arteries developed by Rogers (1947) and Avman and Bering (1961). Murray found that flows in the analogue were in agreement with those in the physical models, and experimented with blocking the carotid and basilar arteries. He found that occlusion significantly reduced flow in the territory supplied by the artery, but did not affect other areas; from this he concluded that the circle of Willis is “a very poor anastomotic channel under acute conditions”.

Murray’s model neglected cerebral autoregulation, in which vasospasm in the capillaries of the oxygen and glucose-starved region would act to draw flow through the communicating arteries to replace the missing blood supply. Early experimental work also included a study by Fasano et al. (1966) who attempted to incorporate the effects of vasospasm by developing a laboratory replica incorporating autoregulation accompanied by a physical electrical circuit analogue with variable resistors. They used an adjustable sponge filter at the model outlets to provide variable peripheral resistance in the physical model and found that varying these resistances could lead to similar pressure changes as from arterial occlusion, and therefore flow would be compensated through both the anatomy of the circle of Willis and distal vasodilation. Fasano et al. suggested that it was not just the anatomy of the circle of Willis, but impaired autoregulatory responses that present the most serious threat to regional cerebral blood flow. Given adequate capacity in the communicating arteries, a blockage of a carotid, vertebral, or basilar artery becomes a threat to cerebral perfusion only if there is also a lack of compensation by autoregulation.

At a similar time, Clark et al. (1965) produced a 0D model using both an electric circuit and a computational model, as well as a fluid model with tubing, based on canine anatomy. Results were compared with physiological measurements for occlusion of the afferent and efferent vessels. In the case of occlusion, Clark

## *Section 1*

et al. found differences between measured and simulated values, and suggested some improvements to accommodate this. Effluent resistances were adjusted such that each vessel supplied flow proportional to the weight of brain tissue it supplies.

Hillen et al. (1988) created a 0D model of the circle of Willis. Being 0D, the model did not include pulsatility and vessel elasticity, but the authors found that the simplified approach allowed analytical solutions which would not be possible with a 1D model. They found that flows did not differ significantly from a 1D model of the same network, and that flows were largely dependent on the peripheral resistances used.

Cassot et al. (2000) also used a 0D model to investigate analytical solutions to flow in the circle of Willis, but in the context of occlusion of the carotid arteries. The authors found that flow response to occlusion was highly variable depending on anatomy of the communicating arteries. They also found little difference between results from a 0D and 1D model. The authors propose that models such as this could be useful tools for assessing patient specific cerebral haemodynamics during carotid occlusion.

### **1.4.2 1D and 2D Models**

Developments in 1D cerebral blood flow models have followed a trend toward simulating human anatomy of increasing complexity, nonlinear pulsatile flow and nonlinear elastic walls, with incremental improvements in flow and wall models. New developments tend to begin by simulating flow in areas of the body that are easier to probe clinically than the brain, and are later applied to the cerebral vessels.

The first published 1D model of the cerebral arteries was by Clark et al. (1965), where a computational 1D model with steady flow and rigid vessel walls was used to simulate the cerebral arteries of a dog. This work was continued in

## *Section 1*

Clark et al. (1967), then Himwich and Clark (1971) and Himwich and Clark (1974). The model was validated for physiological flows, and pathological flows in the case of occlusion of one or both carotids, in this case the authors found agreement between their model and flows measured in the dog. A similar model was also applied to the cerebral arteries of a Monkey in Himwich et al. (1968).

A pulsatile and linear flow model using linear elastic walls was applied to dog cerebral arteries by Chao and Hwang (1971), and continued in Chao and Hwang (1972).

Schaaf and Abbrecht (1972) created a nonlinear flow model with linear elastic walls applied to all large human systemic arteries. The authors compared pressure and flow waveforms with clinical data and found good agreement. They also found that their model offered improved accuracy over a linear lumped-parameter model; they attributed this improvement to the use of elastic vessels.

A study by Duros and Nadvornik (1977) applied a pulsatile model with linear elastic walls to human cerebral arteries.

Clark and Kufahl (1978) added nonlinear pulsatile flow and nonlinear elastic walls to the 1D model and applied it to a dog. The model was progressed to 2D in Kufahl and Clark (1985).

Hillen et al. (1982) studied the posterior communicating artery using a small model of a PComA, PCA and ICA. The authors found that the PComA compensated for flow insufficiency to the brain, regardless of the communicating artery's size, and that efferent flows were mainly dependant on peripheral resistances.

Hillen (1986) studied cerebral blood flow in the human using a pulsatile, nonlinear 1D model with linear elastic vessel walls. In this study, the authors

## *Section 1*

assessed the effect on pressure and flow of variations in the anatomy of the circle of Willis. They determined that flow within the afferent arteries and arteries of the circle of Willis was largely dependent on the length and width of the vessel, but that net flow beyond the efferent vessels was only determined by the ratio of peripheral resistances. The effect of vessel size was most prominent in the communicating arteries. Interestingly, when a posterior communicating artery is above a certain diameter, the size of the contralateral posterior communicating artery has no impact on flow in the ipsilateral carotid artery. They also conclude that results from a simpler 0D model are in agreement with the 1D model, and therefore, that results from 0D models are useful and sufficient when time-dependant information is not required.

Around the same time, Zagzoule & Marc-Vergnes (1986) created a 1D nonlinear pulsatile model of the circle of Willis with linear elastic walls. The authors used the model to study autoregulation during arterial hypotension, they found that vasodilation of the cerebral arteries and arterioles was sufficient to maintain cerebral blood flow at normal blood pressure, but would be unable to do so at low blood pressure, in which case the pial and intracerebral arteries must constrict to maintain blood flow.

Cassot et al. (1995) used two computational models applied to the cerebral arteries. They investigated the effect of AComA diameter and ICA stenosis on pressure and flow in the cerebral arteries. Changes in the AComA diameter had a significant effect on pressures and flows. Pressure reserve in the MCAs ACAs fell below the lower limit of autoregulation under two conditions: when an ICA had 70% stenosis and the AComA was small, and when an ICA had 50% stenosis, the other ICA was occluded and the AComA was large. This is demonstrated in Figure 1.5, which clearly shows that unrecoverable MCA pressure loss occurs only with unilateral stenosis for a small AComA; a large AComA allows significant compensation even with complete ICA occlusion.

## Section 1

The authors define pressure reserve as the pressure difference between an artery and the capillary pressure of 35 mmHg. The lower limit of autoregulation was taken as a pressure reserve of 23 mmHg. This study does not assess the effect of variations in size of the PComAs, which could mitigate the effect of a narrow AComA.

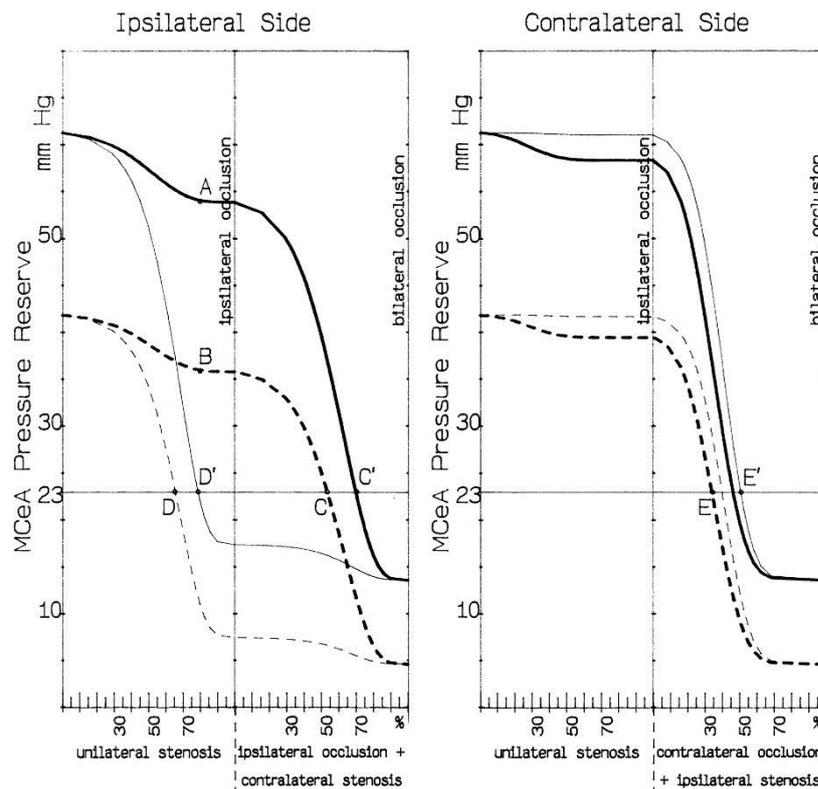


Figure 1.5 Effect of ICA stenosis on pressure reserve in the MCAs (Cassot et al., 1995). The x-axis is divided into two halves, the first half is stenosis of the ICA ipsilateral to the MCA, the second half is with the ipsilateral ICA fully occluded and the contralateral ICA stenosed; on the far right, both ICAs are fully occluded. The thick lines are for vasculature with a large diameter 1.6 mm AComA, the thin lines are for a small diameter 0.4 mm AComA. The solid lines are with a mean arterial blood pressure (MABP) of 100 mmHg, and the dotted lines with 80 mmHg MABP.

Viedma et al. (1997) used a 1D model with inputs from transcranial Doppler ultrasound measurements to explain periorbital reverse flow and arrival pulse time delay which had been observed clinically. They tested occlusion of an ICA and variations in communicating artery sizes. Due to the interest in the periorbital arteries, this model is of note since it includes the ECA leading to

## *Section 1*

the PCA, which anastomoses into the ICA. The periorbital arteries are easily seen on angiograms, but are rarely modelled.

Sherwin, Franke et. al. (2003) describe a different method of 1D modelling using space-time variables rather than space-frequency decomposition. The authors find a good agreement between their model and others, and found that including nonlinearity only adds second order effects, as found in other models. This is applied to model all large arteries in the body. In Sherwin, Formaggia et al. (2003) a similar model is applied to study changes in haemodynamics when a vessel is stented. I highlight these papers to draw attention to the development of 1D modelling techniques and applications that occur in parallel to modelling of the cerebral vasculature.

Moorhead et al. (2004) presented a 1D model of the cerebral vessels under many configurations of absent vessels and use a proportional-integral-derivative (PID) controller to represent cerebral autoregulation. Some results from this study are shown in Figure 1.6, where it is clear that despite afferent occlusion, the PID controller acts to maintain flow to the efferent vessels, even when the circle of Willis is incomplete. The authors also present cases where autoregulation is unable to maintain flow.

Section 1

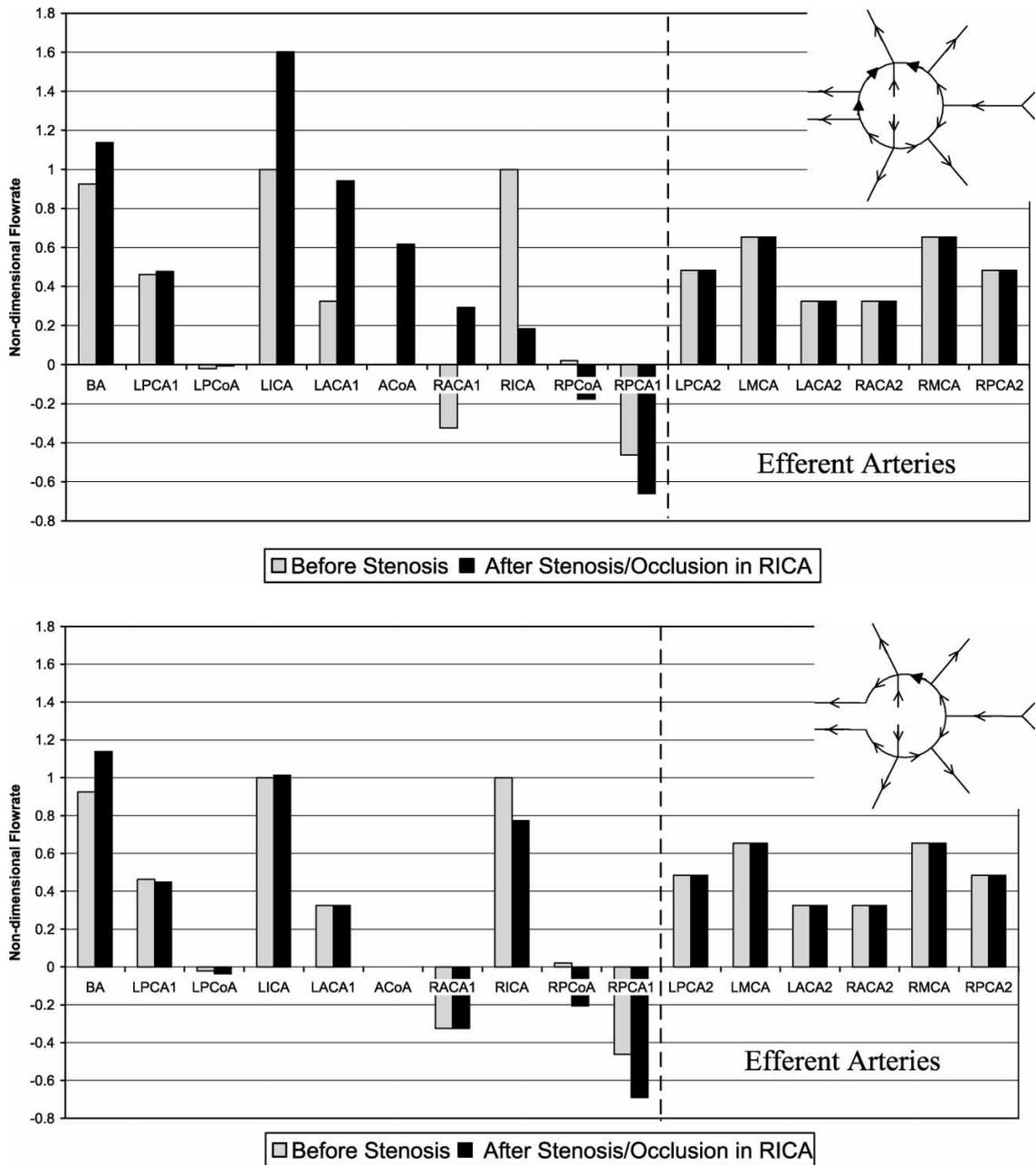


Figure 1.6 Results by Moorhead et al. (2004) showing flow through the circle of Willis before and after occlusion of an ICA, with a present AComA (top) and absent AComA (bottom). Flow to the efferent arteries is maintained in both cases, even though the flows within the circle of Willis vary significantly. The diagrams in the top right of the plot demonstrate flows changing direction.

Alastruey et al. (2007) present an extended model of the cerebral arteries which includes the ascending aorta leaving the heart and extends to the ECA, efferent arteries of the circle of Willis, brachial arteries and thoracic aorta (Figure 1.7).

This extensive model is used to precisely model pulse wave propagation

## Section 1

through the circle of Willis starting from a known waveform at the heart. The authors model many common variations in the circle of Willis under the effects of occlusion to study how well the vasculature is able to maintain perfusion. The model reproduces clinical physiological data well and provides one of the clearest pictures of the function of the circle of Willis to date. This study revealed that the presence of the AComA was more important than the PComAs in case of occlusion of an ICA, and also that occlusion of the ICAs is far more critical than the VAs. Autoregulation is not included in this model, but the model still gives insightful results.

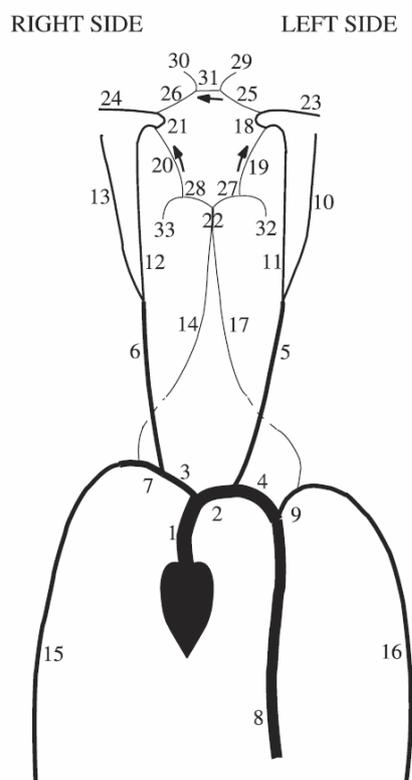


Figure 1.7 1D vascular model by Alastruey et al. (2007). Their model begins at the heart and extends to the thoracic aorta, brachial arteries and head.

Orosz et al. (2009) used a 1D model in combination with data from transcranial colour-coded duplex sonography (TCCD) in moribund patients. After death, the anatomy of the circle of Willis was investigated through autopsy. The authors found that the ability of communicating arteries to provide collateral

## *Section 1*

flow depends on functionality and morphometry, and is therefore better assessed through a functional method such as TCCD than a purely morphological method such as angiography. They suggest that mathematical modelling is a more comprehensive tool for assessing patient risk than imaging alone.

Anor et al. (2010) compare a 1D and 3D model of the circle of Willis and find the results comparable.

Mulder et al. (2011) assess the effect of contrast agent injection on flow in the cerebral arteries using a 1D model. The problem considered is that X-ray videodensitometry is used to measure flow in-vivo, but since it requires a contrast agent, the agent may itself affect the flow. The authors conclude that injection does alter flow, and in the communicating arteries can cause the flow to reverse. This is an example of the kind of problem to which 1D models are now being applied after over 40 years of development.

### **1.4.3 3D Models**

3D flow modelling of the cerebral arteries has only been possible since the turn of the millennium, when advances in computing power began to allow detailed flow simulation in complex geometries.

In an early study on 3D modelling of the circle of Willis, Cebal et al. (2003) use MR angiography to create patient-specific models of the cerebral vasculature. The fluid is modelled using a finite element approach. The authors argue that only approximately 40% of the population have a complete and well balanced circle of Willis, so considering anatomical variation is necessary for obtaining clinically relevant results. As outlet boundary conditions, the authors generate arterial trees which perfuse a given volume within the brain (Figure 1.8); these trees are then described by a 1D flow model connected to the outlets of the 3D

## Section 1

model. Fidelity of the model is hindered by noise and resolution limitations of the MR angiogram, as well as imaging artefacts.

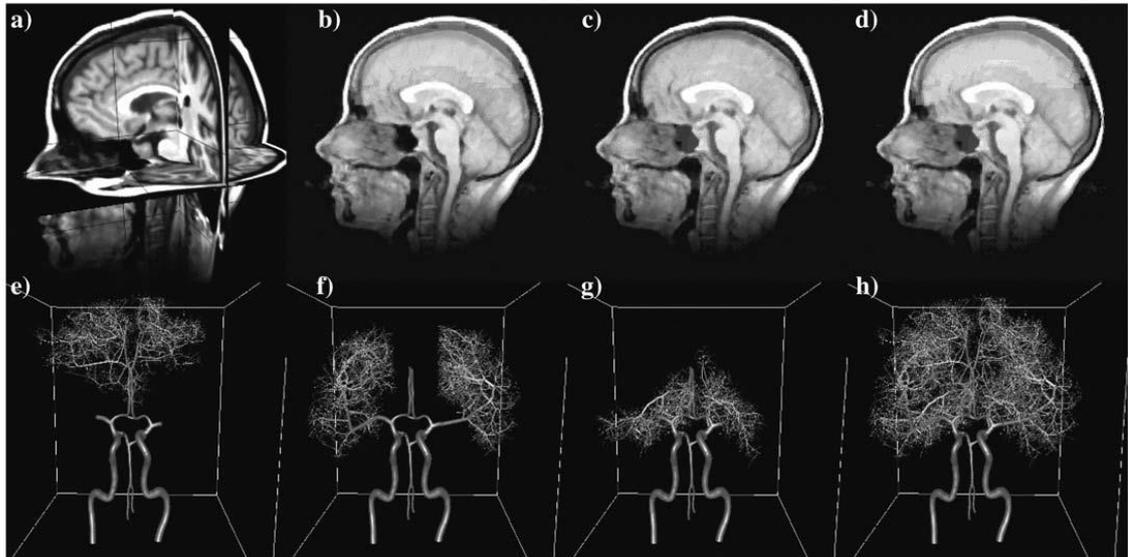


Figure 1.8 Generation of vascular trees to perfuse regions of the brain by Cebral et al. (2003). Perfusion territories are manually selected by labelling voxels in an MRI (top), and the trees are “grown” to perfuse those territories using the methods of Karch et al. (2000).

Oshima (2004) also performed patient-specific 3D modelling of the cerebral arteries, but in this study small sections were chosen to study aneurysms (Figure 1.9).

## Section 1

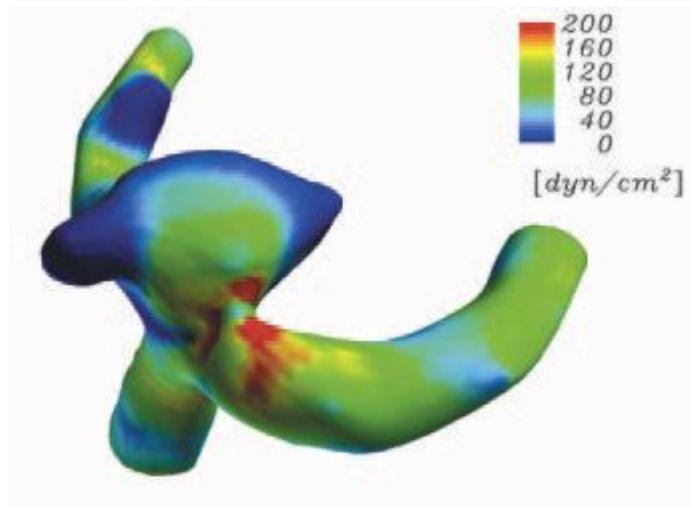


Figure 1.9 Wall shear stress on artery and aneurysm walls at an MCA bifurcation, calculated using a 3D flow model by Oshima (2004).

Moore et al. (2005) compared 1D and 3D flow models of the same cerebral arteries and found the results agreed. A proportional-integral (PI) controller was used to represent autoregulation beyond the efferent arteries. The authors cite the speed of computation of the 1D model as being useful for real-time patient analysis, but note that the 3D model showed increased resistance of the communicating arteries due to extreme geometry which was not modelled by the 1D Poiseuille flow model. This information was then used to modify the 1D model to improve results by increase the resistance of the communicating arteries.

In Moore et al. (2006), the authors discuss creating a 3D flow model of the cerebral arteries using MRI and CAD (computer-aided design) software. The effect of missing vessels on flow patterns was modelled as a proof-of-concept.

## Section 1

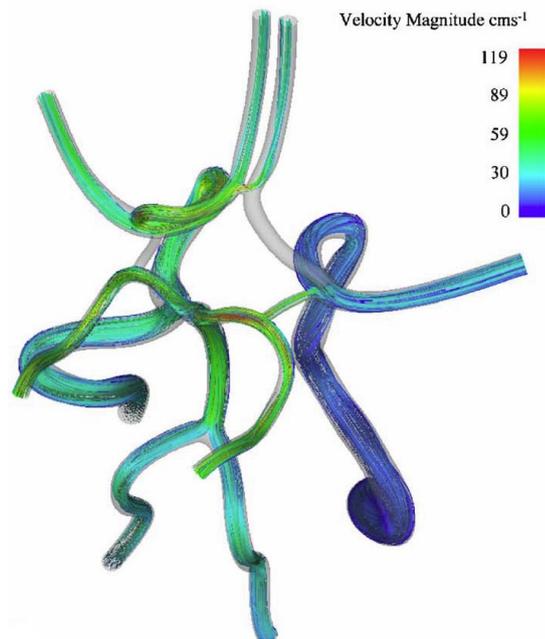


Figure 1.10 3D flow model of the cerebral arteries by Moore et al., 2006. The RACA1 segment is “occluded” to model flow in the absence of the vessel. Flow is indicated by streamlines. The geometry is constructed from MRI and CAD (computer-aided design) software.

Alnaes et al. (2007) investigated wall shear stress (WSS) and pressure in the circle of Willis and found that high WSS occurred where aneurysms are most commonly found.

Chatziprodromou et al. (2007) also studied cerebral aneurysm with a 3D flow model, but in the context of a single vessel distending due to stresses.

Šutalo et al. (2009) created a patient-specific 3D flow model of the circle of Willis (Figure 1.11). The 3D mesh was created from a CT angiogram. The authors coupled the 3D model to a fractal tree model of the peripheral vessels. The results with the CT geometry were compared against a simplified “pipe” geometry, and the mass flows for each model were within 4% of each other.

Section 1

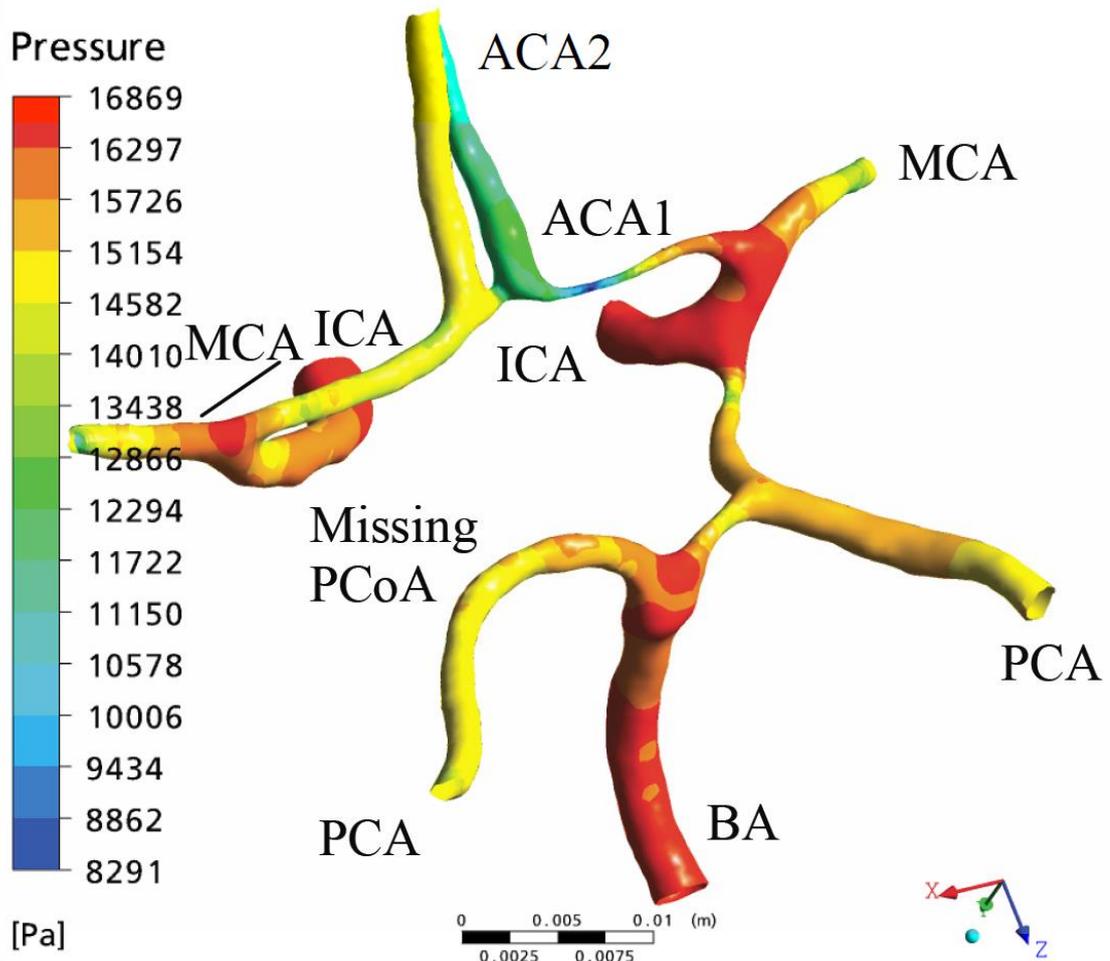


Figure 1.11 3D flow model of a patient-specific circle of Willis by Šutalo et al. (2009).

Jou et al. (2010) also investigated a cerebral aneurysm using a full circle of Willis model.

Anor et al. (2010) provide a review of the state-of-the-art in 3D and 1D modelling of complex arterial networks such as the circle of Willis. The authors conclude that the goal of ongoing research is to combine medical imaging and flow simulation seamlessly, with a view to providing validated patient-specific flow simulations for use as predictive tools in diagnosis, perioperative planning, and for guidance during surgery. Comparing 1D and 3D models, they highlight the accuracy of mass flow predictions by 1D models, and the necessity of 3D models for estimating WSS, to identify areas of stagnating blood, and for any other investigation involving local flow and pressure

## Section 1

patterns. The authors urge that patient-specific boundary conditions are necessary for accurate simulation, but highlight the difficulties in obtaining this information.

Bernabeu et al. (2013) apply pathological thresholds for 'shear-thinning' of blood to a 3D flow model of the MCA to determine the influence of correct blood rheology modelling on WSS. This work again focused on cerebral aneurysms.

Grinberg et al. (2013) created a multiscale model of the cerebral vasculature (Figure 1.12) to study clot formation in a cerebral aneurysm. The 3D flow model within patient-specific vasculature geometry is coupled at the large scale to a 1D model, and at the small scale to a cellular model to predict clot formation at the aneurysm wall. The large scale flow patterns are required to calculate areas of recirculating flow, and the small scale model is needed to determine how cellular interactions lead to thrombogenesis.

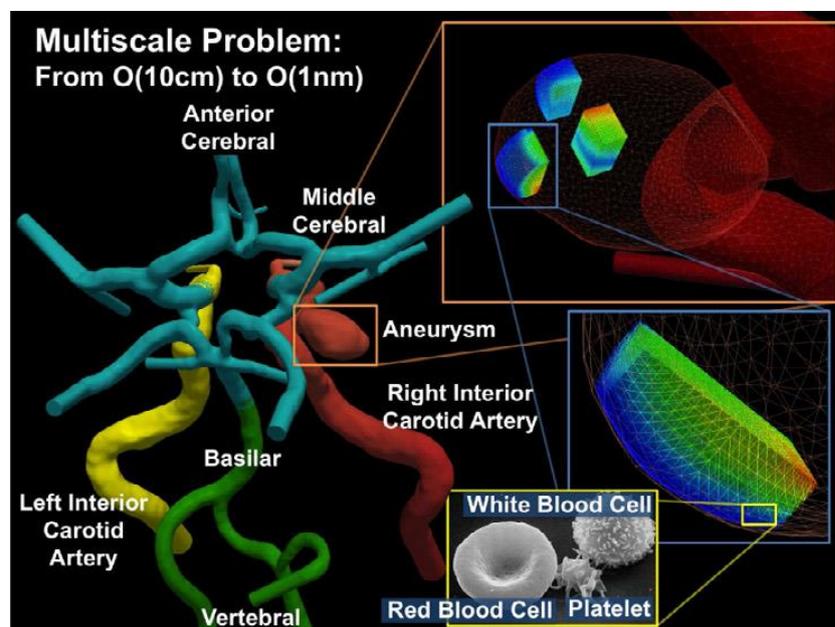


Figure 1.12 Modelling aneurysm in the cerebral arteries as a multiscale problem (Grinberg et al., 2013). To investigate interactions between the blood and aneurysm wall at a cellular level, the domain must be divided into many coupled regions to provide detail where necessary without prohibitively increasing the computational burden.

#### **1.4.4 Conclusion**

Models of flow in the circle of Willis spent many years evolving as 0D, 1D and 2D models, only to move into 3D relatively recently. Advances in computing power are leading to rapidly expanding research in this area, with a shared hope of applying these models to patient-specific problems.

0D models have proven to produce accurate results where bulk flow is of interest. Boundary conditions must be carefully chosen, as for any model, but otherwise the simplicity of the model allows for additional analyses which are not possible with more complex methods.

1D models have used a variety of different techniques, which have all generally produced similar results. One apparent difference between models is whether or not autoregulation is included. It seems that the results are essentially the same in either case, but what is measured to provide those results is varied. When autoregulation is included, the limits of the brain's ability to regulate flow must also be considered; in cases that autoregulation is unable to maintain flow, we see a drop in flow in the model. When autoregulation is absent, flow will drop when there is significant occlusion; in this case we know that autoregulation would attempt to increase that flow, so by investigating the pressure at the efferent arteries it becomes possible to assess whether autoregulation would be capable of maintaining perfusion.

Studies of 3D models of the cerebral vasculature have almost completely focused on cerebral aneurysm. While there are other reasons to be interested in the additional detail 3D simulations provide, aneurysms are a clinically relevant problem which 3D models are ideally suited to investigating. It is therefore unsurprising that aneurysms have been the focus of research. 3D models are now moving into multiscale coupled simulations that may make further use of the additional flow information that 3D models can provide. It will be interesting to see how these new models develop.

### *Section 1*

While a significant body of work exists in simulating the cerebral vessels, there is still a lack of application to large quantities of clinical data. As highlighted in the review by Anor et al., 2010, extensive validation of techniques is necessary if modelling is ever to be accepted for clinical use. Work in later sections of this thesis will investigate patient-specific flow simulation for a large body of patients with differing cerebral vasculature anatomies.

## 1.5 A generalised 0D fluid model of the cerebral vasculature

### 1.5.1 Theory

The accuracy of 0D models can be improved moderately by considering the effect of entrance length flow and curvature on the resistance of each vessel section. Cieslicki and Ciesla (2005) use empirical formulations for a length dependent resistance,  $R_L$ ,

$$R_L = R \left[ 1 + 0.044 \frac{2r}{L} \text{Re} \right] \quad (1.5)$$

and a curvature dependent resistance,  $R_C$ ,

$$R_C = R \left[ 0.526 + \sqrt{0.225 + 0.022 \sqrt{\frac{r}{r_c}} \text{Re}} \right] \quad (1.6)$$

where  $R$  is the resistance defined by Eq. X,  $R_C$  is the radius of curvature of the vessel, and  $\text{Re}$  is the Reynold's number defined for a circular tube by

$$\text{Re} = \frac{2\rho Q}{\pi r \mu} \quad (1.7)$$

where  $\rho$  is the fluid density. These changes not only introduce a dependence of resistance on the density of the fluid, but also a non-linear dependence on the flow rate  $Q$ . This complicates solving for flows in the system, as now a series of  $n$  non-linear equations must be solved. This is easily accomplished using computer algorithms; however, analytical analysis is more difficult than for a linear 0D-model.

The main limitations of 0D models are:

- The accuracy of the flow solution depends on the accuracy of the input vessel dimensions.

## Section 1

- The 0D model described above does not include autoregulation.

Regarding the accuracy of vessel sizes, for studies looking at flow in an “average” circle of Willis, this tends not to be an issue because vessel dimensions can be averaged from imaging data, or measured post-mortem. For patient-specific modelling, however, the dimensions are usually measured from MR or CT angiograms, where it can be difficult to resolve smaller vessels, such as the communicating arteries, and the boundary of the vessel lumen is often blurred due to motion artefacts and the limited resolution of medical imaging techniques. The diameter of vessels, and presence or absence of smaller communicating arteries, can make a significant difference to flows, especially in the presence of a vascular stenosis or occlusion.

Some authors have successfully added autoregulation to 0D models, but it can be difficult to predict autoregulatory behaviour, particularly in pathological cases when autoregulation may be impaired.. Even with the above limitations, 0D models have been shown to produce accurate and revealing results, and their simplicity aids analysis.

### 1.5.2 Generalisation of the cerebral vasculature

A generic 0D electrical circuit model of the cerebral circulation was developed that could be readily adapted to predict flows within patient-specific anatomy based on MR angiography (MRA) data. The theory described in sections 1.3.3 and 1.5.1 was applied using a code written in C++. The reference anatomy circuit diagram is shown in Figure 1.13; vessels can be removed as needed according to the patient anatomy. Non-linear equations were constructed using resistances given by Eq. (1.5) for multiple paths through the system and mass conservation to give 38 equations in total. These were solved using the non-linear solver in the GNU Scientific Library (Galassi et al., 2009). Inputs for the circle of Willis were then taken from geometric information extracted from patient specific 3D meshes to estimate patient-specific flows.



## Section 1

The peripheral resistances are therefore dependent on the simulation anatomy, we adopt the same assumptions here. Total cerebral blood flow is taken as 727 ml/min, which is a typical value for a young adult (Scheel et al., 2000).

### 1.5.3 Model implementation

#### 1.5.4 Example results

Example results for this model are shown in Figure 1.14. These results correspond to modelling of a laboratory flow phantom that has a number of end arteries joined together. Flows in the vertebral arteries are ignored as these are also joined. The superior cerebellar arteries were blocked on the phantom to facilitate introduction of a catheter for measuring pressure. This example shows how it is simple to customise the model based on a particular anatomy, or to introduce occlusion to see the effects on flow. Each vessel section has a characteristic length and diameter, as well as curvature and tapering if included in the model. These properties can then be used to calculate section resistances.

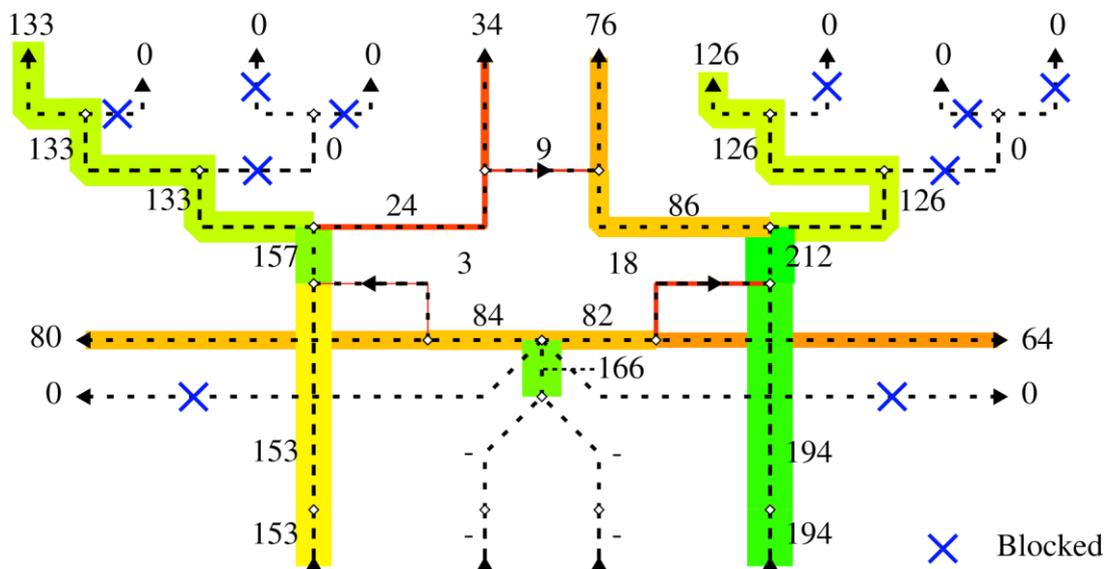


Figure 1.14 Example flow predictions from our OD model. Flows are in ml/min and represented by proportionally scaled and coloured lines.

### **1.5.5 Conclusion**

A generic 0D model of the major cerebral arteries was implemented for fast approximation of flows tailored to an individual patient's angiography data, or phantom anatomy. In future, results of these simulations could be compared with full 3D simulations and used to estimate boundary conditions for multi-scale models (e.g. of emboli in blood flow). Relationships between flows for different anatomical configurations and pathologies (stenosis, aneurysm, or emboli) could then be investigated for the purposes of improving patient risk assessment and treatment planning.

## 1.6 Creating a phantom of the cerebral vasculature for validation of the 0D model

### 1.6.1 Introduction

Phantoms are useful for testing and calibration of instrumentation, testing novel imaging techniques, and provide valuable information for basic science research as an alternative to animal experiments or invasive studies in humans.

### 1.6.2 Acquisition of MRA from a healthy volunteer

The cerebral vasculature of a healthy volunteer was imaged using a 3T MRI scanner (MAGNETOM Skyra, Siemens). Phase contrast and time of flight angiography methods were used to image the vasculature, and T1 and T2-weighted sequences to image brain tissue. Contrast agents were not used. Details of the acquisition are given in Table 1.1. The time of flight angiogram is shown in Figure 1.15.

Table 1.1 Volunteer MRI acquisition parameters.

Property	T1	T2	Time of Flight
Matrix size (pixels)	320x320	320x320	768x696
Slice thickness (mm)	0.8	3	0.5
Pixel size (mm <sup>2</sup> )	0.75x0.75	0.6875x0.6875	0.2604x0.2604
Field of view (mm <sup>2</sup> )	240x240	220x220	199.99x181.24
Repetition time (ms)	1810	6770	21
Echo time (ms)	2.42	108	3.42

## Section 1

Inversion time (ms)	900	2171.3	-
------------------------	-----	--------	---

---

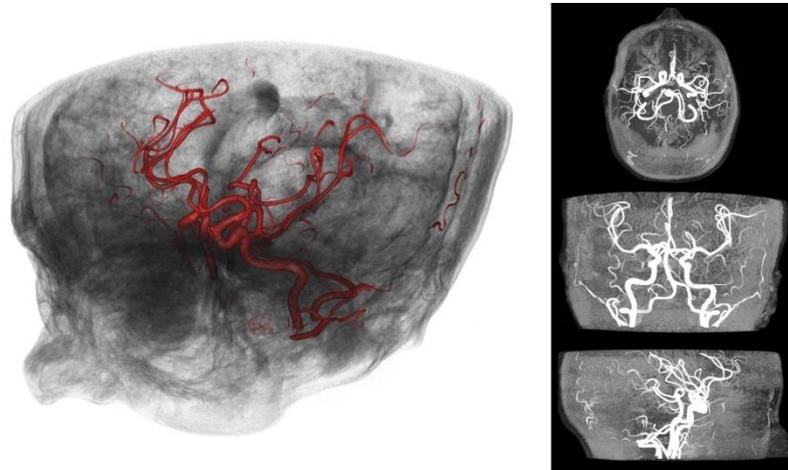


Figure 1.15. Time of flight angiogram captured from a healthy volunteer. On the left is a volume rendering of the angiogram, with voxel values above a threshold displayed in red to highlight the cerebral vasculature. The images to the left are orthogonal maximum intensity projections of the angiogram. These images were created using 3D Slicer. (3D Slicer; Fedorov et al., 2012)

### 1.6.3 Segmentation and processing of the cerebral MRA

The time of flight angiogram was then segmented using Jim (Xinapse Systems). The initial identification and outlining of arteries was performed automatically, then adjusted by hand for each slice to better fit the data and highlight smaller arteries that may be missed.

The posterior communicating arteries were difficult to see in some slices, although their roots at the ICAs and PCAs were often visible. The PCAs were manually segmented based on the position of these roots and by following brighter areas in the slices between the roots.

The outlined slices were then processed into a 3D mesh.

## Section 1

The 3D mesh required further processing for it to be suitable for generating a phantom. Processing was performed using MeshLab (Visual Computing Lab ISTI CNR, 2012), the key steps involved are demonstrated in Figure 1.16.

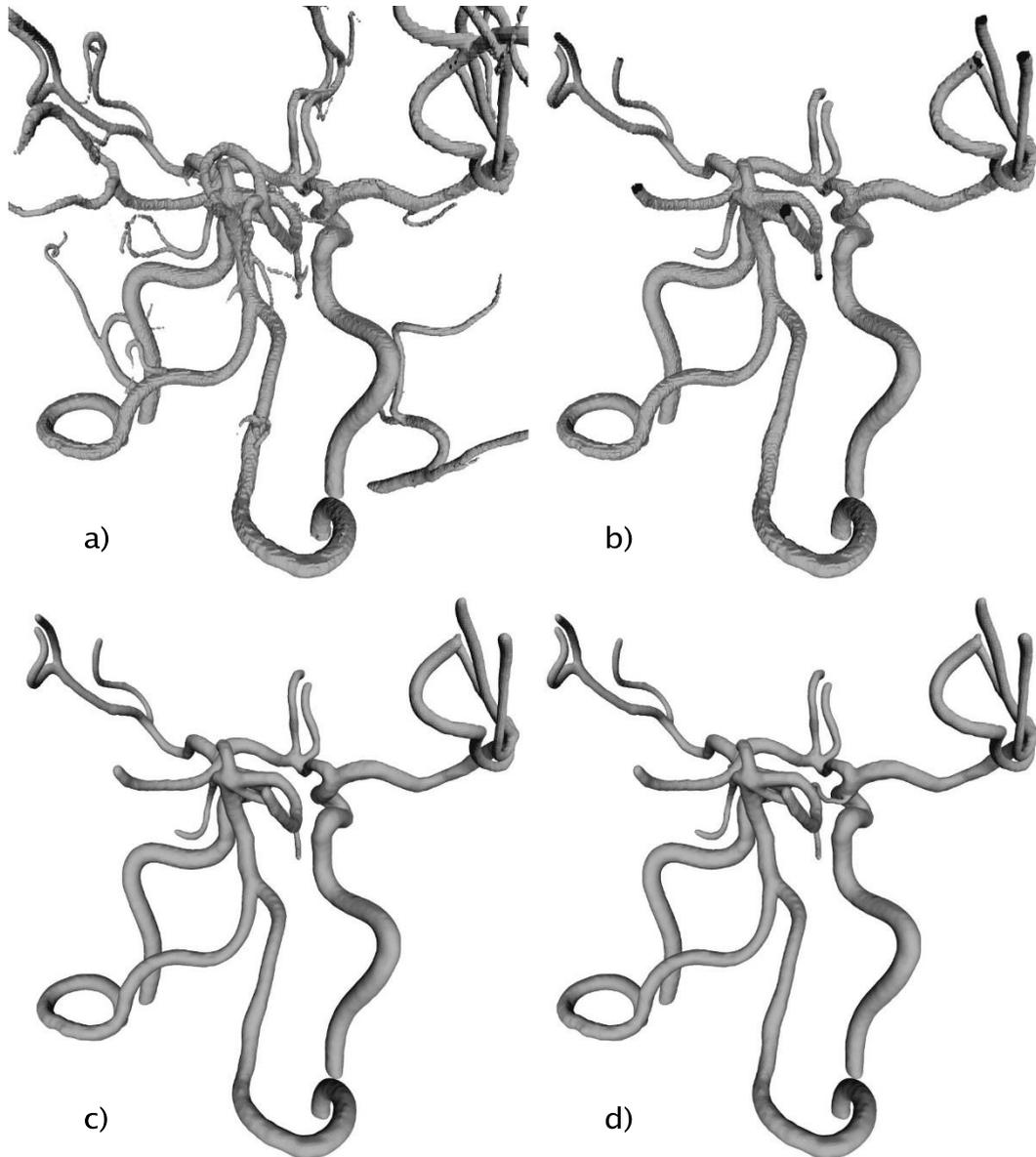


Figure 1.16. Creation of the cerebral vasculature mesh for simulation and phantom fabrication. a) 3D mesh constructed from MRA data. b) Trimming of mesh to select only arteries of interest, posterior communicating arteries removed to avoid distortion of mesh during smoothing stage. c) Mesh smoothing using Poisson filter in MeshLab. (Visual Computing Lab ISTI - CNR, 2012; Kazhdan and Hoppe, 2013) d) Addition of manually constructed posterior communicating arteries which are ignored by the Poisson filter as they are much smaller than the other arteries. The model then only requires cutting at the artery ends depending on the desired application (phantom, CFD, etc.).

## Section 1

Minor vessels stemming from the main arteries were trimmed to simplify the model and avoid including vessels that had not been resolved correctly by MRA due to their small size or flow volume. This involved removing the poorly resolved posterior communicating arteries so that they could be recreated later.

The surface contained ridges caused by the discrete slices of the MRI, so it was smoothed using MeshLab's Poisson surface reconstruction algorithm. The algorithm produces a mesh which approximates the true surface from a set of data points. (Kazhdan and Hoppe, 2013)

Once the mesh was smoothed, the posterior communicating arteries could be added back to the mesh. The arteries were first constructed as a 3D model using 3ds Max. (Autodesk, 2011) The vessels were modelled as narrow tubes with diameters of 1 mm. A 3D line was traced along the original segmented vessels, and the tube was generated around that line.

The resulting mesh with arteries labelled is shown in Figure 1.17.

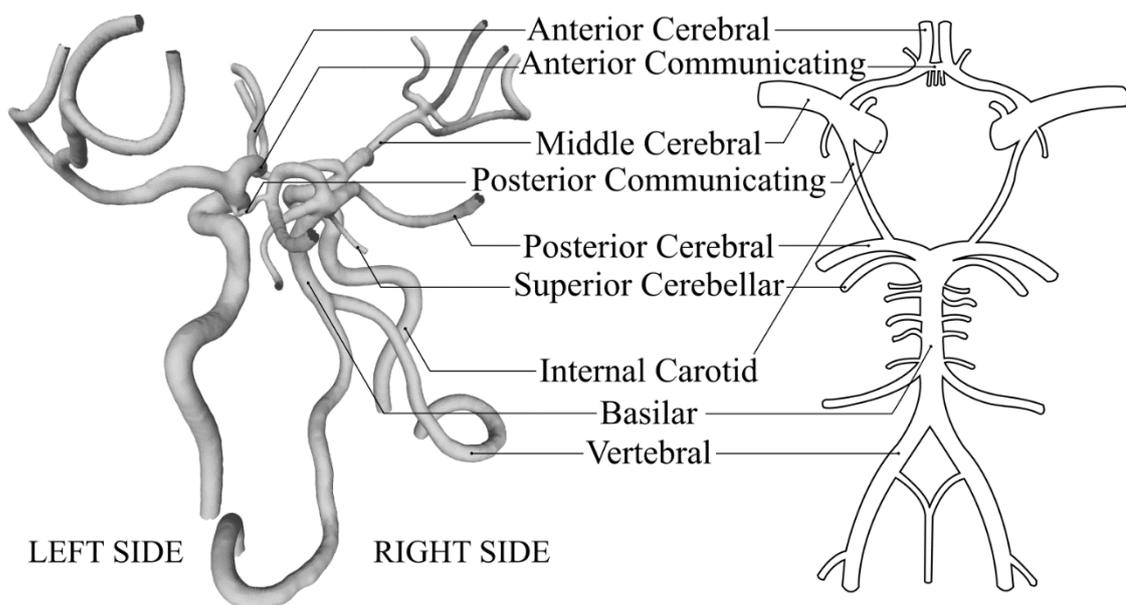


Figure 1.17. Fully segmented mesh of the healthy volunteer's cerebral arteries. The mesh is compared against an anatomical diagram to demonstrate the position of each

## Section 1

vessel. Adapted from source: Wikimedia Commons (Public domain, accessed 2017-09-24).

### 1.6.4 Phantom fabrication

The flow phantom was produced from the 3D reference mesh by Elastrat (Geneva, Switzerland). This process involves casting a wax model of the vasculature, coating the model in silicone, then melting the wax to leave the hollow vessels (Figure 1.18). See Wetzel et al. (2005) for a full description of the fabrication method. The resulting phantom is shown in Figure 1.19.

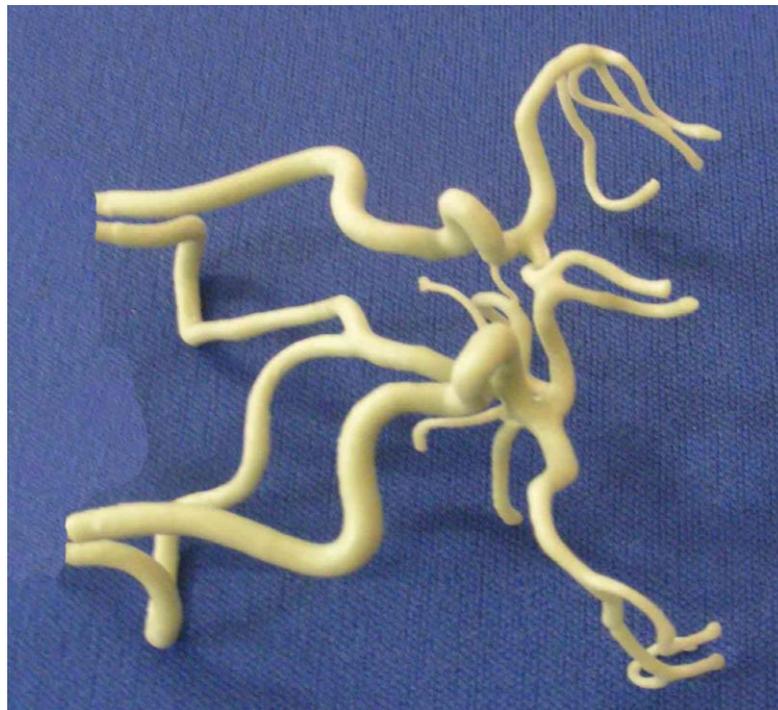


Figure 1.18 Wax model of the volunteer vasculature by Elastrat (Geneva, Switzerland).



Figure 1.19. Phantom of the cerebral vasculature, created from patient specific MRA imaging.

### 1.6.5 Integration of the phantom into a flow rig

The phantom was connected at the afferent vessels to a pulsatile flow pump using flexible silicone C-Flex tubing (Cole-Parmer, UK). Further tubing was connected to efferent vessels to produce peripheral flow resistances. Water was recirculated back to the pump from the outflow tubes via a reservoir as shown in Figure 1.20. Outflow tubing length and diameters were chosen to produce desired resistances, and these resistances determined the distribution of flow leaving the phantom for the physiological case where vessels were not blocked. Inflow tubing and vessels in the phantom have negligible influence on the flow when all vessels are patent because their resistances to flow are orders of magnitude smaller than those of the peripheral resistance outflow tubes. This mirrors the real vasculature, where resistance to flow passing through the

*Section 1*

capillary bed determine the flows in the vessels leading there, not the resistance (or size) of those vessels.

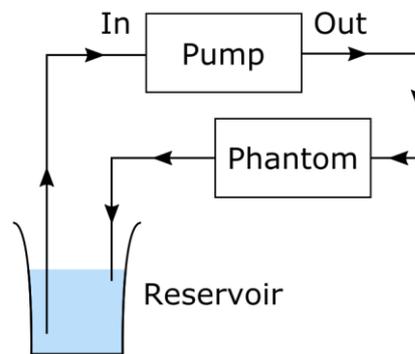


Figure 1.20 Cerebral vasculature phantom flow rig setup.

The outflow tubes were joined in groups (“lumped”) to make the experiment more practical; the setup is shown in Figure 1.21. The 0D model was also lumped for validation purposes, and then modified to use all outlets and produce better, more detailed, results.

Section 1

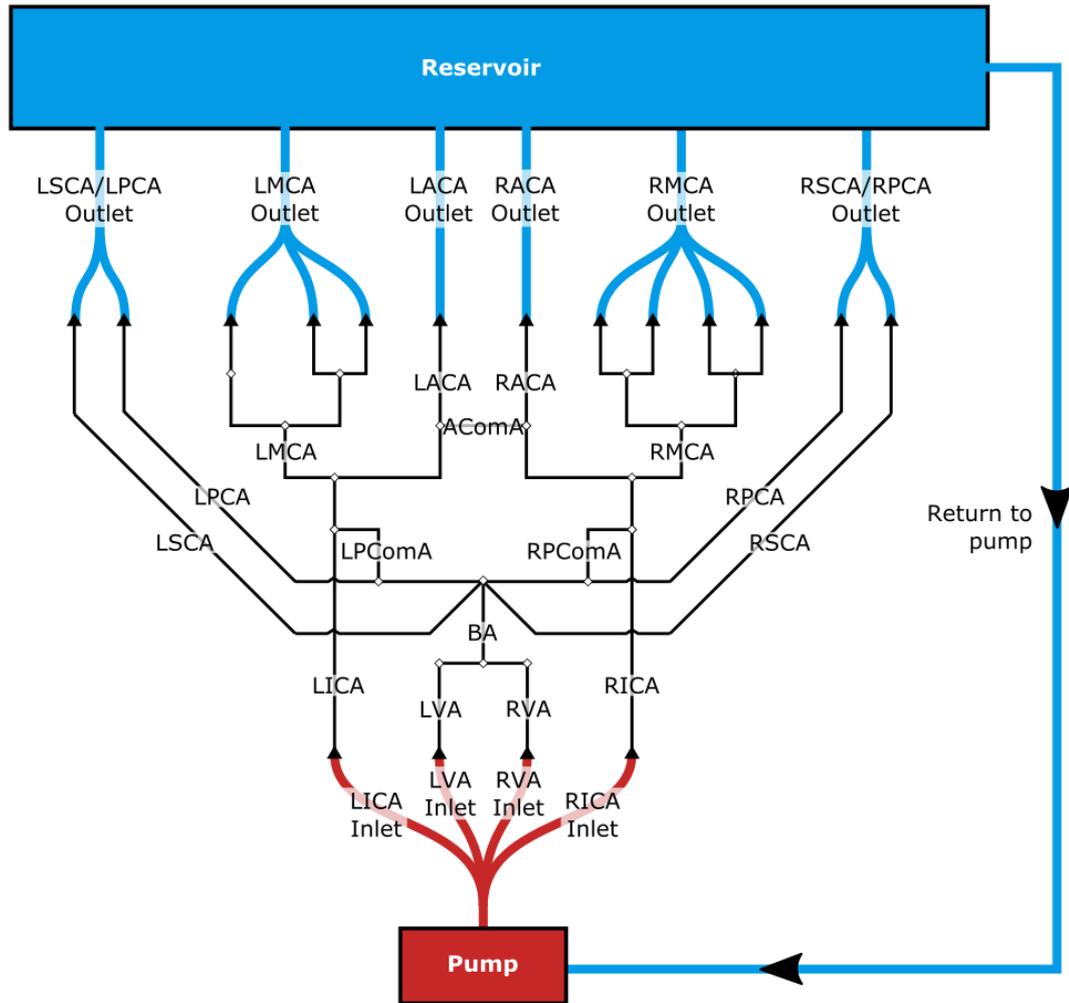


Figure 1.21 Cerebral vasculature phantom inlet and outlet tubing in the flow rig. Inlets are supplied from a common source. Outlets were grouped to facilitate flow measurement; loss of information is not an issue since measuring flow from each efferent vessel is not necessary for validating the 0D model.

The peripheral resistances used were taken from Alastruey et al. (2007) for the ACA, MCA and PCA arteries. Resistance tubing comprised Cole-Parmer™ Silicone C-Flex tubing. The outlet resistances and tube dimensions are given in Table 1.2.

## Section 1

Table 1.2 Outlet resistance tube properties. Resistances and estimated flows are taken from (Alastruey et al., 2007). Tube lengths were chosen based on the required resistance and tube diameters available.

External tube	Resistance ( $10^9$ Pa.s)	Cole-Parmer™ tube ref.	Diameter (mm)	Length (cm)	Estimated flow (ml/min)
LMCA / RMCA	5.97	06424-60	0.76	194.0	104
LACA / RACA	8.48	06424-62	1.60	54.6	70
LPCA / RPCA	11.08	06424-62	1.60	71.3	54

The pump produced continuous flow; since the 0D model and method of measuring flows are not time-dependent pulsatile flow was unnecessary. Flows were measured by timed collection into a measuring cylinder.

The experimental work in this section was carried out by Lucy Collins, Marco Muccio, and Devon Rockafellor, all third year undergraduate physics students of the University of Leicester.

### **1.6.6 Validation of the 0D model using the flow rig by comparing flow in efferent vessels**

Measurements of the flow are given in Table 1.3. Main sources of error were limitations in the accuracy of the timed collection volume flow measurements due to differences in priming of the circuit prior to measurement, and slight differences in the height of the outflow tubes, which have potential to affect resistance to flow.

## Section 1

Table 1.3 Outflows from the phantom in the flow rig compared with expected flow values from Alastruey et al. (2007). Variations are expected since Alastruey et al.'s vasculature dimensions were different, but the total flow should be comparable.

Vessel	Measured flow (ml/min, $\pm 4\%$ )	Expected flow (ml/min)	Difference
RMCA	101	103	-2%
LMCA	101	104	-3%
RACA	74	70	+6%
LACA	75	70	+8%
RPCA	65	53	+17%
LPCA	61	54	+11%
Total	477	454	+5%

The vessel dimensions used as input to the 0D model were measured from the segmented phantom mesh using VMTK. This process is described in detail in section 1.7. The input dimensions are given in Table 1.4; the LMC, RMC, LPC/LSC, RPC/RSC have been lumped to replicate the flow phantom setup. Peripheral resistances were taken from Alastruey et al. (2007).

Section 1

Table 1.4 Vessel dimensions of the phantom flow-rig input into the 0D model. Measurements are missing for vessels which are either missing from the flow-phantom or have been clumped at the inlets or outlets.

Artery	Length (mm)	Diameter (mm)
LIC1	158.1	4.49
LIC2	117.7	3.87
LMC	56.1	2.67
LMCa	-	-
LMCb	34.4	2.09
LAC1	57.4	2.70
RIC1	131.2	4.09
RIC2	102.0	3.60
RMC	55.2	2.65
RMCa	-	-
RMCb	49.6	2.51
RAC1	34.4	2.09
LVA	77.3	3.14
RVA	83.1	3.25
BA	98.5	3.54
LPC1	45.1	2.40
RPC1	52.0	2.57
ACo	-	-
LPCo	8.2	1.02
RPCo	8.2	1.02
LMCaa	-	-
LMCab	-	-
LMCba	-	-
LMCbb	-	-
LAC2	25.8	1.81
RMCaa	-	-
RMCab	-	-
RMCba	-	-
RMCbb	-	-
RAC2	26.2	1.83
LPC2	-	-
LSC	-	-
RPC2	-	-
RSC	-	-

For the 0D model, the lumped vessels were replicated by obstructing clumped sections of the MCA, ACA, and SCA arteries. The remaining MCA, ACA and

## *Section 1*

PCA (which includes the SCA) were given zero length and attached to resistances equal to those of the outflow tubes. The pressure differential used in the model was calculated by combining outlet resistances in parallel, using Kirchoff's laws to estimate total resistance, then using  $\Delta P = QR$ , where  $\Delta P$  is the pressure differential,  $Q$  is total flow (478 ml/min), and  $R$  is the total resistance, calculated as  $1.33 \times 10^9$  Pa.s, giving a pressure differential of 79.5 mmHg.

The results of the 0D model are shown in Table 1.5. We compare the measured data against different models. The non-linear 0D model uses the non-linear flow relation described in section 1.5.1. The simplified model reduces the vessels to only the pressure differential and the resistance tubes; since these resistances are large compared with resistances in other vessels this model should give a reasonable approximation of total flows. The simplified model is useful for comparison in order to assess the effect of considering vessels within the circle of Willis. Finally, we compare against Alastruey et al. (2007). We do not expect the flows to be identical to Alastruey et al., but this is useful for comparison, to demonstrate that vessels in the circle of Willis have an impact on net flows.

## Section 1

Table 1.5 Results of applying the 0D model to the flow phantom compared against measured flows, data from Alastruey et al. (2007), and a simplified model. The non-linear 0D model implements corrections for resistance at bifurcations, whereas the linear 0D model does not. The simplified model approximates the vessels as a pressure applied across the resistance vessels in parallel. Total sum of squares are given to compare the differences of each dataset to the measured flows.

Vessel	Flow rates (ml/min)				
	Measured	Linear 0D model	Non-linear 0D model	Alastruey et al.	Simplified
RMCA	101±4	103	100	104	107
LMCA	101±4	104	101	103	107
RACA	74±3	73	71	70	75
LACA	75±3	73	71	70	75
RPCA	65±3	57	53	54	57
LPCA	61±3	56	58	53	57
Total	477±19	466	455	454	478
Total sum of squares	-	111	178	240	153

The results show that the linear 0D model provides the closest approximation (lowest total sum of squares) to the measured data. Surprisingly, the non-linear 0D model performs worse than the linear model, and marginally worse than even the simplified model, which we do not expect to be accurate. The results from Alastruey et al. (2007) differ the most from the measured data, which is expected and demonstrates that the geometry of the circle of Willis has a significant impact on net flow.

Both 0D models have lower flows in the PCAs than the measured flows, despite replicating other flows well. In the non-linear model, however, only the RPCA flow is outside of the estimated measurement error. I do not have an explanation for the deviation, but it may be related to the way the SCAs and PCAs are lumped.

## *Section 1*

In conclusion, the non-linear 0D model reproduced measured flows in the phantom for all but one outflow vessel (23% difference). These results are promising and deviations from the model assumptions are likely to have been introduced in the experimental setup (e.g. due to additional resistance associated with connectors). Although the non-linear model performed worse than the linear model, experience has shown that non-linear terms can make a significant difference to flows when occlusion causes increased use of the collateral pathways.

### **1.6.7 Imaging the flow rig using MRI**

To determine how well the phantom anatomy approximated segmented images from the original healthy volunteer MRA scan, physiological flow was implemented and the phantom was imaged using MRI.

To generate a signal from the surrounding tissue, the phantom was surrounded by a tissue mimicking material (TMM) with high water content. The TMM used was a polyvinyl alcohol (PVA) based hydrogel containing the following ingredients:

- Water (79.4%)
- Glycerol (9.6%)
- Polyvinyl Alcohol (or 'PVA') (9.6%)
- 3  $\mu$  Calcinated Aluminium Oxide (0.46%)
- 0.3  $\mu$  Ultra fine Aluminium Oxide (0.42%)
- Silicon Carbide (0.26%)
- Benzalkonium Chloride (0.22%)

The ingredients were mixed and placed in a water bath heated to 97 °C, for 2 hours and then cooled to room temperature, the mixture was stirred throughout. The tissue was then frozen and thawed three times, with each freeze and thaw lasting for 12 hours. This process solidifies the mixture

## *Section 1*

(Artmann & Chien, 2008, p115). This work was performed by a fourth year undergraduate physics student from the University of Leicester, James Campbell. By surrounding the vessels with a tissue mimic it was possible to avoid issues that arise when performing MR imaging of small subjects surrounded by air. The TMM is also ultrasound compatible making the flow phantom suitable for hybrid imaging studies.

The flow phantom, connected by long tubes to a pump and reservoir, was placed within a 1.5 T MRI scanner. A blood mimicking fluid (BMF) with similar density and viscosity as blood was pumped through the flow phantom under steady and pulsatile flow. The BMF comprised the following ingredients (mixed for 2 hours and sieved):

- Water (86.37%)
- Glycerol (10.83%)
- Orgasol (1.87%)
- Surfactant (0.92%)

T1 and T2-weighted images were captured for the purpose of imaging vessel anatomy. A time of flight angiogram was also captured, which is why a full flow-rig with pump to generate flow was required. Unfortunately, the time of flight angiography signal was weak, however, the T1 and T2 images were sufficiently high resolution to verify phantom dimensions. These scans are shown in Figure 1.22.

The images in Figure 1.22 appear qualitatively similar to Figure 1.15, suggesting that the phantom provides a reasonable approximation of the original data.

Section 1



Figure 1.22 MR images of the flow phantom, which itself was produced from MR angiography data from a healthy volunteer. A. Maximum intensity projection of the T2 image. B. A slice in the T2 image. C. A slice in the T1 image. In B and C, the vessel wall is clearly visible at the boundary of the BMF within the vessels and TMM outside the vessels

### **1.6.8 Discussion**

We have used MRA images to construct 3D meshes of the cerebral vasculature, extract geometric properties of vessels from imaging data, and to fabricate a flow phantom for in vitro validation. Although it can be difficult to recreate physiological flow within complex phantoms, we show that the choice of appropriate peripheral resistances can be aided by implementing a straightforward 0D electrical analogue model in parallel with the phantom set-up. Reasonable agreement between the 0D model and laboratory measurements was achieved.

Although there was flow through the phantom, the posterior communicating arteries were not visible, despite being patent – this is probably a consequence of low flow and the PComAs being positioned within the slice plane. This is interesting as the PComAs were also not easily seen in the original volunteer MR angiogram. It seems that time-of-flight angiography does not image the PComAs well, which is notable when applying this technique to imaging patient vasculatures, which is studied in the next section.

MR angiography is limited to a resolution on the order of a millimetre, which makes it unable to image very small vessels such as the communicating arteries discussed above, and also arterioles and capillaries. To obtain information about these small vessels we must use techniques such as polymer corrosion casts and arterial tree growth algorithms. As the vessels become smaller, structure is more generalizable between different people. Angiography can be used to image the microvasculature, and methods such as tree growth algorithms can be used to fill in the missing information based on metabolic demand of brain tissue (Keelan et al., 2016).

## **1.7 Patient specific segmentation and analysis of anatomy**

### **1.7.1 Segmentation and geometric analysis using VMTK**

The purpose of blood vessel segmentation is to extract surfaces from a 3D angiogram, which could be obtained from CT or MR data (but here we will be using MRI), and process these data to obtain a 3D mesh representing the vessel lumen (inner surface of the vessels).

#### **Introduction to VMTK**

Traditionally, image segmentation is performed by outlining structures of interest in slices in a 3D image, then connecting those surfaces to produce a 3D mesh. Modern techniques aim to improve upon this by considering known properties of the anatomy and using algorithms to better extrapolate missing data from known data (Antiga and Steinman, 2004). For vascular systems, the open source, community driven software, vascular modelling toolkit (VMTK), has been shown to produce accurate models and measurements from imaging data (Antiga et al., 2008). VMTK uses the “level sets” method to grow a deformable mesh to fit the imaging data; the process is fully automated to avoid subjectivity. When comparing results between patients, objectivity is important and automated techniques improve the reliability of results.

VMTK, in addition to its mesh construction algorithms, can extract geometric information from vascular meshes. Centrelines are mathematically estimated paths passing through the centres of vessels, estimated by maximising the minimal distance to the surrounding surface, i.e. the centrelines are as far away from the wall in all directions as possible (Antiga et al., 2008). Once calculated, centrelines are used to evaluate vessel cross sectional areas, length, and curvature. This method of quantification is robust; small changes in the initial

## Section 1

mesh, which may result from imaging inaccuracies or resolution limitations, will not alter the results significantly.

### Using VMTK to segment the volunteer angiogram

VMTK was used to extract quantitative geometric information from the volunteer reference mesh. VMTK produces a variety of vessel statistics, some of which must be processed to get useful information. Here I used a tool to extract vessel section lengths and average diameters. Lengths are output directly, but diameters required further calculation based on cross sectional areas calculated by VMTK along the length of each vessel. In-house scripts were created in Python to extract the average diameter of each vessel section from these cross sectional area values. Scripts were also developed to automate running of the required VMTK instructions, so that patient data can be processed with little user input. VMTK also provides statistics such as tortuosity and curvature, and the cross sectional areas can be processed to assess tapering and stenosis, which may be useful for future work.

The VMTK commands used for processing the volunteer mesh are described below. In this case VMTK was only used for analysing the mesh, not for creating it.

1. Create centerlines from a .stl file, split the centerlines into branches, calculate branch lengths and view the resulting centerlines. The GroupIds for each branch must be recorded to link each GroupId to a vessel name.

```
vmtkcenterlines \  
    -seedselector openprofiles \  
    -ifile in-1-src.stl \  
    -ofile in-2-cl.vtp \  
--pipe vmtkbranchextractor \  
    -ofile in-3-br.vtp \  
--pipe vmtkbranchgeometry \  
    -ofile in-4-brgeom.dat  
--pipe vmtkcenterlineviewer \  
    -cellarray GroupIds
```

## Section 1

2. Calculate cross-sections along each branch, these will be used to calculate the diameter of the branch.

```
vmtkbranchsections \  
-ifile in-5-branch-clipped.vtp \  
-centerlinesfile in-5-branch-clipped-cl.vtp \  
-distancespheres 1 \  
-radiusarray MaximumInscribedSphereRadius \  
-groupidsarray GroupIds \  
-blinkingarray Blanking \  
-centerlineidsarray CenterlineIds \  
-tractidsarray TractIds \  
-ofile in-6-brsec.vtp
```

3. Convert the branch section output to a data file we can use.

```
vmtksurfacereader \  
-ifile in-6-brsec.vtp \  
--pipe vmktsurfacewriter \  
-celldata 1 \  
-ofile in-6-brsec.dat
```

The `in-4-brgeom.dat` file contains a table of GroupIds, length, curvature, torsion and tortuosity. The `in-6-brsec.dat` file contains a table of cross-section coordinates, the GroupId they belong to, cross-sectional area, minimum and maximum diameters, elongation shape factor, and index of the cross-section along its branch.

Vessel lengths were taken directly from `in-4-brgeom.dat` by correlating each GroupId with vessel name using notes made from visual inspection.

Vessel diameters were calculated using Microsoft Excel by creating a pivot table which lists the average area for all cross-sections in each GroupId. The average area was then used to calculate diameters, and the GroupIds were again correlated with vessel names.

The resulting cross-sections from `vmtkbranchsections` and centrelines in the phantom mesh are shown in Figure 1.25. Centrelines are traced through the centre of each artery, with cross-sections at intervals along the line. Cross

## Section 1

sections are coloured depending on the vessel section to which they belong, with some colours duplicated.

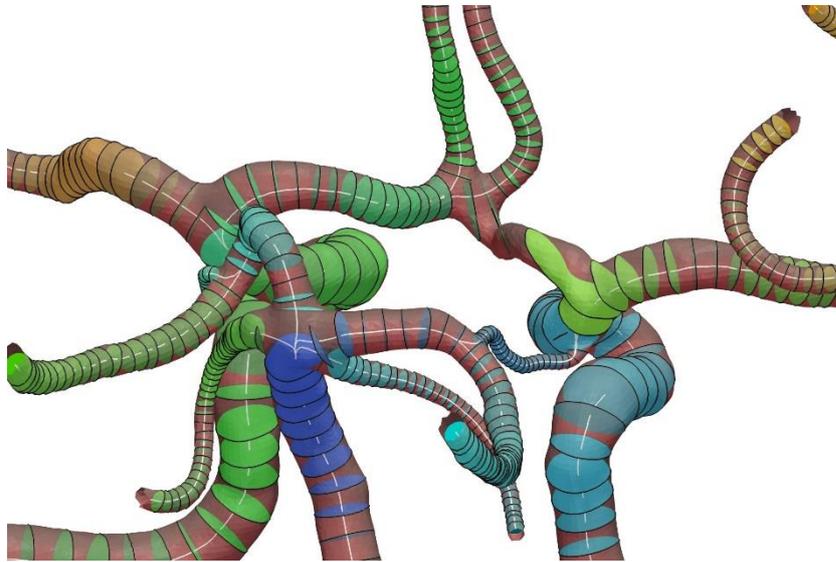


Figure 1.23. Extraction of vessel dimensions for the circle of Willis in volunteer reference data. Note that the RComA is missing a centreline; this is because VMTK cannot natively handle closed vessel networks such as the circle of Willis. To process all vessels of the circle of Willis in the volunteer mesh, the mesh was processed twice using different inlet vessels as seed locations.

The phantom mesh measurements are provided in Table 1.6.

## Section 1

Table 1.6 Measurements of phantom mesh vessel length and diameter using VMTK. LMCa does not branch, so that and LMCab are not measured. The RMC bifurcates into three vessels, so RMCa is ignored, leaving only RMCa and RMCb instead.

Artery	Length (mm)	Diameter (mm)
LIC1	100.2	4.49
LIC2	0.6	3.87
LMC	29.7	2.67
LMCa	N/A	N/A
LMCb	27.4	2.09
LAC1	18.7	2.70
RIC1	95.9	4.09
RIC2	0.7	3.60
RMC	35.5	2.65
RMCa	N/A	N/A
RMCb	1.0	2.51
RAC1	19.0	2.09
LVA	92.7	3.14
RVA	83.9	3.25
BA	23.8	3.54
LPC1	14.9	2.40
RPC1	9.6	2.57
ACo	N/A	N/A
LPCo	9.8	1.02
RPCo	12.4	1.02
LMCaa	31.6	1.46
LMCab	N/A	N/A
LMCba	13.3	1.62
LMCbb	22.2	1.52
LAC2	21.2	1.81
RMCAA	22.8	1.41
RMcab	47.6	1.65
RMcba	25.4	1.55
RMcbb	27.0	1.34
RAC2	17.2	1.83
LPC2	22.8	2.09
LSC	22.0	1.46
RPC2	26.1	1.95
RSC	20.7	1.32

Measurement of vessel geometry with VMTK produces more robust and accurate measurements than manual measurement in medical image viewers,

## *Section 1*

or in this case by measuring the phantom physically. In the case of the phantom, it is possible to measure vessel length externally, but difficult to measure internal diameter.

A weakness of this approach is averaging all cross-sections to calculate the diameter. As is apparent in Figure 1.23 **Error! Reference source not found.**, some cross-sections are placed poorly and are either markedly undersized or oversized, these cross-sections will skew the average.

### **Semi-automation of VMTK processing**

The process described above can take a substantial time to complete due to slow computation of some steps (around 30 minutes for `vmtkbranchsections`), having to repeat the process if the vascular network includes a closed loop, and manual processing of the output data. In this section we consider automating some of the steps to speed up analysis.

The process was broken down into 5 scripts, each of which describe their usage to the user when user input is required. The scripts were primarily tailored for extraction of MCA diameters since this was of interest for other research, which is further discussed in section 2. The scripts are separated so that each step can be run separately in case the user wishes to redo an operation. The scripts are described below in order of usage:

1. `convert.py`
  - a. Converts a DICOM angiogram into a vti file, which is the principle imaging file format used by VMTK.
2. `segmentation.py`
  - a. Segments the angiogram using the VMTK level sets algorithm, then constructs a surface from the level sets using the VMTK marching cubes algorithm.

## Section 1

- b. The script suggests to the user which surface construction algorithm and thresholds to use, as well as the level sets threshold, but these can be adjusted if needed.
    - c. The script then asks the user to place “seeds” and “targets” for the marching cubes algorithm at the ICA bifurcation and MCA ends respectively. The user is shown a 3D view of planes in the angiogram and can click on the appropriate areas.
  3. clipmesh.py (optional)
    - a. This script is used if the patient has a complete circle of Willis, which results in a closed loop which VMTK is unable to process correctly.
    - b. The script shows the VMTK surface clipper tool to the user which is used to “cut” a vessel to break the closed loop.
    - c. Even with a closed loop, VMTK may still process the MCAs without issue, in which case this script does not need to be used.
  4. geometry.py
    - a. This script processes the segmented vasculature surface to trace centrelines through the vessels, divide the centerlines into branches, and calculate cross sections through all branches.
    - b. The user is asked to place a single “seed” on one ICA shortly before the ICA bifurcation and “targets” on the other ICA at the same position, at the end of all branches from MCAs, and a short way up each ACA. These points were chosen to produce consistent results in a timely manner. Computing centerlines for the whole vasculature is slow and unnecessary if only the MCAs are of interest.
  5. displaygeometry.py
    - a. Displays a 3D visualisation of the vasculature with branches labelled with group numbers.

## Section 1

- b. This script is used to find the groups comprising each MCA, which is used as input in the next script.
6. branchdiameters.py
- a. Extracts branch diameters from the VMTK cross-section data output.
  - b. The user must specify which group IDs belong to each MCA, these should be found using displaygeometry.py to visualise the groups.
  - c. The diameter of each MCA is calculated by averaging all cross-sectional areas included in the chosen groups (see Figure 1.24).

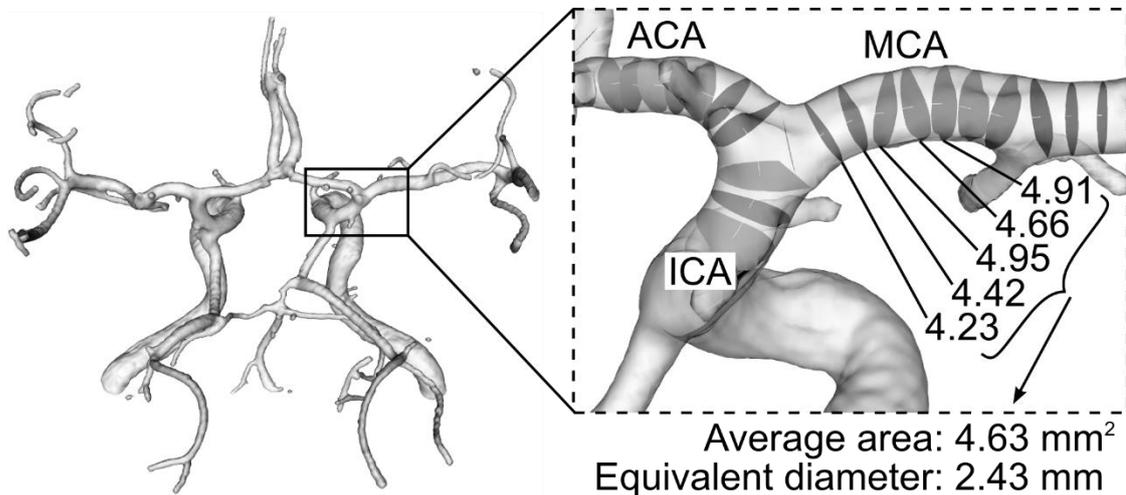


Figure 1.24 Calculating MCA diameter by averaging cross sectional areas computed by VMTK.

See Appendix A for the full scripts discussed in this section.

### 1.7.2 Acquisition of patient data

Patients were imaged using a 3T MRI scanner (MAGNETOM Skyra, Siemens) using a head coil. Several sequences were used to capture T1 and T2 weighted images, as well as time of flight angiograms and Fluid-Attenuated Inversion Recover (FLAIR) images. For vascular segmentation, only the time of flight angiogram is of interest, where the pixel size was 0.26 x 0.26 mm<sup>2</sup>, repetition

## Section 1

time 21 ms and echo time 3.43 ms. Contrast agents were not used in any of the image sequences.

### 1.7.3 Segmenting MR angiograms for 47 patients

The script described in section 1.7.1 was applied to 47 patients who received MR imaging as part of a British Heart Foundation study investigating brain injury following cardiac interventions (the BICI study). All clinical data were collected following an ethically approved protocol (REC reference: 10/H0401/78, University Hospitals of Leicester NHS Trust and Derbyshire Research Ethics Committee).

Figure 1.25 shows a selection of the cerebral vasculatures for cardiac surgery patients recruited to this study. It is clear that anatomy of the circle of Willis varies widely between patients.

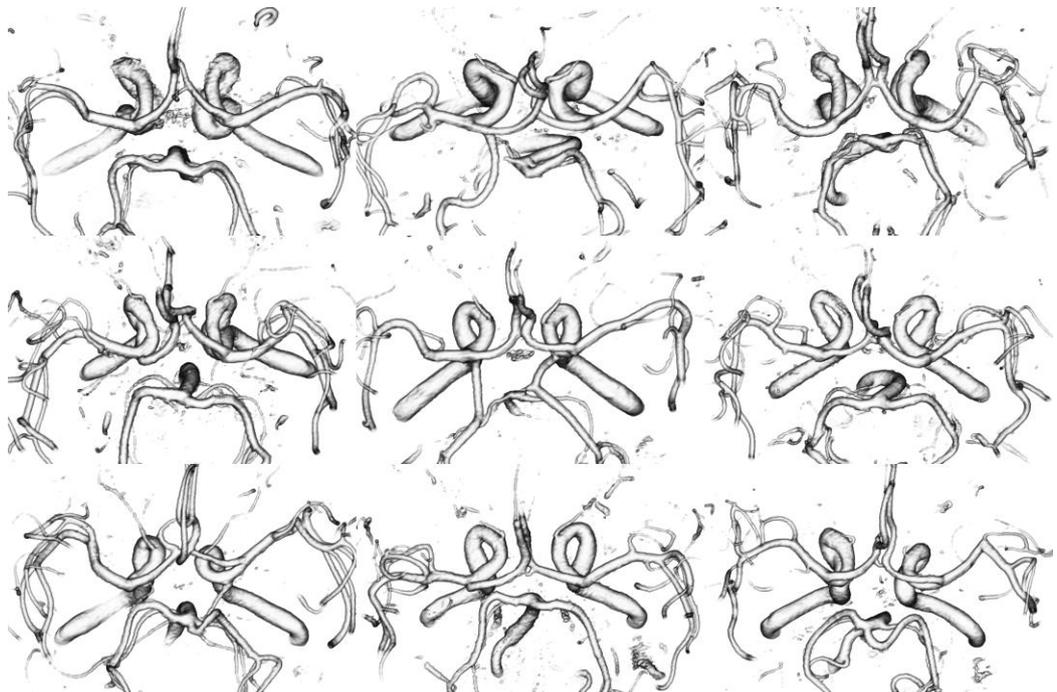


Figure 1.25. Time-of-flight MR angiograms of the circle of Willis in nine patients. Volumetric rendering in 3D Slicer (3D Slicer; Fedorov et al., 2012) was used to visualise the vessels.

## *Section 1*

### **1.7.4 Categorising anatomical variants**

The anatomy of the cerebral vasculature was highly variable among the patients imaged in this study. A number of variants of the circle of Willis are highlighted and discussed in this section. We demonstrate six of the seven circle of Willis variations studied by Alastruey et al. (2007), the seventh variation is an absent AComA which is only present in around 1% of the population (Lippert and Pabst, 1985) and was not found among the patients in this study.

The vasculatures presented are from patients with existing cardiovascular disease and may not be representative of vasculatures in the healthy population.

The volumetric images of the MR angiograms were created using 3D Slicer (3D Slicer; Fedorov et al., 2012). VMTK was not used for this part of the study since volume rendering is often better for displaying the anatomy than segmentation, whereas segmentation allows quantitative analysis.

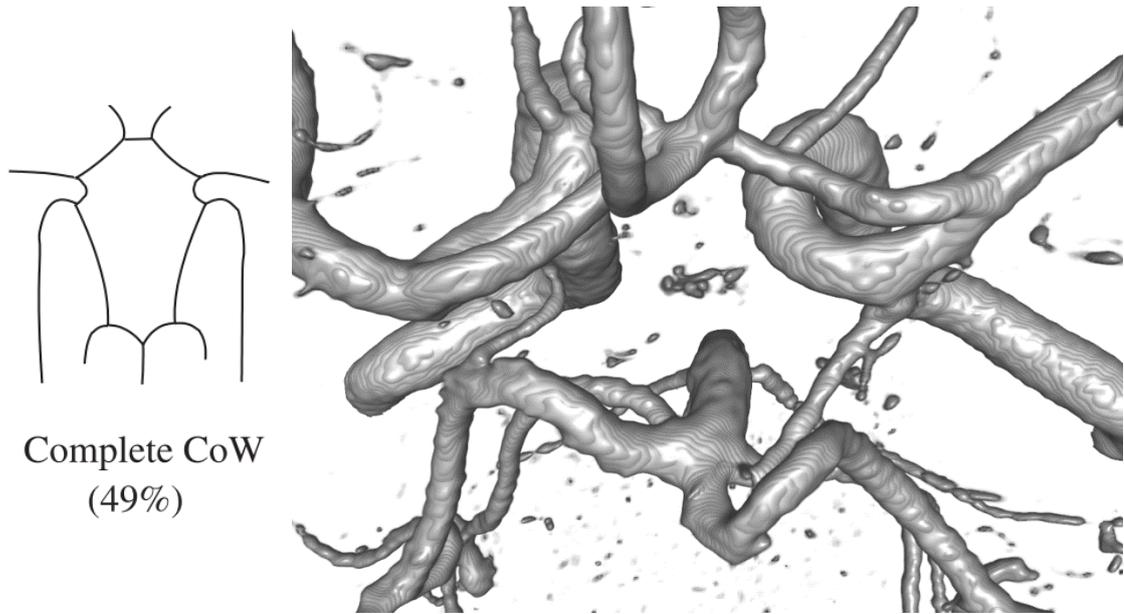
Figure 1.26 shows a patient with a complete circle of Willis. Figure 1.27 shows a patient with an absent RComA. Figure 1.28 shows a patient with both PComAs absent. Figure 1.29 shows a patient with an absent LPCA1. Figure 1.30 shows a patient with an absent LComA. Figure 1.31 shows a patient with an absent RACA1.

Viewing MR angiograms for this number of patients demonstrated the limitations of time-of-flight angiography. Many patients appeared to be missing both PComAs, far more than would be expected from the 9% prevalence given by Lippert and Pabst, 1985 compared with the 49% prevalence of a full circle. It may be that these patient had fetaltype communicating arteries, but it seems more likely that they are simply not imaged using this method since it relies on flow through vessels during

*Section 1*

imaging. Small PComAs may not have substantial flow. In addition, use of contrast agents would improve sensitivity but were not used during this study.

Section 1



Complete CoW  
(49%)

Figure 1.26 Patient with a complete circle of Willis. Figure on left is from Alastruey et al. (2007), the percentage incidence shown in brackets is from Lippert and Pabst (1985).



PCoA  
(9%)

Figure 1.27 Patient with an absent RComA. A single absent PCoA does not significantly affect the collateral flow capability of the CoW since the contralateral PCoA can provide collateral flow. The LComA is marginally larger than the LPCA1, so the LPCA territory is likely supplied from the LICA as well as the BA. Figure on left is from Alastruey et al. (2007), the percentage incidence shown in brackets is from Lippert and Pabst (1985).

Section 1

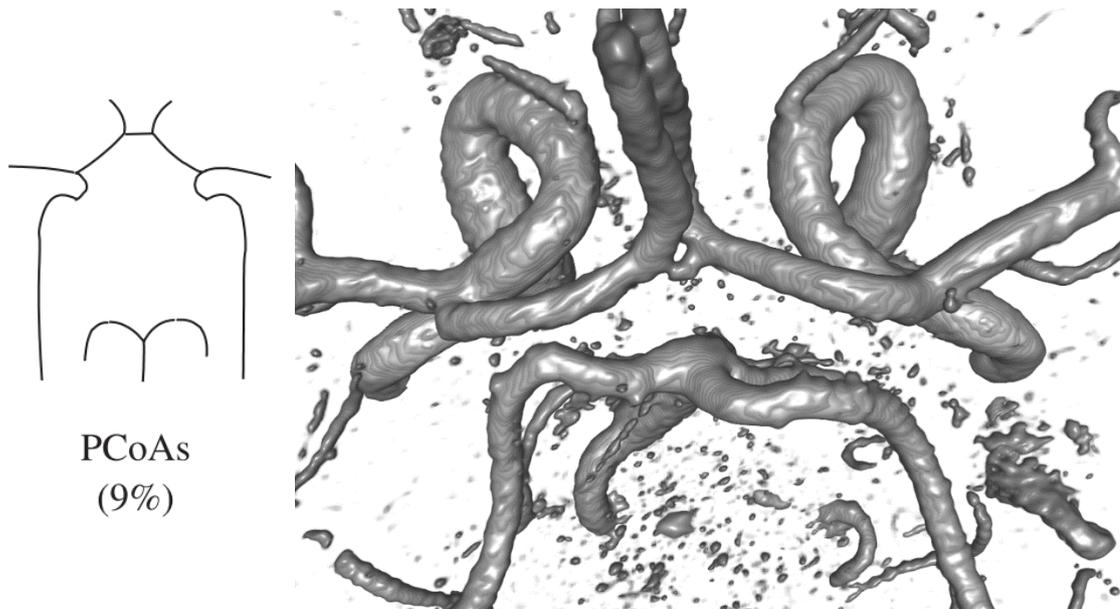


Figure 1.28 Patient with both PComAs absent. This anatomy cannot provide collateral flow between the posterior and anterior vessels, this only becomes an issue in case of occlusion of both ICAs or VAs. Figure on left is from Alastruey et al. (2007), the percentage incidence shown in brackets is from Lippert and Pabst (1985).



Figure 1.29 Patient with an absent LPCA1. In this configuration the LPCA territory is fully supplied from the LICA. In case of LICA occlusion, flow to the LPCA and LMCA territories can still be maintained through the AComA; although, for this particular patient, the LACA1 is small and may not have capacity to carry sufficient flow. Figure on left is from Alastruey et al. (2007), the percentage incidence shown in brackets is from Lippert and Pabst (1985).

Section 1

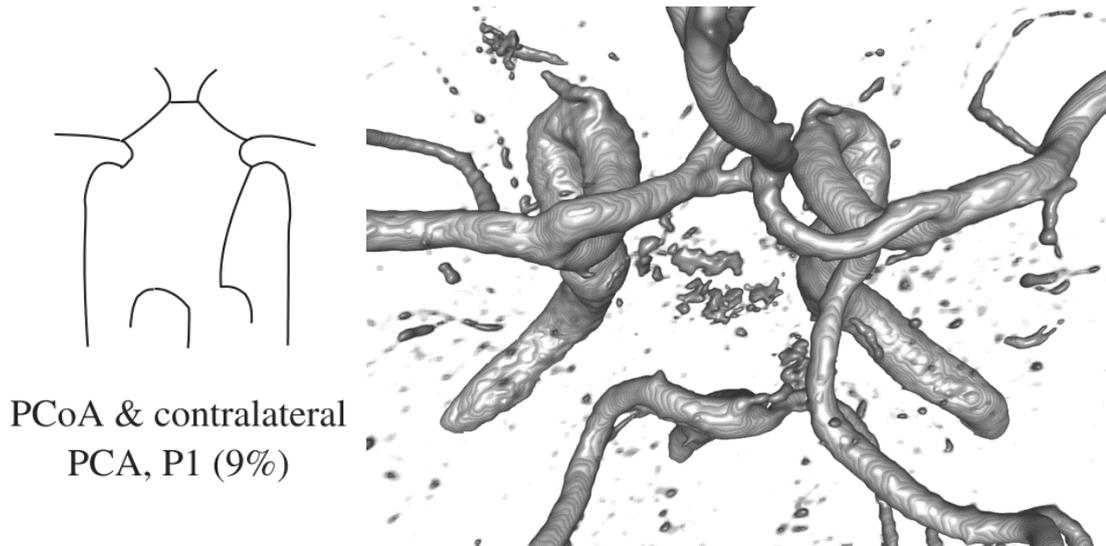


Figure 1.30 Patient with absent LPCoMA and fetaltype RPCA1 (narrow and cannot carry significant flow). This anatomy uses the RICA to fully supply the RPCA territory. There are no anterior-posterior collateral vessels to supply flow in case of complete occlusion of the ICAs or VAs. Figure on left is from Alastruey et al. (2007), the percentage incidence shown in brackets is from Lippert and Pabst (1985).

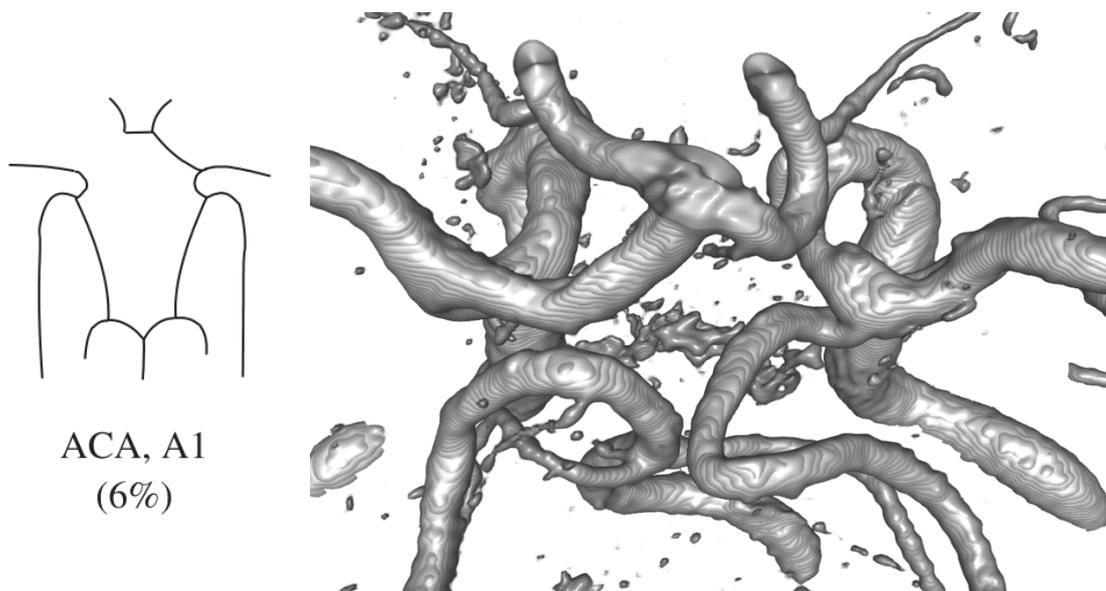


Figure 1.31 Patient with absent RACA1. This patient also has absent RPCA1 and LPCoMA. The omission of an ACA1 segment results in the anterior territory being fully supplied by a single ICA. This can become a problem if that ICA becomes occluded. For this particular patient, the absent LPCoMA means that LICA occlusion would prevent flow to the anterior territory, and the absent RPCA1 means that RICA occlusion would prevent flow to the RMCA and RPCA territories. This patient has no collateral capability in case of occlusion of either carotid. Figure on left is from Alastruey et al. (2007), the percentage incidence shown in brackets is from Lippert and Pabst (1985).

## *Section 1*

### **1.7.5 Discussion**

VMTK was a good tool for segmenting and analysing patient data since its versatile interface allowed customised scripts to be created which could extract specific geometric information with a minimum of user input. The scripts created in this section are applied to provide robust measurement of patient MCA diameters in section 2.

The script described only measures left and right MCA diameter. This could be extended to all vessels but restrictions of using VMTK with the closed loop of the circle of Willis makes the process more labour intensive. Measurements always require manual input of branch identifiers by a human since the scripts are unable to automatically determine which branch is which part of the cerebral vasculature.

In future work the mesh extraction and analysis scripts developed here could be refined to allow extraction of more information, to create patient specific meshes in a more automated manner, and include batch processing of data from multiple individuals.

The patient data collected demonstrated many variations of the circle of Willis. The images shown in this section demonstrate the detail which can be captured using MR angiography without contrast agents. Omitting contrast agents from the study means that patients do not require intravenous injection during imaging, which reduces the time taken to capture a full set of MR sequences from each patient, as well as improving likelihood of patient participation in the study (since all methods are non-invasive).

## **1.8 0D modelling of patient-specific cerebral blood flow**

In this section, a 0D model is applied to the cerebral vasculature of a single patient. The vessel dimensions for that patient were extracted as described in the previous section. The generic 0D model was then adapted to predict normal flows, as well as flows when a vessel is occluded.

This case study is provided as a proof of concept study to demonstrate how segmentation and analysis of vessel geometry can be used to non-invasively predict flows in a patient specific model under physiological and pathological conditions.

### **1.8.1 Calculating flow in the cerebral vasculature for a single patient**

The 0D model was applied to the patient vessel anatomy shown in Figure 1.32. Vessel dimensions for the patient and calculated flows in the linear and non-linear 0D model are given in Table 1.5.

Section 1

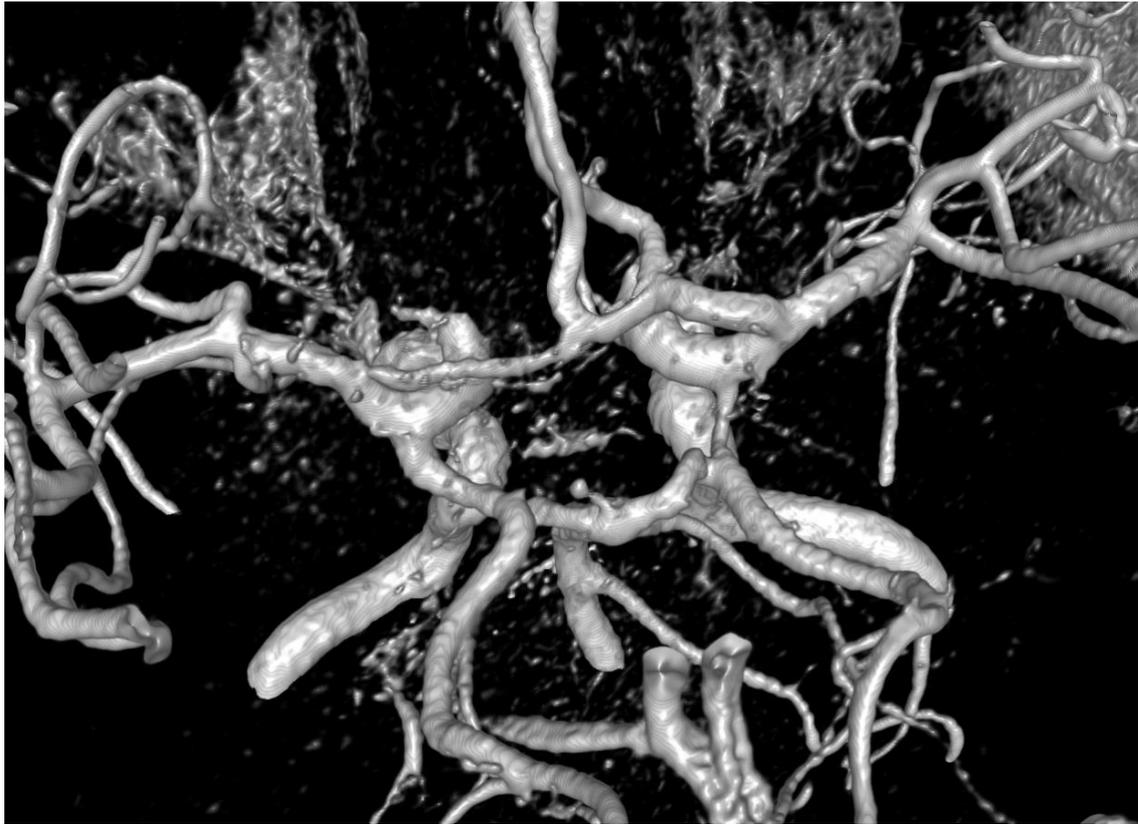


Figure 1.32 Patient circle of Willis. This patient has a full circle of Willis. Note that the LPCoM<sub>A</sub> is large compared to the RPCoM<sub>A</sub>, and similarly the RACA<sub>1</sub> segment is large compared to the LACA<sub>1</sub> segment. Assuming that vessel size correlates with flow volume, it would appear that the patient's left-posterior blood supply would be mainly supplied by the LICA, and the left-anterior blood supply supplied predominantly by the RICA. In the event of occlusion of a carotid artery we would expect good collateral flow in this patient.

Table 1.7 Flow estimates from the OD model for a cardiac surgery patient with all vessels open and with the left internal carotid artery occluded.

Vessel	Physiological flows (ml/min)		LICA Occlusion flows (ml/min)	
	Linear	Non-linear	Linear	Non-linear
RMCA	96	98	98	100
LMCA	104	110	96	93
RACA	82	73	85	86
LACA	67	64	69	72
RPCA	59	55	63	64
LPCA	52	60	49	45
Total	460	460	460	460

## Section 1

### 1.8.2 Modelling the effect of carotid stenosis on patient-specific cerebral blood flow.

The results of occluding the LICA are shown in Table 1.5 and Figure 1.17.

As expected, significant flow was supplied to the LPCA from the LICA, so flow to the vessel decreases when the LPCA is blocked. The LMCA flow also decreases. Due to this patient's complete circle of Willis, flows are virtually unchanged. Assuming that the patient's autoregulation response was not impaired, autoregulation should be capable of compensating for the occlusion.

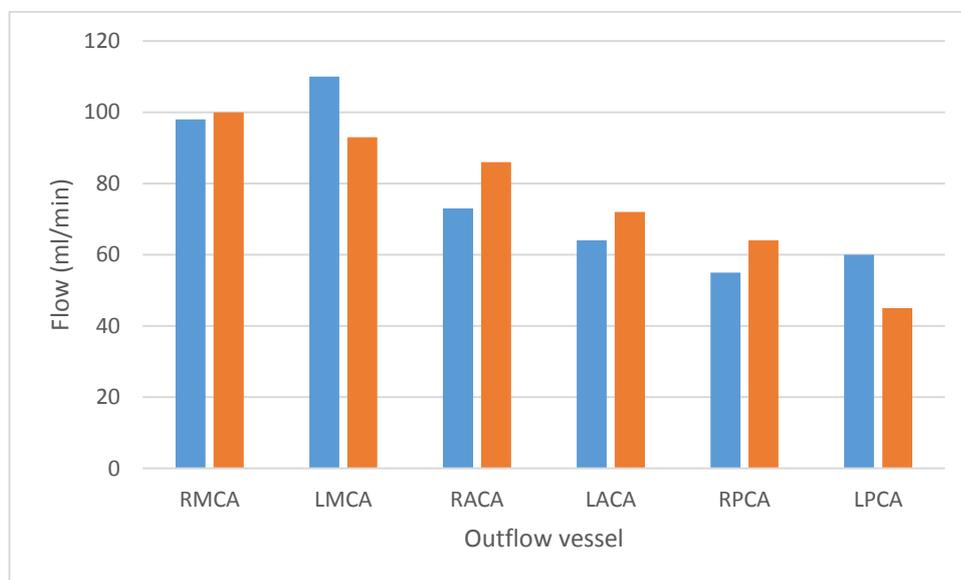


Figure 1.33 Change in vessel outflows in LICA occluded and non-occluded cases. Physiological flows are in blue and pathological flows in orange. Both data sets are from the non-linear 0D model. The data shows a decrease in flow to LPCA and LMCA due to an LICA occlusion.

### 1.8.3 Discussion of 0D modelling of a patient.

In this section we have modelled flow in the cerebral vasculature of a patient using a 0D model.

Vessels were measured using the ruler tool in 3D slicer (3D Slicer; Fedorov et al., 2012) on slices through vessels. This has some weaknesses since the measurement is taken in one position rather than averaged along the vessel. In

## *Section 1*

addition, error is introduced by the sensitivity and resolution of the MRI, as well as the thresholding used to visualise the vessel. Choice of threshold value has a significant impact on the apparent size of arteries, so it is important to keep the threshold constant for all vessels for consistency. The patient has a large LPCoM and small LACA1, but otherwise has a complete circle of Willis, from which we expected good collateral flow in the event of carotid occlusion.

Physiological flows calculated by the 0D model are similar to those reported by Alastruey et al. (2007), and also flows in the flow phantom. The main differences are due to addition of the SCA in the 0D model, which results in increased flow to the posterior vessels. It is good to see consistency here, because unlike the phantom, which used efferent resistances from Alastruey et al., for the patient we calculated resistances from the distribution of flow expected from the size of the peripheral vessels in the MR angiogram.

The effect of occlusion was to change the flow distribution, but not total flow. Given that the net resistance of the modelled vessels is not significantly altered by occlusion of the LICA, we expect the same pressure differential to generate roughly the same total flow. In the model, this results in any reductions in outflows to be accompanied by increases in outflow from other vessels. In the real case, autoregulation would be a significant factor in managing these flow changes. Because of this, the model is useful for studying the effect of occlusion should autoregulation be impaired, and also to study the burden applied by occlusion of the vessel which must be accommodated. If flows became too low in the model, autoregulation mechanisms would struggle to compensate. In summary, given the simplicity of our model the results are still insightful.

Based on our simulation results we can expect that the patient under study may not suffer any deleterious effects from occlusion of the LICA due to a build-up of carotid plaque. It is interesting to note, however, that even with a complete circle of Willis, occlusion will still result in flow changes.

### *Section 1*

Data similar to these could be used as inputs for other models such as fractal-tree models of arterioles and capillaries (see section 2). The distribution of flow may also be of interest when studying embolisation, which is discussed more fully in the next section.

## **1.9 Discussion**

In section 1 I discussed flow in the circle of Willis and cerebral vasculature. I looked at the anatomy of the circle, computational and physical modelling of flow in the circle, and moved on to implementing my own 0D model. A flow phantom was created based on a MRI captured from a healthy volunteer, and was validated against a 0D computational flow model, and through imaging the phantom with MRI. We discussed angiogram segmentation and applied this to a large set of patient data, using our results to analyse anatomical variants. The 0D model was then applied to patient data as a proof of concept case study to illustrate how angiograms could be used to estimate flows and predict the effects of vessel occlusion.

In section 1.6 we confirmed that the phantom and flow rig showed good agreement with population physiological data, By taking an MRI of the phantom itself, we confirmed that the phantom geometry was comparable to the healthy volunteer's original scan. The phantom reflected real anatomy, and was suitable for fluid flow, MRI and ultrasound studies. Implementaion of the phantom was both guided by, and used to validate a 0D flow model, so that the 0D model can be used to analyse patient data with confidence.

In section 1.7, a method to segment the cerebral vessels of a large number of patients was implemented. This produced data suitable for quantifying morphological variations in the circle of Willis.

In section 1.8, the 0D model validated in section 1.6 was applied to patient data. This proof of concept study showed that using an MR angiogram, and semi-automated processing of the imaged vasculature, it becomes possible to make predictions about bulk flows in the cerebral vessels of an individual patient. To demonstrate the usefulness of this sort of model, the effects of carotid artery occlusion were studied.

## *Section 1*

As discussed in the introduction in section 1.1, modelling blood flow can be approached in many ways, and requires a combination of techniques to produce high quality, useful data. The techniques developed in this section demonstrate how these methods can be applied to produce useful data, in particular, when applied to patient-specific problems.

In addition to the techniques covered here, other methods of studying blood flow, such as phase-contrast MRI, which provides information on direction of flow in vessels, and ultrasound, can be used to investigate cerebral blood flow.

Since this section was largely concerned with development, rather than application of techniques, future work on this topic would include application of these techniques to new problems and data. Later in this thesis, the segmentation technique will be applied to analyse patient vessel geometry as inputs to computational models of emboli. In addition, applying these techniques to provide a better understanding of vascular pathologies and patient-specific carotid plaque development could be an interesting subject for further research.

# **Section 2. Development of Patient Specific Simulations of Embolisation**

## **2.1 Introduction**

Cardiovascular procedures such as coronary artery bypass surgery and cardiac valve surgery carry a high risk of cognitive decline detected by neuropsychological testing, and can also result in new ischemic lesions seen on MRI. (Barak & Katz, 2005; Nietlispach et al., 2010) Although new focal brain injuries following surgery, and many cases of stroke, are thought to result from solid emboli entering the brain circulation during surgery, whether there is a relationship between air bubbles entering the circulation and cognitive decline remains unclear. Understanding the adverse effects of both solid and gas emboli would allow surgeons to better appreciate the risks associated with differing stages of the procedure, providing opportunities to improve patient outcome. If, for instance, emboli or other factors were found to significantly contribute to neurocognitive decline, this may drive the development of new technologies or guidelines. Novel neuroprotection technologies may be introduced to reduce the probability of emboli passing to the brain by filtering the blood (Baumbach et al., 2015) or by deflecting emboli to less sensitive organs (Nietlispach et al., 2010). Patient-specific anatomy, such as the geometry and topology of blood vessels, and identification of areas of calcification or disease, may prove useful for guiding insertion of cannulae and catheters during cardiovascular interventions, and could be used to predict the trajectories of cardiac emboli in modelling stroke.

## *Section 2*

The purpose of modelling embolic stroke is to better understand how the physical behaviour of emboli leads to deleterious effects in patients. It is rational to expect that solid emboli and bubbles of sufficient volume will inevitably occlude vessels somewhere downstream in a bifurcating vasculature; it is less simple to determine the location, magnitude and negative effects of these occlusions. The source of emboli may be pathological (e.g. atherosclerosis) or associated with an intervention (e.g. vascular surgery). Understanding the relationship between the origins and final resting places of emboli may help clinicians to understand the aetiology of stroke, pinpointing the source of embolisation, and determining patient-specific risk factors. With a better understanding of embolisation, we hope that patient specific simulation of embolisation can be used to improve outcomes, and lead to the development of new technologies to prevent or mitigate the damage caused by embolism. This section focuses on patient specific simulations to estimate the impact of emboli during heart surgery.

### **2.1.1 Models of embolus transport**

Emboli in the blood have been modelled in various ways. Physical replicas of bifurcations and whole vascular structures have been used, with emboli inserted upstream to study their motion and distribution among the outflow vessels. These models can test the premise that emboli are more likely to travel along certain paths compared to others, depending on patient specific vascular anatomy. This is of interest in stroke modelling, where it is found that a larger proportion of lesions appear in areas supplied by the middle cerebral artery of the brain than would be expected by considering the proportion of flow received. Anatomical replicas of the major arteries have been used to reproduce the distribution of emboli seen in TCD monitoring of patients during surgery (Chung et al., 2010). In laboratory embolisation studies, the fluid used may simply be water, or a water-glycol mixture with similar viscosity and density as

## Section 2

blood. This allows visualisation of embolus trajectory in the flow using a high speed camera. The phantom itself may be created from medical imaging data, as described in section 1 (p ...). Resistance to flow downstream of the phantom outlets is emulated using long tubes to represent the peripheral vessels and capillary bed. The length of tubes is varied between outlets to reflect the proportion of flow and blood pressure expected at each location.

Emboli transport has also been modelled *in silico* (computationally) using a wide variety of computational fluid dynamics (CFD) techniques. Depending on the way the fluid is considered mathematically, these can be divided into Eulerian and Lagrangian methods. Eulerian methods consider flow as moving through a static reference frame and include finite difference, element and volume methods, and the spectral method. Lagrangian methods consider flow from a reference frame moving with elements of the flow, and include smoothed-particle hydrodynamics (SPH). Simulations may be 2D or 3D; 2D models are limited in scale to small vessel branch sections but are usually easier to implement. The scale of 3D simulations range from small idealised models of straight vessels and single branches, to vessels generated from imaging data including single anatomical bifurcations and whole systems of vessels such as the cerebral vasculature and heart. Typically, emboli are modelled to study occlusion of vessels (Vahidi and Fatourae, 2012), the microscopic interactions of emboli with the vessel wall (Gholami et al. (2014)), and the path taken by emboli travelling through the system (which is of most interest to us) (Mukherjee et al., 2016 (1)).

Occlusion by gaseous emboli (usually air bubbles) is more complex than for solid emboli. Bubbles larger than the diameter of the vessel become elongated along the vessel, until resistance on the bubble due to contact with the vessel wall ('stiction') overcomes the pressure forces either side of the bubble at the two ends. At this point the bubble will block the vessel until it dissolves

## *Section 2*

further. This can be modelled on a large scale across millions of bifurcating vessels using a recursive computational approach (Hague et al., 2013). By modelling vessels as simplified straight tubes with a linear relationship between flow, pressure and resistance; vessels can be recursively lumped together to study the bulk behaviour of many bubbles introduced into a vascular system. The bubbles are introduced sequentially and propagate down through the tree until blocking a vessel. The large number of emboli introduced, and random travel, leads to a bulk description of the effects of embolisation referred to as a Monte Carlo embolisation model. Embolus data generated from embolic signals recorded during surgery can be input to the model, to assess the level of occlusion for a particular patient. This is largely of interest for assessing whether bubbles introduced to the brain circulation during surgery are likely to be clinically significant.

Atherosclerosis is a disease of the vessel wall characterised by a build-up of plaque; raised areas of the vessel wall comprising active white blood cells, lipids, and dead cells. The surface of the plaque is covered in a layer of endothelial cells, which has potential to rupture, leading to thrombus formation and the release of plaque debris. Thrombus and plaque pieces can then travel downstream and occlude vessels supplying the brain resulting in sub-clinical ischemic lesions, transient ischemic attack (mini-strokes), or stroke. Modelling flow through atherosclerotic vessels can be informative. Wall-shear stress described forces parallel to the direction of flow exerted on the vessel wall by viscous action of passing blood. Low wall-shear stress is associated with plaque formation, while high wall-shear stress is associated with plaque rupture. Modelling flow to assess wall-shear stress can be useful in assessing plaque formation and stability. Thrombus formation may be modelled to understand how thrombus forms in flowing blood and to determine the point at which a piece of thrombus will break away from the plaque. Vessels can be

## *Section 2*

reconstructed for modelling using imaging from ultrasound, MRI or CT. The reconstructed vessel may be either 2D or 3D.

Here I review literature presenting computational models of embolism. These divide into research modelling embolus motion using computational fluid dynamics applied to large vessels only, and research modelling the bulk behaviour of emboli in large trees with reduced models. The literature available on both areas is limited, and to the best of my knowledge this review below covers all existing journal articles (excluding conference proceedings) on computational modelling of emboli.

### **2.1.2 Non-invasive detection and sizing of cerebral emboli using ultrasound**

Recent breakthroughs in embolus sizing make it possible to size emboli in the cerebral arteries using transcranial Doppler ultrasound measurements (Banahan et al., 2012). This can be applied to surgical procedures that are known to produce air and/or solid emboli such as cardiac surgery, catheter ablation, and carotid endarterectomy. The method requires TCD recordings to be made during surgery which can be analysed later.

In this section, I first provide an overview of current techniques for modelling emboli moving through the cerebral circulation. I then go on to describe methods combining embolus sizing with a fractal tree simulation applied to patient data collected during cardiac surgery.

## **2.2 Literature review of computational modelling of cerebral emboli**

### **2.2.1 The stenosed carotid**

Vahidi and Fatourae (2012) study the motion of an embolus passing through a stenosed common carotid artery using the arbitrary Lagrangian-Eulerian (ALE) finite element method to model the fluid, with the thrombus modelled by a Sussman-Bathe hyperelastic model. The authors consider an embolus formed upstream of the carotid, which then passes through a stenosis, perhaps created through thrombogenesis at the cardiac wall. This 2-dimensional simulation considers a straight tube (infinite parallel plates in 2D) with stenosis reducing the vessel diameter by 40-50%; a large clot (2-4mm), which starts as a sphere (infinite cylinder in 2D), is released into the flow and allowed to deform. The thrombus is fixed in place while the flow reaches a steady state before it is released. The parameters of the thrombus model were fitted to stress/strain values measured from a real thrombus. The modelled flow without an embolus was validated against previous results. Thrombi deformed significantly as they passed through the stenosis (Figure 1), which is interesting as the trajectory of the thrombus could potentially be altered by deformation. The stresses acting on the thrombus and wall were also assessed, but the main success of this study was that it provided a proof of concept for modelling embolus passage through a complex geometry, with a view to advancing this toward patient-specific modelling. The authors state that there are no experimental results comparable with their computational model due to the small scale of the system and limitations of existing techniques for measuring fluid and embolus physical properties experimentally. This highlights the advantage of computational simulations of this nature, tackling problems which cannot be addressed experimentally.

## Section 2

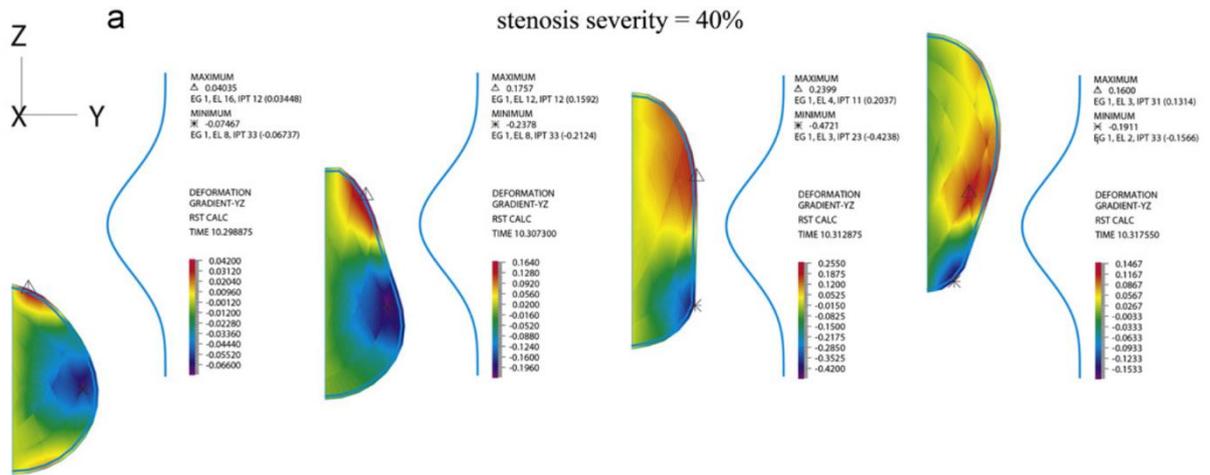


Figure 1. Simulated embolus, deforming as it passes through a stenosis. (Vahidi and Fatourae, 2012)

### 2.2.2 The carotid bifurcation

The work of Vahidi and Fatourae (2012) was extended by Abolfazli et al. (2014) using the same methods, and models a thrombus passing through a 2D bifurcation. The bifurcation angle is varied, as is the thrombus release position relative to the vessel wall. The embolus velocity is dependent on its size and density, but this could be because the embolus is released from a stationary position. Velocity graphs show that the embolus does not reach a stable velocity before coming to the bifurcation, so it is possible that only the embolus acceleration is affected, not its velocity relative to the blood. A 3D simulation was also performed for a carotid bifurcation reconstructed from a CT angiogram, however, quantitative results are not presented. The distribution of emboli between the ICA and ECA for the 2D case was 59:41, averaged over various bifurcation angles and release positions. The flow distribution between branches and the relationship between embolus trajectory and release point is not given, which makes interpretation of this research difficult. This study mainly focuses on stress on the embolus and wall, largely due to collisions between the embolus and the bifurcation point on the vessel wall. Stress in this scenario is not of great interest unless the embolus deforms, leading to a change

## *Section 2*

of trajectory. Any stress applied to the wall can be considered insignificant compared with the location of embolism and resultant ischemia. Wall shear stress estimates may also be inflated due to modelling the vessel wall as a rigid edge, and not considering the probability that an embolus would impact the bifurcation point.

Mukherjee et al. (2016 (1)) also model embolus passage through the carotid bifurcation. A 3D bifurcation is reconstructed from a CT angiogram. A coupled Euler-Lagrange simulation is used, flow is pulsatile and solved using the finite volume method; embolus motion is modelled using a modified Maxey-Riley equation. Outlets are modelled as lumped resistances and total flow through the daughter branches assumed to be proportional to the cross sectional area of those branches. Emboli are spherical and range in size up to 17% of the model ICA diameter. The coupling framework used requires solids in the flow to be no larger than this. For each case, 5000 emboli are released independently into the flow at a range of positions covering the inlet. The authors relate embolus size to embolus division compared with flow division between branches for both a one-way and two-way coupled technique (the one-way coupled technique allows emboli to be added to a pre-generated CFD simulation since the embolus itself does not influence the flow). Mukherjee et al. found that embolus division deviated from the flow significantly more in the anatomical vessels compared with the idealised model. For one size of particles the embolus division was 50:50 compared with 60:40 for the flow. The authors investigated this further and concluded that helical flow was important in skewing the preferred trajectory of emboli between branches. They also found that embolus trajectory depended heavily on the time in the pulsatile flow cycle at which the embolus was released (Figure 2). Because of this, the authors note that analysing how particle trajectories correlate with flow division alone is insufficient when trying to better understand stroke and stroke risk factors.

## Section 2

Also that trying to model embolus transport with idealised geometry in-vitro or in-silico will not be accurate due to lack of swirling flow and vessel curvature. This may not hold true for vessels downstream with steadier flow.

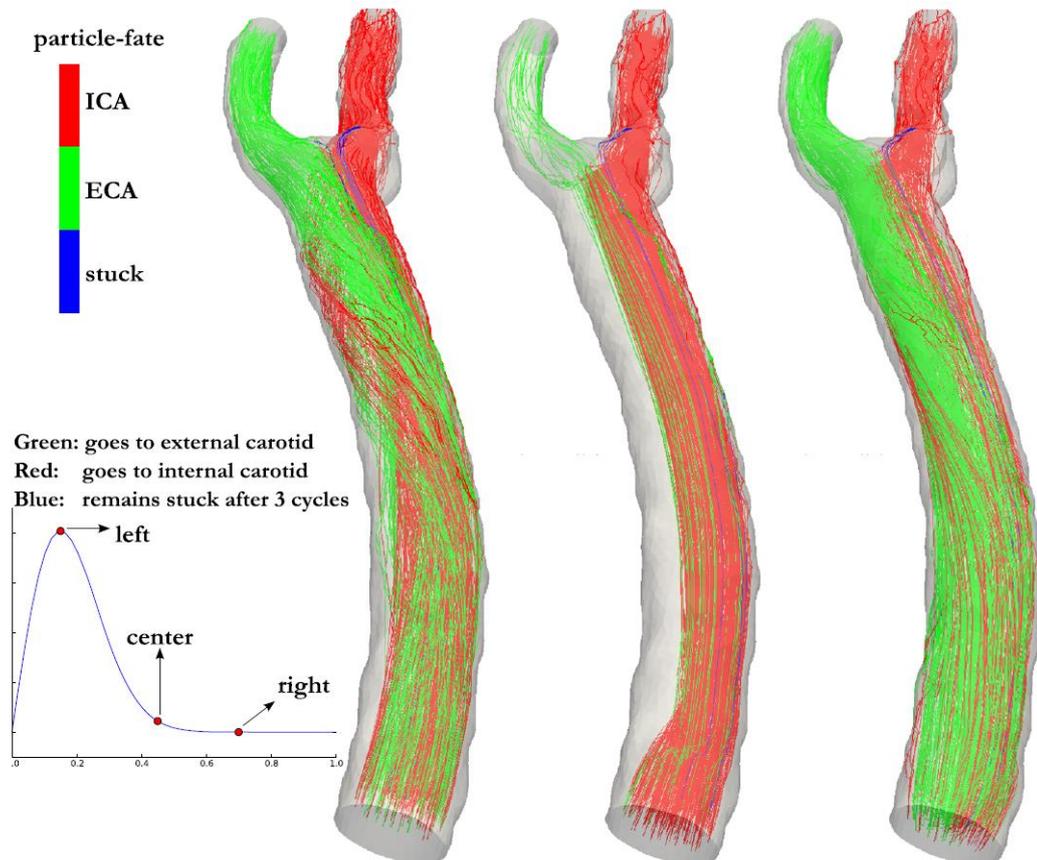


Figure 2. Distribution of emboli across the carotid bifurcation released at three times during the cardiac cycle. Trajectory depends heavily on release time. (Mukherjee et al., 2016 (1))

### 2.2.3 The aortic arch

Carr et al. (2013) have modelled the trajectory of cardiogenic emboli passing through the aortic arch. Flow was modelled using the finite element method with a solver adapted from that used in SimVascular. The authors aimed to determine the effect of embolus size on the number of emboli passing through to the left and right arteries leading to the head. 10 patients were modelled. Reconstruction was performed using SimVascular. Pulsatile flow was modelled and outlet boundaries were modelled with lumped resistances such that flow through each branch was proportional to its cross-sectional area. Model emboli

## *Section 2*

were spherical and coupled to the fluid using the Maxey-Riley equation with one-way coupling (fluid influences emboli). Collisions with the embolus-wall were modelled. Emboli were injected assuming the inlet velocities of the blood were spread uniformly across the vessel cross-section. Model emboli had diameters ranging between 0-4 mm. This study found that small to medium ( $\leq 1$  mm) emboli travelled through the carotid and vertebral arteries in greater proportions than expected based on the flow volume for all patients. The distribution of larger emboli to these arteries varied greatly between patients, for some, the transport of emboli exceeded the flow proportion, for others the transport was less than the flow proportion, significantly so for several patients. Generally, particles were transported preferentially to the branches of the aorta rather than continuing to the wider descending aorta. The authors suggest that this could be due to the curvature of the aortic arch, pulsatile flow, and embolus inertia. The authors note that the one-way coupled model used may affect results, particularly for larger particles, but in general this choice of model should not change the tendency of emboli to enter the cerebral over the peripheral arteries.

### **2.2.4 The cerebral arteries**

Khodae et al. (2016) extended their work on carotid stenosis (Vahidi and Fatourae, 2012) and the carotid bifurcation (Abolfazli et al., 2014) to modelling the cerebral vasculature. Vessels were reconstructed from an MR angiogram to create a single patient-specific mesh. Emboli are again deformable and the ALE method was used with fluid-structure interactions. Emboli are released at different times during the cardiac cycle. Khodae et al. found that larger emboli tend to enter the MCA, and smaller emboli tend to enter the ACA. The rigidity of emboli was varied and affected the path taken by small emboli; with small, deformable emboli entering the ACA and small, rigid emboli taking the MCA. This is interesting since deformation of emboli is often ignored in favour of a

## *Section 2*

simplified model to reduce computational cost. Release time also affected the distribution of emboli between branches.

Fabbri et al. (2014) model the paths of emboli travelling through the cerebral vessels with a view to comparing embolus size, mass, and entry position (with respect to the vessel wall) with distribution among the terminating arteries of the model. Flow is simulated using the control volume finite element method. Vessels for a single subject were reconstructed from a time-of-flight MR angiogram using 3D Slicer<sup>1</sup> (Fedorov et al., 2012). Emboli had diameters of 0.2-0.8 mm and various densities; lower, equal to, and higher than, the density of blood. Emboli and blood were fully coupled and collision with the vessel wall was modelled. The authors assumed that flow resistance at the end of each terminating vessel would be similar, and so applied a pressure drop boundary condition, without lumped resistance, representing smaller vessels beyond the outlets. This contrasts with the common approach using resistance to ensure flow is proportional to vessel cross section, although the flows are comparable with clinical measurements. The proportion of emboli travelling through the MCAs was around 80% compared with 57% MCA blood flow, which demonstrates the MCA bias also presented in previous in vitro and TCD studies. Low density emboli were more likely to enter the MCA than high density emboli. Some emboli from the ICAs cross through the posterior communicating arteries, but no emboli from the basilar artery cross to the anterior circulation. Analysis of tortuosity of emboli paths showed that larger particles follow more tortuous paths and small particles take a nearly direct path through the vessels. Releasing emboli close to the wall compared with the centre of the vessel produced differing results, which shows that it is important

---

<sup>1</sup> <https://www.slicer.org/>

## *Section 2*

to consider a range of entry points. Modifying the circle of Willis by occluding the communicating arteries had a significant impact on emboli distribution.

Work on the cerebral vessels by Fabbri et al. (2014), extended to the carotid bifurcation by Mukherjee et al. (2016 (1)) was advanced to modelling the full vasculature leading from the aortic arch to the cerebral vessels by Mukherjee et al. (2016 (2)). CT angiograms were used from 4 patients, who all had a full circle of Willis and no fused vessels superior to the aortic arch. Outlet flows were taken from phase contrast MRI. Fluid simulation was similar to Mukherjee et al. (2016 (1)) and emboli were coupled one-way with the fluid. Emboli were spherical with diameters between 0.1 and 1 mm. Several thousand emboli were generated at the inlet of the aorta to represent cardiogenic emboli, and along the wall of the ascending aorta to represent aortogenic emboli. The effect of factors influencing the probability of emboli being transported to the circle of Willis varied widely between patients. Generally cardiogenic emboli were more dependent on release time (in the cardiac cycle) than aortogenic emboli, and cardiogenic emboli released during systole were more likely to pass to the circle of Willis than those released at diastole. Dependence of transport probability on embolus size varied greatly across patients, release locations and release times. The distribution of emboli across the major cerebral arteries branching from the circle of Willis was similarly patient dependent. The most probable destinations were the left and right MCAs. All patient RMCAs received cardiogenic emboli in greater proportion than expected based on the flow. The PCAs received more autogenic than cardiogenic emboli, the MCAs received more cardiogenic than autogenic, and the ACAs received similar numbers for both. The right brain hemisphere received emboli in greater proportions than the flow, and received a higher fraction of cardiogenic than aortogenic emboli. The probability of an embolus released at each location on the aortic wall travelling to the brain was mapped

## *Section 2*

(Figure 3) and showed that the majority of aortogenic emboli passing to the brain originated on the upper arch. Interestingly, aortogenic emboli released downstream of the cerebral-leading arteries branching from the aortic arch may still travel to the brain due to retrograde flow. The result that cardiogenic and aortogenic emboli are distributed differently among the cerebral arteries is important for further modelling of cerebral embolism. The distribution of emboli changes the pattern of lesions seen in patients, and in future may be correlated with embolus source through models such as this; due to the many factors involved, this would require further research to apply clinically. The authors suggest that their models may be used to improve surgical planning, and to improve stroke risk stratification in patients. The use of a one-way coupled model was justified by the authors' previous study showing that small to medium emboli behaved similarly in one and two-way coupled simulations. Since the previous study investigated the carotid bifurcation only, it is possible that the more complex anatomy and longer paths modelled here would result in larger deviation between one and two-way coupled emboli trajectories. Inertial effects have been shown to be important in small bodies in flow as well as large; Matas et al. (2004) show inertial migration for particles down to  $1/42$  of the vessel diameter for Reynolds number in the range encountered in the aorta. Given that the emboli in this paper ranged from around  $1/300$  to  $1/3$  of the vessel diameter between the aorta and circle of Willis, it is probable that these inertial effects will affect behaviour, and may not appear in a one-way coupled simulation. I suggest that it is therefore unreasonable to assume that the results of a two-way coupled simulation would be similar. The authors noted that they were working on two-way coupled simulation in parallel with this study so it will be interesting to see their results.

## Section 2

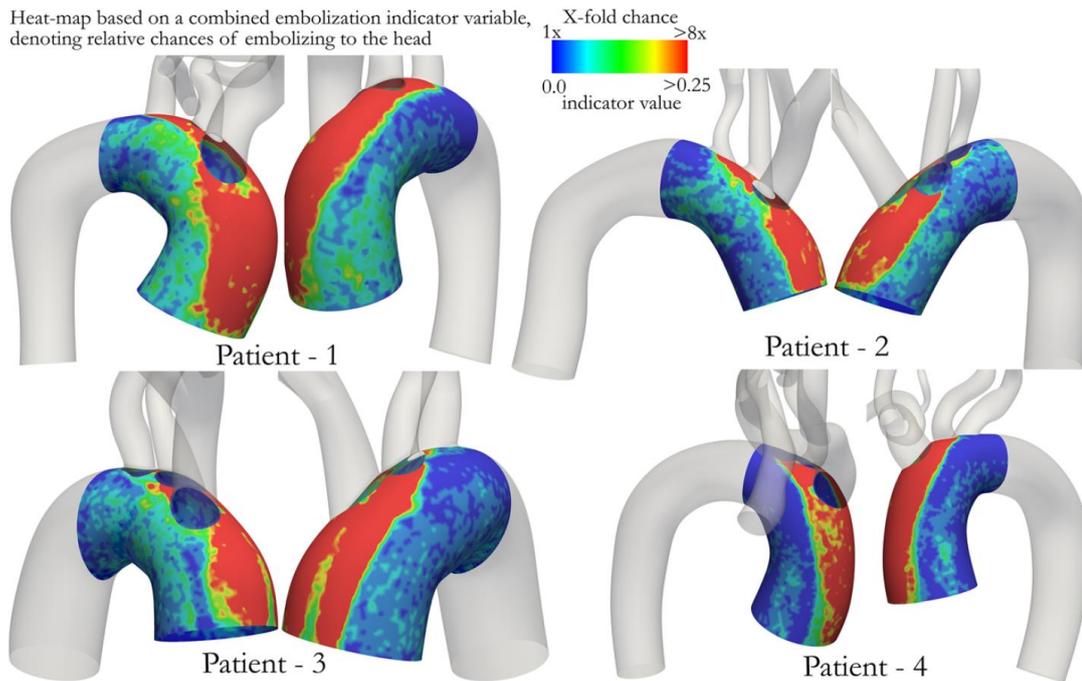


Figure 3. Map of probability of an embolus released from a location on the aortic wall travelling to the circle of Willis. (Mukherjee et al., 2016 (2))

### 2.2.5 Fractal tree models

Chung et al. (2007) present the first application of computational simulation to emboli transport. The authors note that applying the Navier-Stokes equation to simulate the cerebral arteries would be limited, since “with so much branching in the vasculature, the specific trajectories of emboli would become exceptionally sensitive to their initial placement in the root vessel.” Instead the authors use a minimal model with a deeply bifurcating fractal vasculature tree, ranging from the size of major cerebral arteries to capillaries; I will refer to the vessels in the last layer in the tree as capillary nodes. Large numbers of emboli are introduced into the system to explore the average statistical behaviour of the system using a Monte Carlo technique. The sensitivity of embolus behaviour on initial placement was apparent in the studies previously discussed, and was resolved by also applying Monte Carlo methods using large numbers of emboli to calculate statistical behaviour. This is now possible as a result of advances in computational power, algorithms and techniques; although 3D simulations are still significantly smaller in scope than the fractal

## *Section 2*

tree model. Emboli propagate through the tree, taking a random daughter branch at each bifurcation with a probability corresponding to the number of free capillary nodes downstream of each branch (as emboli are more likely to flow into open vessels). The embolus stops if it reaches a vessel smaller than itself, and occludes the flow in that vessel. Emboli then dissolve while stationary until they have reduced in size sufficiently to become dislodged and move further downstream, where they will occlude further vessels. This continues until the embolus has reduced to a size smaller than capillaries and can therefore pass through the rest of the vascular system unhindered. Both solid and gaseous emboli were modelled, with solid emboli dissolving at a much slower rate. Occluded vessels prevent flow in all downstream vessels (Figure 4). Emboli are introduced to the root of the tree at random times following a probability function. The authors found that once sufficient numbers of emboli are present in the tree, emboli will pass to flowing vessels such that the state of flow restriction quickly changes from benign to severe. This highlights the potential for showers of smaller emboli introduced in a short period to cause significant ischaemia. Gas emboli must be introduced in greater quantities than solid emboli to have any effect. This study also found that solid emboli result in longer periods of focal ischaemia than gaseous emboli, and that multiple small emboli of the same total volume are predicted to be less harmful than a single embolus. This model assumes that emboli will distribute across daughter branches in proportion to their flow, and that the flow is proportional to the number of supplied capillary nodes downstream.

## Section 2

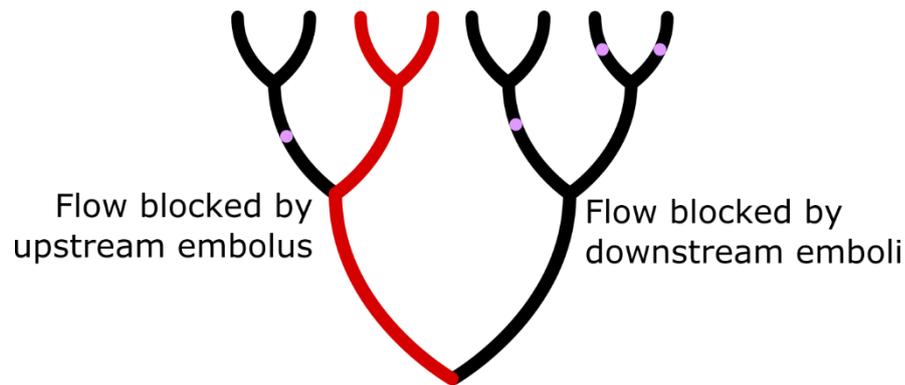


Figure 4. In the fractal tree embolism model, flow in a vessel segment ceases if either an upstream vessel is blocked, or if all downstream vessels are blocked.

This work was extended by Hague and Chung (2009). The authors analyse the assumption that flow at a bifurcation depends on the number of blocked capillary nodes downstream. The Hagen-Poiseuille equation, modified with a non-linear term to account for incompletely formed flow, is used to calculate flows downstream of a vessel based on the number of blocked capillary nodes. The results show that a linear approximation relating flow to free nodes sufficiently approximates the calculated values because capillaries have much larger resistance than other nodes in the tree due to their size. The dependence of trajectory on flow is no longer assumed to be linear, and is instead modelled using a sigmoid curve based on experimental data; which means an embolus is more likely to take the branch with greater flow by a probability greater than its flow proportion. Otherwise the model is the same, but with a more detailed analysis of the results. The authors determined analytically that the rapid change from flowing to heavily occluded flow (stroke) was consistent with a phase transition.

Further progress was made by Hague et al. (2013) in the modelling of gaseous emboli. Bubbles were allowed to stretch along a vessel and only became stuck when the friction on the wall matched the pressure force on the bubble. A buoyancy weighting was also added to each bifurcation such that bubbles prefer to take the higher path. The angle of vessels for this weighting was random, but the authors intend that this data could in future come from

## Section 2

imaging data or models. In this model pressures need to be calculated to determine whether or not a bubble becomes stuck, which required a new method to model the flow. Flow was modelled using the Hagen-Poiseuille equation together with an electrical circuit analogue approach used to estimate resistances for vessels in parallel and series. This is applied recursively up the tree and allows all flows and pressures to be solved in a reasonable time. Introduction of deformable bubbles significantly reduced the instantaneous number of blocked capillary nodes compared with previous results (Figure 5). The authors found that an increase in blood pressure reduced the time spent by bubbles in the vasculature. Replacing the bubbles with a more soluble gas, such as CO<sub>2</sub>, significantly reduced both washout time and total occlusion, which is relevant to the practice of CO<sub>2</sub> field flooding during cardiac surgery.

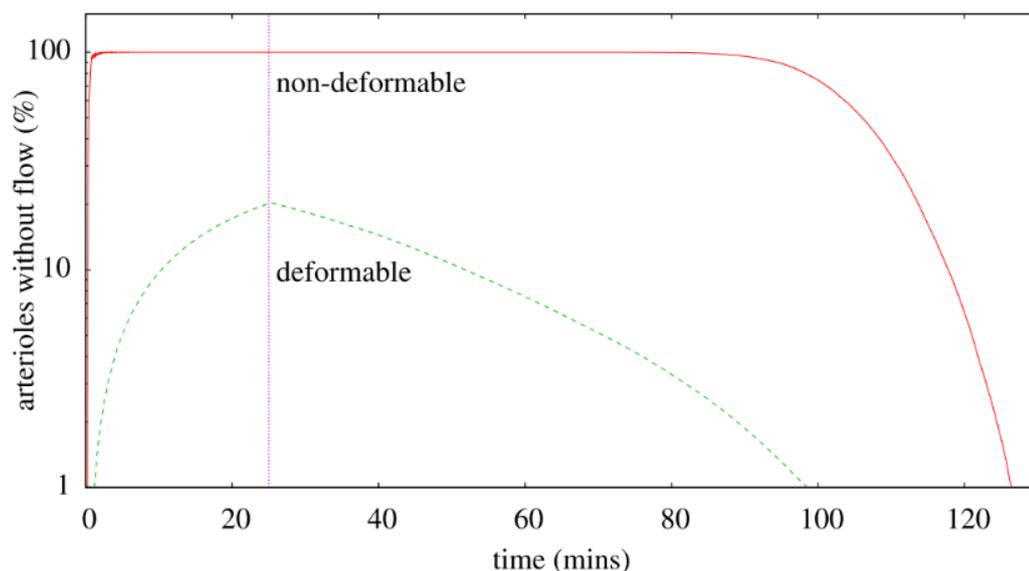


Figure 5. Percentage of blocked capillary nodes after a shower of gaseous emboli estimated for deformable and non-deformable model bubbles. Treating bubbles as deformable significantly reduces the number of blocked nodes. (Hague et al., 2013)

### 2.2.6 Literature discussion

While the literature is limited, I hope it is clear from the research described that significant progress has been made in the study of embolus transport. Research in this area benefits from advances made in the areas of computing and

## *Section 2*

computational fluid dynamics, so it is promising to see new techniques being utilised to improve the accuracy of simulations. There is potential in this field for the application of other fluid dynamics methods. So far, models have used finite volume, difference and element methods for 3D applications, but there are other techniques well suited to multiphase and objects in flow such as the lattice Boltzmann method and smoothed-particle hydrodynamics, which have yet to be fully explored.

It is good to see models increasing in scope, moving from isolated aortic arch, carotid and cerebral simulations to combined models. This advance has only been possible due to improving technology and the growing experience of individuals and groups in embolus simulation. From the studies presented, it is clear that results are extremely sensitive to initial conditions, so isolated models require careful consideration of boundary conditions if outputs are to be meaningful. Combined models should be less sensitive since the boundaries are further from the behaviour of interest, and less complicated embolic behaviours need to be assumed. For example, an isolated model of the carotid bifurcation can reveal the division of emboli at the bifurcation for emboli evenly distributed across the inlet at the common carotid, but what if flow upstream of the boundary means that emboli are not evenly distributed? If we also simulate the upstream portion of the flow, then our results are more likely to be clinically realistic.

The number of emboli in models has increased in later models, with some simulations using many tens of thousands of emboli. This is important when trying to reduce the sensitivity of output data to initial conditions; without sufficient sampling it can be easy to read too much into the motion of individual emboli and it becomes difficult to appreciate the broader behaviour. Without large samples, we are unable to resolve differences between patients

## *Section 2*

and better understand what facets of an individual anatomy influences the paths taken by emboli.

Common goals have been expressed by many authors cited in this review; these include gaining insight into embolic stroke, developing novel treatments, improving existing procedures, and patient-specific modelling for risk assessment and targeted treatment. From the perspective of better understanding embolic stroke, modelling can be used to provide additional information about embolus - blood flow interactions. With regards to patient-specific modelling, developing the detail and accuracy of models will improve trust in this approach, and hopefully lead to clinical adoption for identifying the source of embolisation and tailoring treatment.

The 3D flow and fractal tree models have both progressed sufficiently that these could be combined, leading to full models of embolus propagation from source to filtration in capillary beds. In the same way that expanding the scope of the 3D models to include both the aortic arch and cerebral vessels revealed nuances in embolic behaviour, I expect that combining 3D and fractal models will lead to further revelation. Due to the necessity for groups and individuals to build experience in order to create these complex models, I hope that this sort of full model could be achieved through collaboration and use of open tools. The work by Carr et al. (2013) and Mukherjee et al. (2016 (1)) on the open-source segmentation and simulation package SimVascular (Simvascular, 2016) is particularly promising as this makes replication and collaboration easier.

I would like to see further validation of models on simple problems. An example of this would be demonstration of the Segré -Silberberg effect (Segré and Silberberg, 1962) in a straight pipe. Neutrally buoyant spheres of a given size, in a tube of particular size and flow rate should migrate to a particular position relative to the tube wall. Modelling this behaviour demonstrates most of the physical behaviour necessary to properly model emboli, and since the

## *Section 2*

simulation is relatively simple, it is an ideal validation case. I believe that running this test would quickly highlight issues present in one-way coupled simulations. I understand that one-way coupling is appealing when first trying to generate data since the computational demands can be orders of magnitude lower, but unless known physical behaviours of emboli in the flow are replicated there is limited merit to the results.

The fractal tree model will require anatomical data in some form to progress, since there is a limit to how much we can learn from generic branching trees, particularly for bubbles whose trajectories are strongly weighted by branching angle. There is also the question of how such a model can be validated. Due to the small scales involved, the results of microvascular simulations are difficult to validate using laboratory models. Replication of clinical data such as infarct distributions in stroke patients would of course give credence to the model, but otherwise, it would be beneficial if some method of validation could be developed, even at a significantly reduced scale.

A major difficulty facing adoption of embolus modelling is the following: How do we get meaningful information from chaotic behaviour? On average a particular embolus in a particular patient may travel with higher probability down a particular branch; but in the case of a single embolus, or even a small shower of emboli, we are unable to accurately predict where any particular embolus would travel. Ideally we would like to be able to image the locations of ischaemic lesions, and incorporate patient specific vascular anatomy to our models to confirm the sizes and locations of cerebral emboli in the vascular tree, and trace these back to confirm the source of embolisation. This would aid diagnosis and treatment of patients with multiple sources of cardiovascular disease, Atrial Fibrillation, or a Patent Foramen Ovale (PFO-Hole in the heart) by enabling clinicians to determine the source and composition of emboli and take appropriate preventative action. The case of large showers of emboli is

## *Section 2*

simpler, as once statistical behaviour dominates, our models gain merit. Using patient-specific models to relate anatomy to data such as infarct locations would be worthwhile as a means to better understand stroke, however, the barriers to applying this sort of modelling clinically have yet to be addressed.

### **2.2.7 Literature conclusion**

I have presented a comprehensive review of literature studying the motion of emboli in the bloodstream. The clinical motivation for modelling this behaviour is to better understand embolic stroke, and improve treatment. 3D fluid simulations of solid embolus transport have progressed from isolated models of the carotid bifurcation and other sections of the vasculature, to large models encompassing the aortic arch and cerebral vessels. Tens of thousands of emboli are now introduced to these simulations to produce statistical data highlighting the extreme sensitivity of embolus trajectory to initial conditions, and dependence on patient anatomy. Fractal tree simulations have been used to model the transport of solid and gaseous emboli through millions of vessels in a deeply branching tree. These simulations have highlighted the potential for showers of emboli to have significant impact on perfusion where numbers and volume of emboli rise above a critical threshold. However, the potential for influencing patient treatment is unclear, and there are significant limitations which must be addressed and overcome before patient specific modelling comes into clinical use.

In summary, I believe that simulation of emboli holds great promise for making sense of complex problems in understanding the aetiology of stroke but will require collaborative end-to-end models covering all scales of the problem.

### **2.3 Adaptation of a fractal model and embolus sizing algorithm to patient specific intra-operative TCD monitoring**

This part of my PhD involved combining a fractal model of air bubbles moving through the arteries (Hague et al. 2013) with a novel bubble sizing algorithm (Banahan et al., 2012) to perform patient-specific simulations based on emboli detected intra-operatively during cardiac surgery.

In previous work, Hague et al., 2009 had created a model of air embolus propagation through a fractal vascular tree (described in section 2.2.5) to study how large quantities of small air bubbles introduced into the blood might lead to obstruction of arteries and impaired perfusion. In Hague et al., 2013, this model was applied to the cerebral vasculature.

Parallel work by Banahan et al., 2012 described and validated a novel method of sizing air emboli based on analysis of transcranial Doppler (TCD) ultrasound recordings obtained from the middle cerebral arteries (MCAs).

By combining these techniques it becomes possible to model the effects of emboli introduced to the cerebral circulation on cerebral perfusion to determine whether the size, number, and timing of bubbles received is likely to result in brain perfusion injury. This study was conducted as part of the BICI (Brain Injury during Cardiac Surgery) BHF study.

To apply the fractal tree Monte-Carlo embolisation model to cardiac surgery required two patient-specific inputs:

1. Embolus timings and sizes during the surgery.
2. The diameter of the patient's middle cerebral artery (MCA).

First the simulation was adapted to enable the introduction of emboli to the root node of the vascular tree according to timings given in a data file. Embolus

## *Section 2*

timings and sizes were provided to the occlusion simulation by analysing TCD recordings for embolic signals and further analysing the properties of these signals using the embolus sizing algorithm to estimate bubble diameter.

Bubbles entered to the model were assumed to pass through the tree as described previously (section 2.2.6) At each time-step the proportion of blocked vasculature was estimated and displayed to enable researchers to explore the impact of bubbles detected during the surgery.

As TCD monitoring of emboli is used to detect bubbles entering the MCA it was important to match the root size of the tree to each patient's MCA diameter extracted from their time-of-flight MR angiogram. Accurate estimation of patient-specific MCA diameter is also important for accurate bubble sizing, since the backscatter intensity of emboli relative to blood depends on the diameter of the vessel. In this study, both the occlusion and embolus sizing models used MCA diameters estimated for individual patients from the same angiogram. Given both models' reliance on MCA diameter, it is important to ensure that this is measured for each patient using a robust and accurate method.

Embolus sizing is heavily influenced by the size of the MCA (Figure 2.1). Efforts should therefore be made to minimise the MCA diameter error, since the maximum bubble error is roughly 3x the MCA error. To give an example: For a MCA diameter error of  $\pm 0.3$  mm, a bubble which is 1 mm has an error of 25.6%. The actual bubble size will therefore be between 0.75 mm and 1.25 mm.

## Section 2

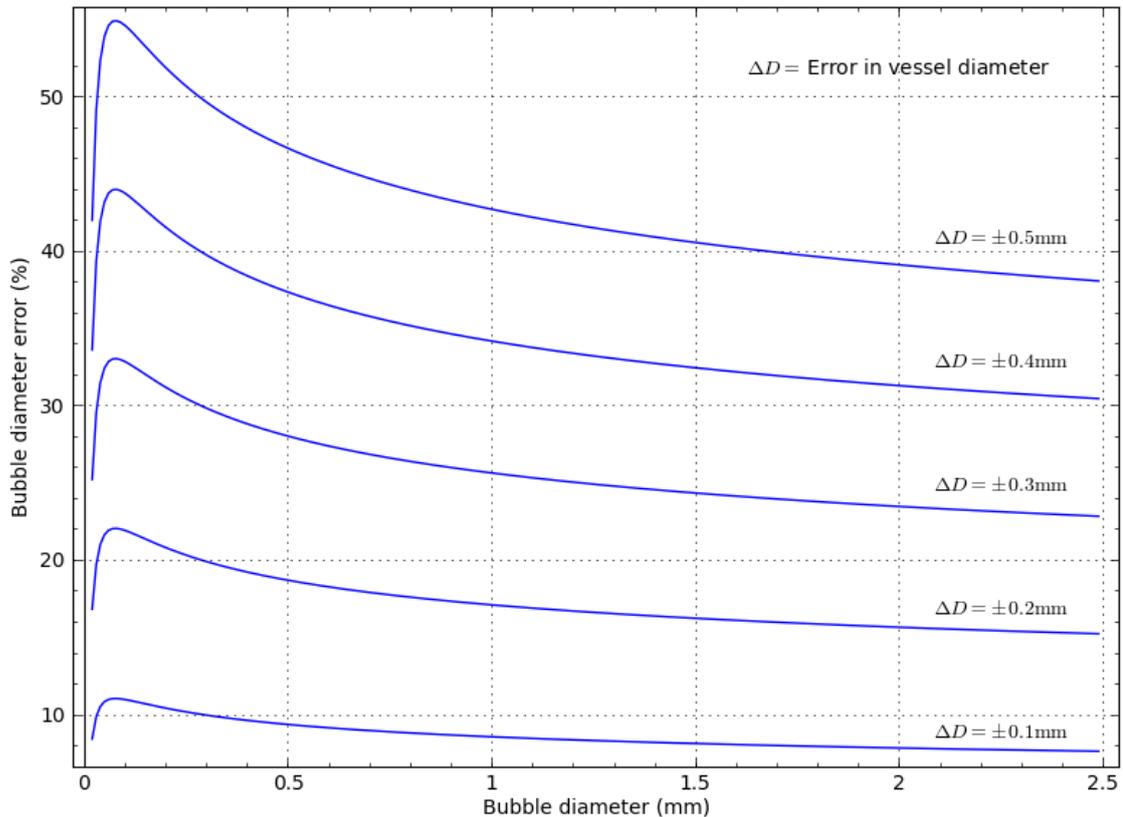


Figure 2.1 Curves calculated to demonstrate uncertainties in bubble sizing due to errors in measuring the MCA. Each line represents a particular MCA diameter measurement error. The bubble error increases linearly with MCA error. Data were generated for the case that haematocrit = reference = 0.45, SVL = reference = 10 mm, R = reference = 2.5 mm (roughly average for MCA).

The aim of this study was to overcome some of the practical limitations encountered when translating basic science models of embolisation to potential clinical applications. At present there are very few studies applying flow or embolus modelling methods to large patient cohorts and translation of flow models to suitable clinical applications has been slow.

In the work described in this, and the following, section I consider how to apply the models described above to a large number of patients, to help address barriers and practical challenges. Results of applying the models to a set of 48 patients are provided.

## **2.4 Methods**

### **2.4.1 Introduction**

My contribution to the BICI study was to run bulk simulations from patient data. Previously this would need to be done manually, which took a long time and was not practical for large numbers of patients. There was a need to automate this for all patients.

I created a job submission script and adapted the code to run on the ALICE high performance computing cluster at the University of Leicester.

The outputs of the embolus sizing model were used as inputs to the embolus propagation model to produce patient-specific simulations. The sized emboli data files were added to a prescribed directory structure, along with data on patient MCA diameters. The submission script used this information to run many patient simulations in parallel. The simulation results were output to a standard location so that results for all patients could be processed into graphs using another set of scripts.

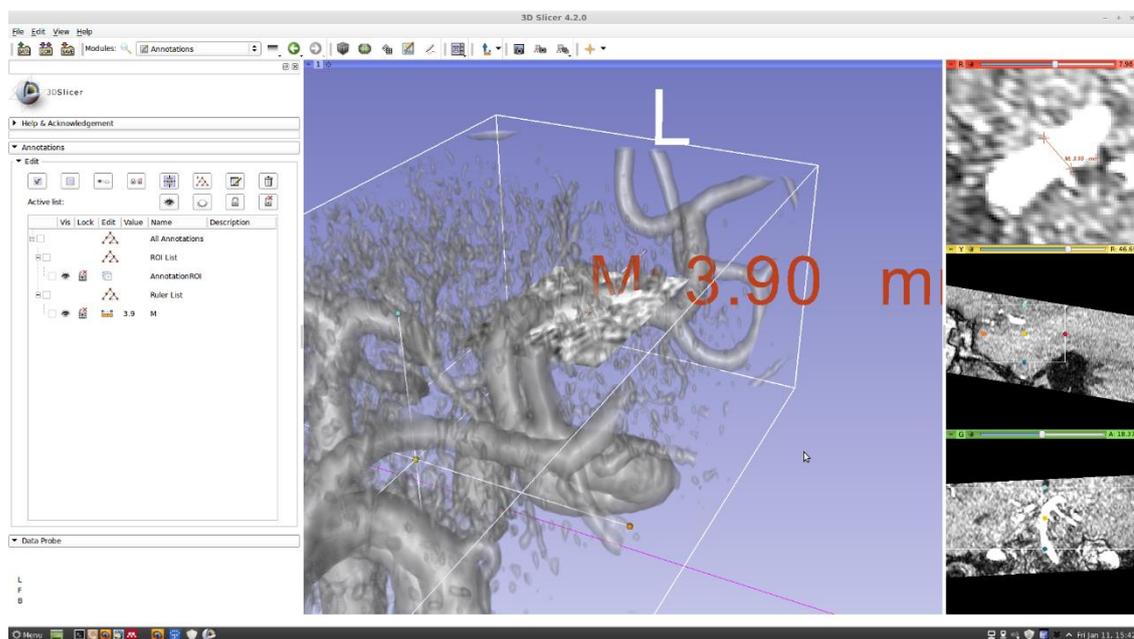
### **2.4.2 Extraction of MCA diameters from patient MRA**

MCA diameters were measured using quantitative analysis of time of flight magnetic resonance angiography images obtained prior to surgery. Two methods were tested for the extraction of MCA diameters from imaging data; 3D Slicer and the Vascular Modelling Toolkit (VMTK). The aim of this analysis was to estimate the diameter of the main M1 segment of the MCA used to detect emboli during TCD monitoring. Ideally, the software used for MCA diameter estimation should provide robust objective measurements to avoid generating errors in bubble sizing or the results of our occlusion simulations.

## Section 2

### 3D Slicer

In 3D Slicer, measurements were made from a single location soon after the ICA->MCA,ACA bifurcation. As the MCA did not appear to vary appreciably in radius, and considering the large errors involved with this approach, a single measurement was obtained. The image resolution is 0.26 mm. Considering difficulties involved in selecting a cut-off intensity for distinguishing the MCA from neighbouring vessels, tissue, and noise; the estimated error in estimating MCA diameter using 3D slicer was  $\pm 0.5$  mm.



### VMTK

The Vascular Modelling Toolkit (Antiga et al., 2008) was used to segment images using level sets at a constant threshold for all patients. Mean diameter was calculated by averaging 5 cross-sections separated by distances equal to the vessel radius, starting from a position located two vessel radii from the ICA bifurcation. Sections on poorly resolved areas were discarded. Diameter errors were estimated based on standard deviation of area measurements. See Figure 2.2.

The full MCA diameter measurement process is described in section 1.7 of this thesis.

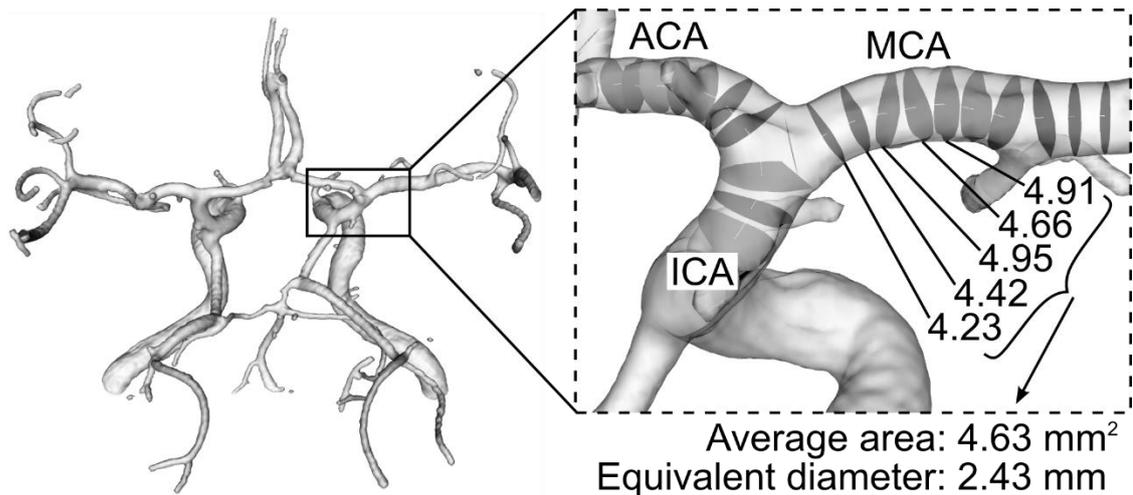


Figure 2.2 Extraction of MCA diameter using VMTK. Cross sections are calculated through the vessel and areas are averaged to estimate vessel diameter.

### 2.4.3 Producing patient-specific embolisation simulations

Embolus data from TCD recordings, patient-specific MCA measurements and the occlusion simulation were combined to estimate how the percentage of occluded vasculature varied during cardiac surgery procedures.

Scripts were used to submit batch simulation jobs for all patients to the computing cluster. A general template script was modified based on specific patient data such as MCA radius and location of sized embolus input data.

The output from each job is a text file containing, among other information, the instantaneous number of blocked end arterioles at each time during and after surgery. Each job took up to 2 days to complete, running on a single CPU core. Since the simulations were run in parallel, all patient simulations could be run in 2 days rather than around 100 days if run on a single PC.

Full scripts are given in Appendix B.

Example results from the simulations are given in the results section.

## 2.5 Results

### 2.5.1 Measurement of MCA diameters for 10 patients using 3D Slicer

The MCA diameter measurements are plotted in Figure 2.3. The sample size is small so it is not possible to deduce any general trends, but the mean and median for the RMCA and LMCA are 2.8 mm with standard deviation of 0.2 mm. The range of values was 2.6-3.3 mm. Four patients had a larger left than right MCA, and 6 patients had a larger right MCA. The left and right diameters were weakly correlated ( $r = 0.41, p = 0.25$ ).

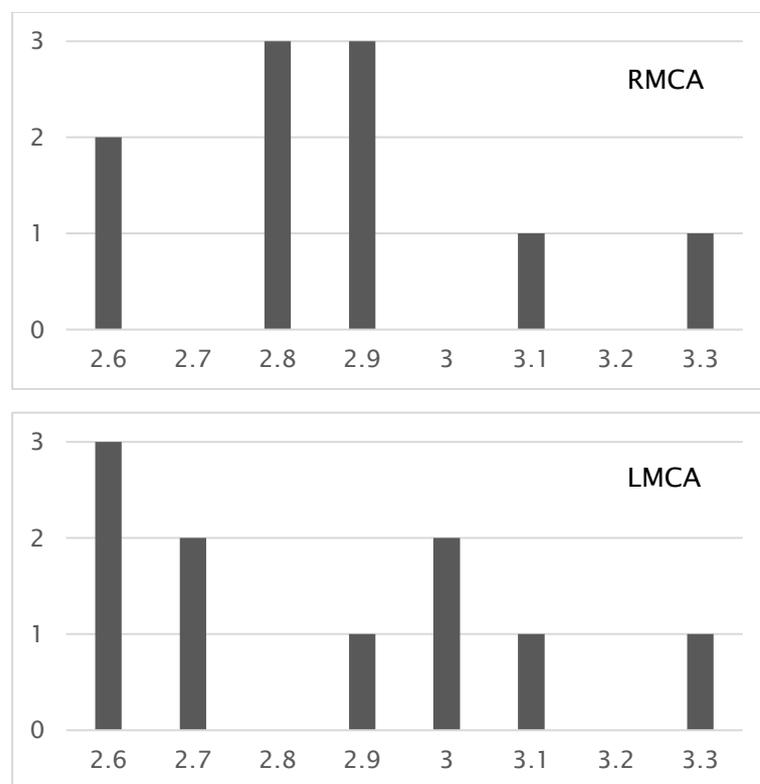


Figure 2.3 Distribution of patient MCA diameters measured using 3D Slicer.

### 2.5.2 Measurement of MCA diameters for 48 patients using VMTK

The MCA diameters of 48 patients estimated using my VMTK script are plotted in Figure 2.4. Mean and standard deviations were  $2.69 \pm 0.26$  mm for the RMCA and  $2.66 \pm 0.25$  mm for the LMCA. With the larger sample, we can now see that left and right diameters are related. Left and right diameters are correlated with

## Section 2

$r = 0.74$ ,  $p < 0.00001$ . 27 patients had a larger RMCA than LMCA, and 21 patients had a larger LMCA.

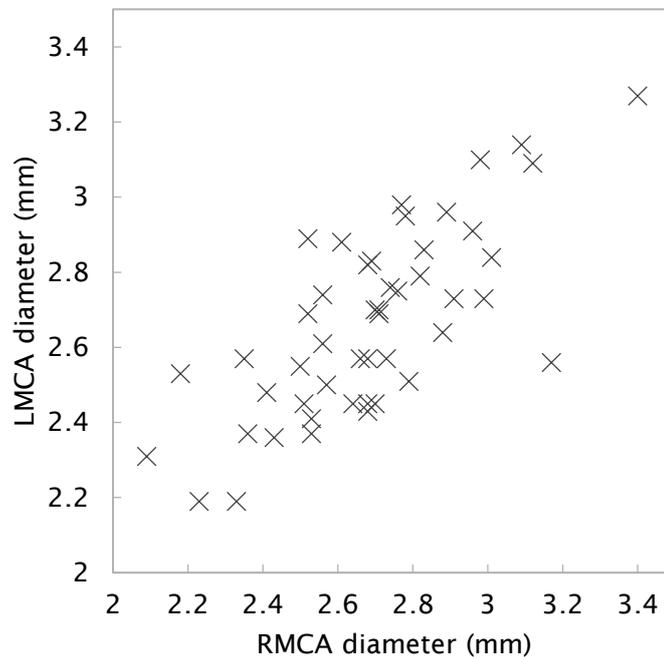


Figure 2.4 MCA diameters measured using the VMTK script.

The 10 patients in the previous section were a subset of the 48 patients in this section, so we can directly compare the results of MCA estimates for these 10 patients obtained with 3D slicer against results obtained using the VMTK script, Figure 2.5. On average, measurements with 3D slicer were larger than those obtained using the VMTK script; only two measurements were smaller. This could be due to several factors: The chosen threshold for measurement will vary more between patients in 3D slicer than VMTK, and was adjusted so that the vessel could be clearly seen against the background noise. This likely results in the vessel appearing larger in 3D slicer than using the less subjective VMTK analysis. In the 3D Slicer method, two orthogonal measurements were taken across the artery at a single location, whereas the VMTK script based the diameter estimate on cross-sections taken along the length of the vessel. Usually mid-way along the vessel was chosen in 3D slicer for measurement,

## Section 2

but the average diameter of the overall artery may be higher or lower than at the mid-point.

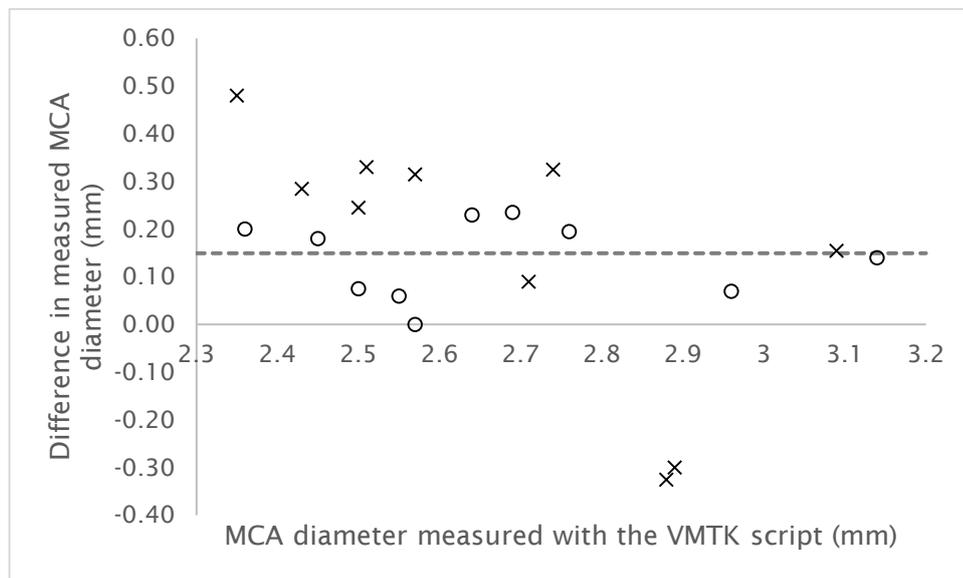


Figure 2.5 Plot of differences between MCA diameters measured in 3D Slicer and VMTK. Circles and crosses indicate LMCA and RMCA measurements respectively. The dotted line indicates the mean measurement difference of 0.15 mm. Almost all measurements in 3D Slicer are larger than measurements using the VMTK script.

### 2.5.3 Example embolus propagation simulation results

Results of sizing emboli in the MCA of a patient during cardiac surgery are given in Figure 2.6. Plots of the sized emboli provide insight in to the large number of emboli introduced during surgery as well as highlighting parts of procedures which lead to large embolus numbers. This information on its own does not fully describe flow occlusion which may occur due to these air bubbles.

## Section 2

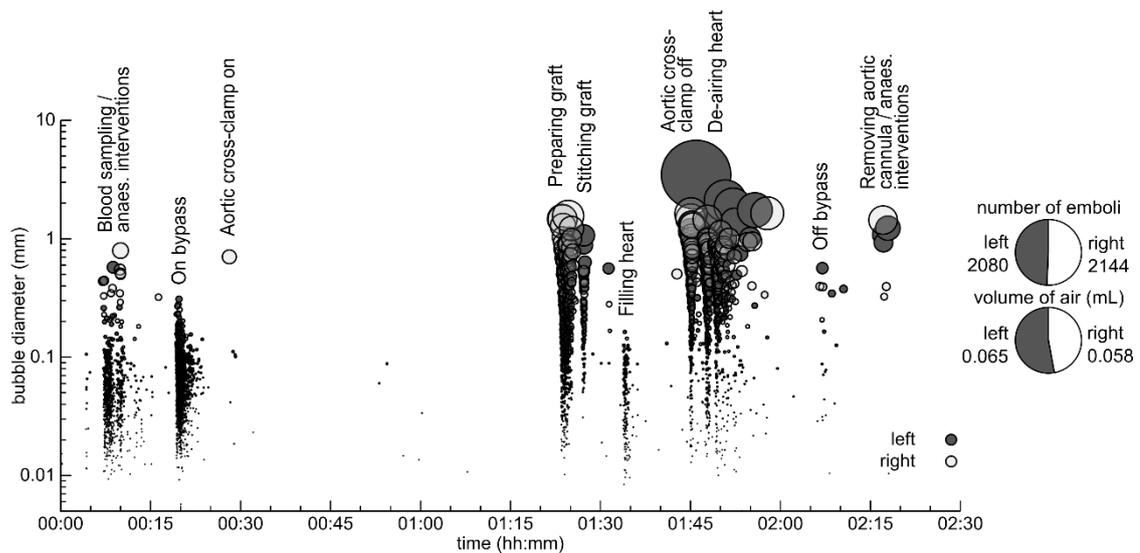


Figure 2.6 Sized emboli data results for a patient recorded during cardiac surgery. Notable points in the surgery are labelled on the figure. Some parts of the procedure lead to a large number of emboli being introduced into the cerebral vasculature such as removing the aortic cross-clamp and de-airing the heart. Data recorded as part of the BICI study and published in Chung et al. (2015).

Simulating embolus propagation from sized embolus data gives data such as that shown in Figure 2.7. The simulation models theoretical net occlusion of end arterioles in the vasculature beyond the left and right MCAs. From this we can assess whether the emboli are likely to have caused significant occlusion leading to harmful lack of tissue perfusion. The purpose of the study was to compare this information, as well as other factors, to patient outcome to attempt to identify factors leading to reduced cognitive function in patients. Full results of the study are published in Chung et al. (2015).

## Section 2

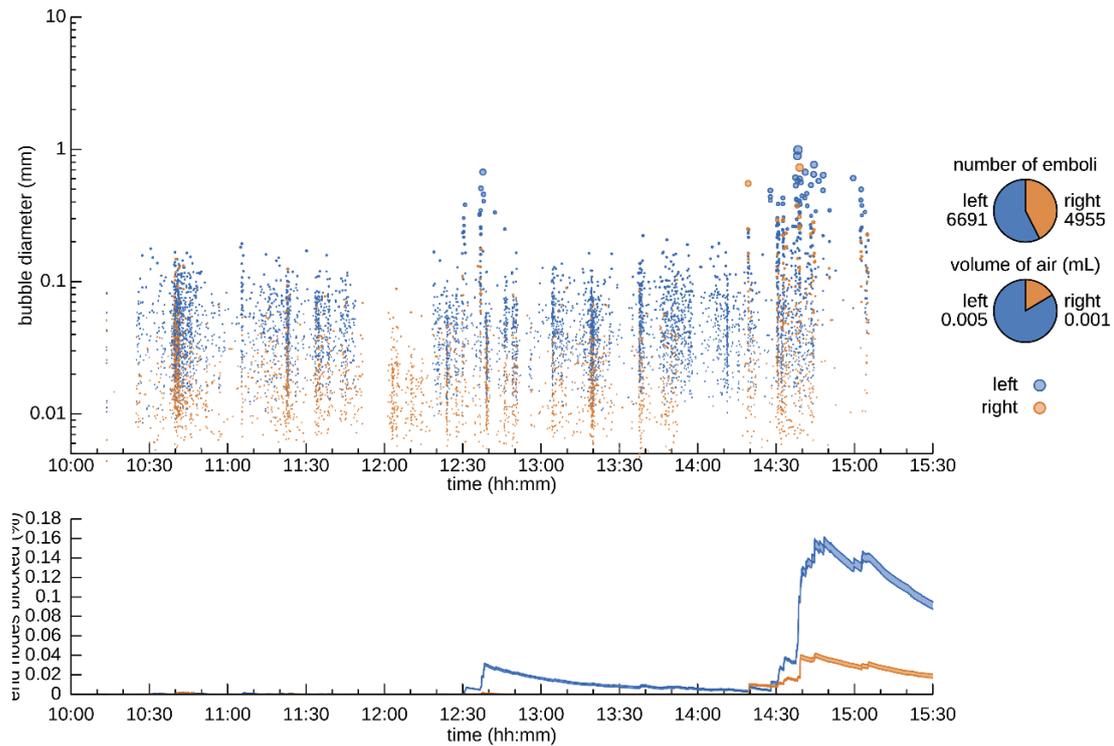


Figure 2.7 Modelled percentage of blocked “end nodes” due to measured emboli introduced into the brain of a patient during cardiac surgery. Although a large number of emboli are introduced to both sides of the brain during the procedure, most occlusion occurs due to the large emboli introduced towards the end of the procedure.

The number of vessels blocked for more than an hour is plotted in Figure 2.8. We can see that most patients have less than 1% of end-arterioles blocked for this time, with three patients having 1-1.5% and a single patient having over 4.5% occlusion for this time. From comparing simulation results to plots of occlusion, we found that large occlusion percentages were caused by large air emboli rather than many smaller emboli.

Overall, the study did not find a relationship between number of emboli or modelled occlusion and negative patient outcomes.

Section 2

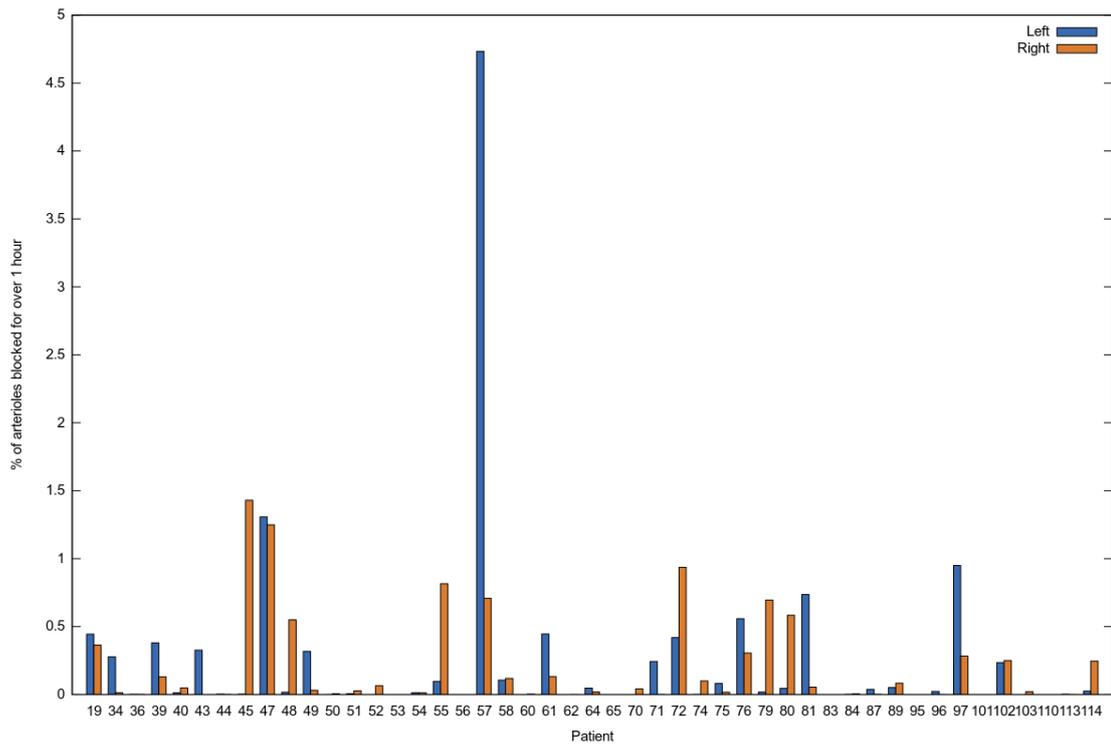


Figure 2.8 Fraction of end-vessels beyond the MCA which were occluded for longer than an hour in all patients included in the simulation study. Most patients have less

## **2.6 Discussion**

In this section I have discussed models of emboli in the vasculature and applied the MCA measurement methods developed in section 1 to embolus models applied to patient data.

There are many methods of modelling emboli, which have been discussed, but few have been applied to the clinical setting. The modelling I describe here was a successful clinical application, and is the first model to achieve this for cardiac surgery.

The results of these simulations give an estimate of the occlusive effect of large numbers of emboli introduced to the brain during cardiac surgery. The results of the larger BICI study, of which this work is a part, suggest that the modelled occlusion does not statistically result in negative outcomes for patients.

To summarise the published literature from this study by Chung et al. (2015) and Patel et al. (2015): The bubbles are small, and they're too small to be clinically significant. The volume of air was tiny. Models showed that very small sections of the vasculature would be clinically obstructed. Together with the simulation data we found no correlation between patient cognitive decline or brain lesions seen in MRI with number of bubbles, volume of air, or modelled occlusion.

The lack of correlation could be due to either limitations of the techniques, or because cognitive decline and brain lesions resulting from cardiac surgery are due to other factors.

The embolus monitoring method has known accuracy dependence on factors such as TCD angle of incidence and the value of MCA diameter. Care was taken with MCA measurement and the TCD angle is unlikely to vary drastically, so we can expect reasonable results. The Doppler signal may be dropped during surgery, but if this happens it is apparent and recorded.

## *Section 2*

The strength of the embolus transport model is that number or sizes of emboli alone are not enough to predict occlusion, this sort of fractal tree model is necessary for getting this information. Unfortunately the model has some weaknesses. In the model, bubbles cannot split; bubbles stretch along each vessel, resulting in larger frictional force on the vessel walls. In reality, bubbles are likely to stretch only so far and then split at bifurcations. This could result in large bubbles quickly becoming smaller and perhaps being flushed from the system faster.

Because the study did not produce significant results, it cannot be used as a predictor of cognitive decline. The hope would be that an accurate model could predict cognitive decline and be applied to clinical practise. As it stands, the model is still useful as a research tool.

Participating in this study has shown me that making real impact on clinical practise is difficult. Whether the difficulties stem from limitations of technology or methods, or from the complex nature of the problem; finding ways to apply basic science clinically involves overcoming many hurdles.

It is somewhat frustrating that despite the increasingly detailed information we are able to capture in a patient-specific manner, we are unable to find the key to “unlocking” the information for clinical use. In the case of the BICI study we have broad and detailed patient data on neurocognitive capability, details of the surgery, ultrasound monitoring and sizing of thousands of emboli per patient, and computational methods to simulate emboli. Even with all this information, the result is still that we can't say what causes neurocognitive decline during cardiac surgery.

I hope that as techniques improve, our understanding of how to apply new modelling methods improves, and technology allows faster capture of the information needed for this type of broad study. The BICI study took multiple

## *Section 2*

researches several years to produce results, which puts a limit on the information that can be captured. Improving the quality, quantity and understanding of this information requires either more time, money or better technology. My hope is that in future better technology will bridge the gap between data and clinically applicable results in this application.

# Section 3. Application of SPH to modelling of emboli in blood flow

## 3.1 Introduction

As outlined in previous sections, understanding the behaviour of emboli in blood flow is important for developing methods for avoiding the damaging effects of emboli and understanding the distribution and spatial severity of embolic lesions in patients at risk of stroke. Section 2 showed that embolus monitoring can be used during surgical procedures that introduce solid and gaseous emboli into the blood, such as cardiac surgery (Barak and Katz, 2005; Nietlispach et al., 2010). In this section I explore the potential of using a simulation method call Smoothed Particle Hydrodynamics, to model the motion of emboli in blood flow.

The motion of emboli in blood flow is complicated by various forces which result in a “ring” of particles distributed at a preferred distance from the vessel axis. This phenomena is known as the Segré-Silberberg effect (Segré and Silberberg, 1962), and is demonstrated in Figure 3.1. This phenomena occurs due to a balance between a “wall effect” force which pushes particles to the centre (due to the particle’s wake interacting with the vessel wall) and a ‘lift’ induced by the fluid velocity gradient across the vessel (Di Carlo et al., 2007) that causes migration of emboli away from the centre. The resulting distribution of emboli is usually centred on  $\sim 0.6$  times the vessel radius from its centre. This arrangement has been observed clinically in the middle cerebral artery of patients undergoing carotid endarterectomy (Chung et al., 2006).

Various laboratory and *in silico* models of emboli in the flow have been developed, ranging from CFD models (Vahidi and Fatouraee, 2012), laboratory

### *Section 3*

phantoms (Chang et al., 1981; Chung et al., 2010; Bushi et al., 2005), experiments in excised vessels (Suzuki and Eckmann, 2003), and other mathematical models (Chung et al., 2007; Hague and Chung, 2009; Branger and Eckmann, 1999; Hague et al., 2013). Solid and gaseous emboli (bubbles) require individual models due to the different physics involved. Bubbles mostly move along the highest path at a bifurcation due to buoyancy, even if the lower branch is receiving a higher proportion of the flow (Chang et al., 1981). Solid emboli tend to follow the flow, although the relationship is non-linear (Chung et al., 2010).

The movement of bodies in flow has been studied for modelling the microvasculature, where solid interactions are used to treat red blood cells as separate bodies, which is important for understanding vascular haemodynamics (Fung, 1997). Another field which concerns itself with these interactions is lab-on-a-chip devices. These require the precise manipulation of biological matter in microfluidic channels (Di Carlo, 2009); this has led to several developments in modelling bodies in flow at a microscopic level. Advances in understanding fluid-solid interactions from these fields can be applied to the macroscopic vasculature. For flow in capillaries the dynamics are heavily dependent on the behaviour of the blood's constituents as separate, solid bodies. Models treating blood in this way are now possible due to recent advances in modelling fluid-structure interactions (Zhang et al., 2012; Li et al., 2012).

CFD models of emboli require accurate simulation of interactions between the embolus and the fluid. In the case of solid emboli, the surface is usually rigid and the pressure and shear forces across the surface can be evaluated to determine the resultant force and torque. For gas bubbles, the surface is deformable so models must be more complex. The geometry used depends on the problem in question. A single bifurcation may be used to understand how

### Section 3

an embolus behaves under generic flow conditions, which can then be applied to many bifurcations in a larger model without the computational cost of modelling the full geometry in 3D. The full anatomy may also be used, for example, in modelling embolus trajectory past the circle of Willis.

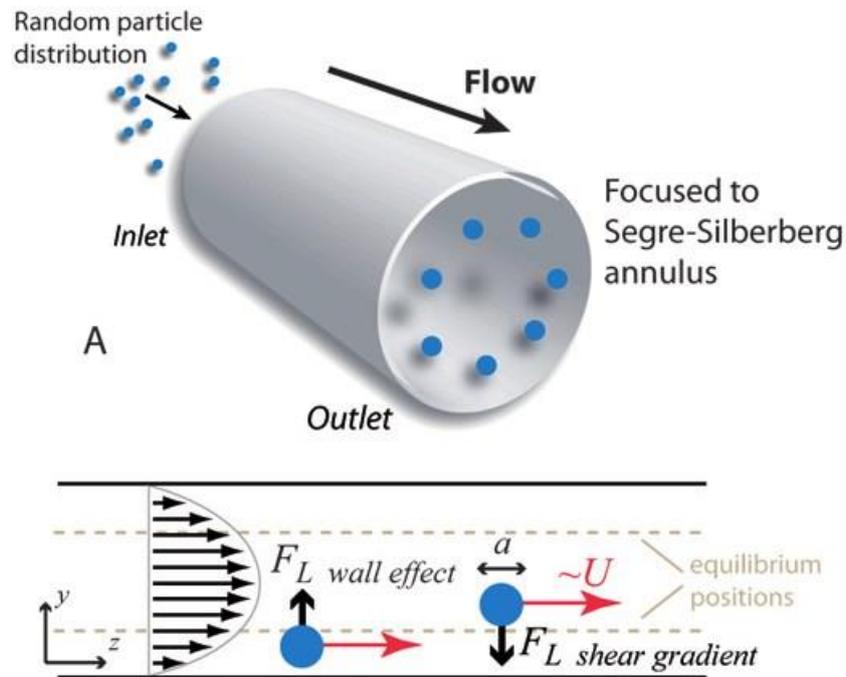


Figure 3.1 Example of the Segré-Silberberg effect. Particles migrate to an equilibrium “ring” located a preferred distance from the vessel wall. Adapted from Di Carlo (2009).

Phantoms and excised vessel models, have used both synthetic and real emboli introduced to the flow. Emboli are tracked using a high speed camera to determine the paths they take through the vessels and how they distribute themselves through the vascular network. Laboratory models can test full and complex anatomy without the computational cost of a simulation and may be more accurate. These models cannot, however, fully replicate the behaviour of the blood and vessel walls *in vivo* and generally a blood-mimic and vessels that are stiffer than real arteries are used. The inherent accuracy of a laboratory replica is therefore often offset by unavoidable differences in physiology. Modifying phantoms is difficult, so new phantoms must be created for each anatomy, which is expensive and time consuming. Some models allow minor

### *Section 3*

modifications, such as attachment of “aneurysms” at certain points (Fahrig et al., 1999), but generally these models do not approximate anatomy well.

Computer models utilising the fractal properties of arterial trees (Zamir, 1999) have recently been developed (Hague and Chung, 2009; Hague et al., 2013). The vascular system of these models consists of large arteries originating from the MCA or major cerebral arteries, which successively bifurcate until reaching the layer of arterioles above the capillary mesh. The capillaries anastomose to form venules and then veins which reconnect to the heart. From the larger arteries down to capillaries, the diameters range from  $\sim 2$  cm at the aorta to  $\sim 10$   $\mu\text{m}$  at the capillary bed. By modelling this system as a large set of connected nodes, emboli can be introduced into the system and propagated using equations governing their movement at each node to determine at what level an embolus will block a vessel. This shows the effects of a number of solid or gaseous emboli entering the system on global and local blood flow, which may be used as a risk indicator for patient specific models. For procedures such as cardiac surgery, information from techniques such as TCD may be used as model inputs, which could potentially highlight patients at high risk of brain injury.

## 3.2 Introduction to SPH

Smoothed-particle hydrodynamics (SPH) was originally developed for astrophysics (Monaghan, 1992) but has since been applied to a wide range of fluid and solid mechanics problems (Monaghan, 2012). It is a mesh free Lagrangian particle method for solving the equations of fluid dynamics. This means that the 3D mesh describing the vessel lumen required for a conventional fluid simulation is not required. Particles are used to represent the fluid and the properties of a region of the fluid are interpolated from the properties of the particles in that region using relevant equations. SPH is a Lagrangian method, and therefore considers the fluid as a number of packets which travel through space with time, as opposed to Eulerian methods, which treat the fluid as an array of absolute locations that vary in their properties as the fluid flows. Unlike SPH, most conventional CFD techniques are Eulerian. The advantages of SPH for fluid applications come from the use of “virtual” particles to represent the flow. This avoids several difficulties that are associated with conventional CFD. Conventional CFD tends to use fixed boundaries and solve relevant equations according to those boundaries. This requires the mesh used as the boundary to have to certain qualities, which take a great deal of time to optimise for accurate results. SPH does not require a fixed boundary mesh and instead uses particles to represent boundaries. Treatment of boundaries is still under development, but current SPH methods work reasonably well for a range of bounded applications and do not require mesh optimisation. This flexible treatment of boundaries allows easy inclusion of elastic vessel walls, which is necessary for accurate models of the cardiovascular system.

Traditionally, to calculate Lagrangian streamlines in Eulerian models, massless tracer particles are introduced into the flow and their motions tracked according to the fluid properties. This appears to work well but adds additional

### *Section 3*

complexity and computational cost to simulations. In SPH, tracking of particles 'comes for free'. Virtual particles follow streamlines; this is useful for adding interaction physics that involves tracking local flow volumes, such as those needed for modelling blood clotting.

Unfortunately, SPH simulations tend to be slower than Eulerian methods for non-astrophysics problems. This is due in part to the relative infancy of the technique, however, development has been rapid and speeds are now becoming more competitive, particularly due to the rise of massively parallel processors such as those used in modern graphics cards.

Our interest in SPH stems from the ease with which complicated geometry and fluid-solid interactions can be introduced. Applications of SPH to blood flow are few and recent, but the results are promising, and with further development, it is hoped this technique will allow insightful simulations to be created.

#### **3.2.1 SPH applied to blood flow**

To date, SPH has seen limited, but promising, applications to blood flow. A number of different applications have been found, which can be roughly divided into four areas: 1) Flow in vessels or the heart, for problems such as plaque formation, or modelling the effectiveness of heart valves. 2) The micro-scale mechanics of red blood cells and other blood components. 3) Virtual surgery simulations for training purposes. 4) Simulations for improvement of healthcare technology. The first area is most relevant to the research of our group and has displayed some interesting preliminary results which demonstrate the feasibility of using SPH for modelling blood flow.

In an early conference paper, Sinnott et al. (2006) demonstrated an SPH simulation through a 3D carotid bifurcation geometry constructed from MRI data. Steady and pulsatile flow were modelled, with velocities and pressures at

### *Section 3*

the bifurcation analysed in each case. Rigid walls were used, with boundaries constructed using SPH particles (although the method for this was not described). This study was important as it proved the feasibility of adapting SPH for simulating blood flow. The authors found a difference in flow division at the bifurcation between steady and pulsatile flow models.

Flexible walls were introduced by Farahani et al. (2008), who tested pulsatile flow in 2D, with wall boundaries constructed using SPH particles and “ghost” particles for treating the no-slip boundary condition. The results were minimal, but again demonstrated the feasibility of using SPH to simulate blood flow.

2D simulations of flow through a blood vessel were used by Liang et al. (2012) to develop SPH methods for curved boundaries. Simple Poiseuille flow was also tested and agreed well with the analytical solution. Flow was continuous and walls were rigid.

Shahriari (2011) used SPH to simulate several biomechanical problems in 2D. Simple oscillating flows were simulated for validation purposes (Shahriari et al., 2013), with different SPH implementations tested against analytical solutions to determine which would provide the greatest accuracy in more complex simulations. It was found that SPH works well for Poiseuille flow and oscillatory flow.

The model was then extended to simulate a 2D left ventricle of the heart (Shahriari et al., 2012a). The simplified test case of a ‘lid-driven’ cavity flow was compared against analytical solutions, and a pulsatile jet in a cavity was compared against finite volume results. Both benchmarks agreed well with known solutions. The 2D ventricle was tested with rigid boundaries and reproduced a pulsatile jet during the filling phase, followed by a continuing vortex, which is consistent with physiological measurements. Shahriari et al. (2012a) introduced a method for determining inlet/outlet boundaries, which

### Section 3

drives particles in fixed boundary regions at a controlled velocity, before removing them at the edge of the boundary. This ensures that particles always have a full number of neighbours and allows precise control of velocities at the boundary.

Bileaflet mechanical heart valves were also simulated (Shahriari et al., 2012b), see Figure 3.2, with an emphasis on studying the shear stress on blood components, which can lead to haemolysis or thrombosis. SPH is well suited to this sort of analysis since it is a Lagrangian method, so local fluid properties are easily evaluated. Pulsatile flow was simulated with rigid vessel walls and valves. A normal and dysfunctional valve were considered. The valves do not move during the simulation, so the results were not sufficient to reach any meaningful conclusions regarding valve function; however, the model was compared to other computational and *in vitro* models which use the same geometry, and the SPH model was confirmed to produce similar results.

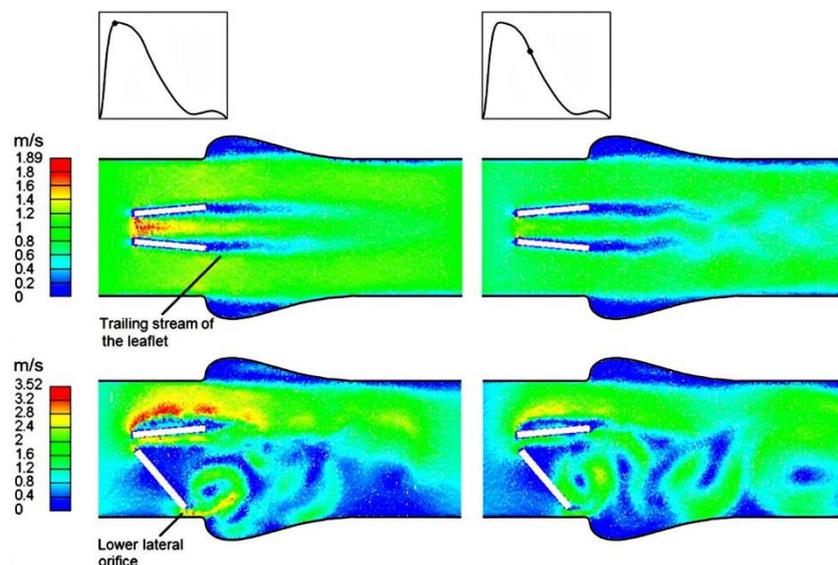


Figure 3.2 2D SPH simulation by Shahriari et al. (2012b) showing the flow velocity profile through a functional (top) and non-functional (bottom) model heart valve, at two stages of pulsatile flow.

A different approach by Gholami et al. (2014) studied atherosclerosis by modelling leukocytes and red blood cells separately in the flow, assuming a

### *Section 3*

hypothesis that endothelial dysfunction increases production of adhesion molecules which bind leukocytes to the cell wall, leading to plaque formation. The leukocytes were modelled as passive tracers as they do not constitute a large enough proportion of the blood to affect the flow dynamics. They are generated close to the vessel walls to avoid excessive computational cost. A simple backwards-facing step case was considered with constant flow and rigid walls. The preliminary results were promising and the author hopes to extend the model to complex patient-specific geometries and improved modelling of leukocyte interactions.

Red blood cell models using SPH have been attempted, and have produced results that agree well with experiment. While these models do not directly apply to our work, the treatment of cell membranes is helpful for developing models of vessel walls. A model by Tanaka and Takano (2005) (Cited by Hosseini and Feng (2009)) used a combination of standard SPH particles and elastic “membrane” particles to create the cell. The results reproduced the expected motion of cells, as well as axial migration and pinching, which demonstrates the ease with which fluid structure interactions can be included in SPH simulations. Further work by Hosseini and Feng (2009) improves the membrane model to avoid ‘leakage’ of internal SPH particles and produced results which agreed well with physiological measurements. Extending the model to include other organisms, Hosseini and Feng (2012) studied the effect of malaria parasites on red blood cells. This required complex handling of the fluid-membrane interaction and produced reasonable results.

Several studies have integrated SPH into virtual surgery software, which is designed for training surgeons and therefore requires full modelling of tissue and fluid properties. These developments stem predominantly from “video game” style SPH, where accuracy is less important than achieving a desired impression in real-time interactive simulation. The methods are therefore less

### *Section 3*

applicable to our work, but are included here to ensure a comprehensive overview of SPH use for biological flows. Preliminary work by Müller et al. (2004) used SPH to simulate bleeding of a severed artery. SPH is ideal for this sort of simulation, as treating the deformable vessel, and creation of a free surface as the blood spreads, is relatively simple. We note that the number of particles used in this simulation is 3000, which is low compared to the  $10^5$ – $10^7$  commonly used in precise simulations. Liu et al. (2008) added irrigation, as well as blood flow, to a cranial base simulator. Kerwin et al. (2009) used SPH to simulate the filling of cavities with blood during surgery, this improves the appearance of the simulation and suction can be practised. The simulation by Chui and Heng (2010) incorporates both fluid and solid SPH models to simulate the blood and vessel walls. Around 6000 particles are used for real-time, interactive frame rates. The model also incorporates a clotting model to test the effects of surgical interventions. This final work takes the most physically realistic approach to surgical simulation and may be adapted for future research.

An interesting application by Cenova et al. (2011) uses SPH to improve the resolution of magnetic resonance angiography (MRA) flow data. An SPH implementation with constrained average velocities, determined from the MRI data, is introduced. The model is tested using a simple tube flow phantom. The results are promising and appear to add information to the imaging data, but it is unclear how well this method will scale to images of complex patient anatomy.

A heart pump model by Sinnott and Cleary (2010) was used to optimise pump designs. There are a number of publications which use SPH for industrial applications with various fluids, but this paper is useful as the effect of shear forces on the blood was considered. The model produced good results for a complex case. Another partially related model by Sinnott et al. (2012) applies

### *Section 3*

SPH to fluid transport in the colon. Although this is not a blood flow simulation, much of the physics is similar; the contracting colon wall presents a difficult problem and their handling of the fluid interaction at the boundary is of relevance to our work. It is promising to see SPH applied to problems that would conventionally use grid-based approaches, because of the benefits of using a Lagrangian method.

In summary, the progress of SPH in blood flow modelling is modest, but the results so far show great promise and increasing applications are being found where conventional CFD models struggle.

#### **3.2.2 SPH applied to bodies in pipe flow**

The first and to our knowledge only application of SPH to the study of bodies in pipe flow is by Pazouki & Negrut (2015). The authors modelled sparse and densely packed solid spheres suspended in flow through circular and square-channel pipes (Figure 3.3). Flows over a wide range of Reynold's numbers (0.1 to 1400) are considered. Their simulations showed good agreement with theory when modelling Poiseuille flow, tubular "spheres" in a 2D channel (between two parallel plates), and in a 3D pipe. In particular, they were able to reproduce the tubular "pinch" effect described by Segré & Silberberg (1962) where bodies in the flow migrate to a preferred distance from the tube centreline (Figure 3.1, top). They finally run a large simulation of up to 23,000 solid ellipsoids in a square channel, demonstrating the applicability of SPH to computationally expensive and complicated problems.

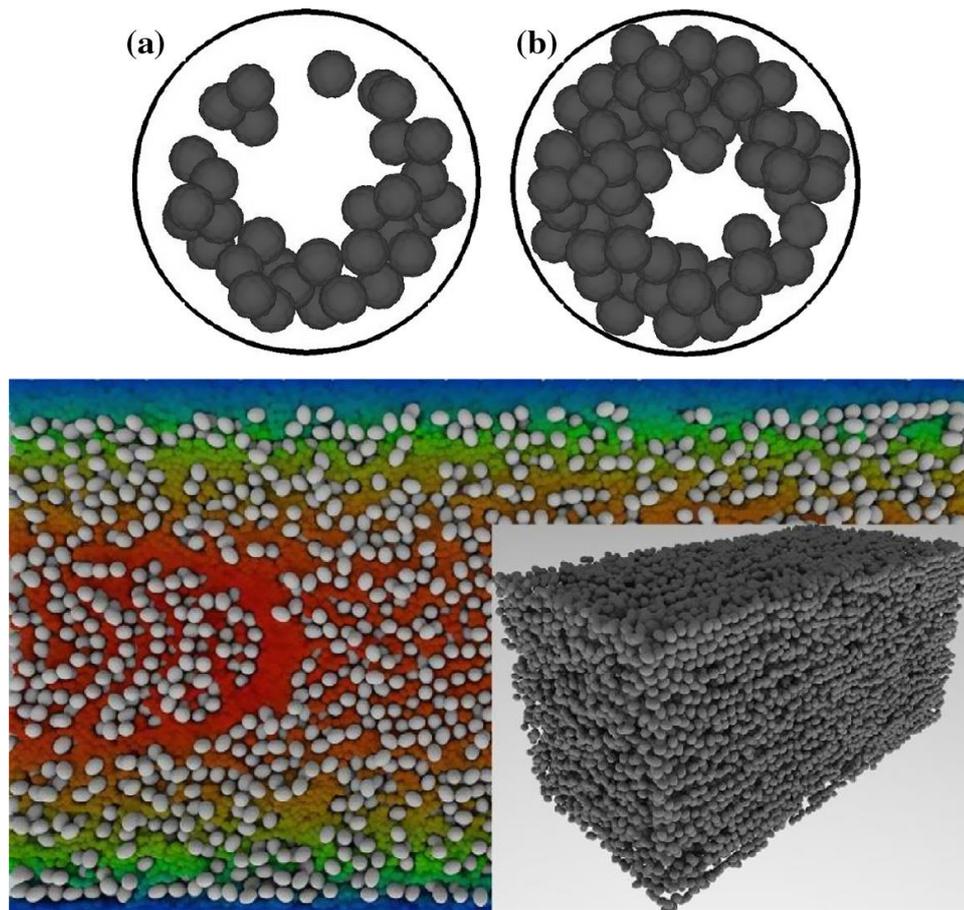


Figure 3.3 SPH simulation of the flow of solid spheres suspended in a circular (top) and square-channel (bottom) by Pazouki & Negrut (2015). The spheres and fluid repeat over a periodic boundary at the end of each tube. In the top image, varying concentrations of spheres are tested, ranging from sparse (a) to dense (b). Sparse distributions demonstrate a sharper Segré-Silberberg “pinch” effect than dense distributions (Segré & Silberberg, 1962).

### 3.2.3 Using SPH to simulate emboli in blood flow

The limited number of studies on computational modelling of embolus motion in the blood described in the previous section all used conventional finite volume, difference and element methods. The models have been shown to be very sensitive to initial conditions, and often require a variety of techniques to add emboli to the standard flow calculation methods.

Studies applying SPH to blood flow have produced promising results in agreement with results from more conventional techniques. SPH is being used

### *Section 3*

in cases where the constituents of the flow must be modified to model the problem, such as leukocyte transport.

The work of Pazouki & Negrut (2015) shows that SPH can accurately simulate spheres on the scale of emboli (sphere/pipe radius ratio  $\sim 0.1$ ) in flow within a Reynolds number range suited to blood flow (they test Reynolds numbers in the range of 0.1 to 1400; cerebral blood flow is around  $Re=500$ ).

SPH has characteristics which will be helpful for modelling emboli. The Lagrangian nature of the technique lends itself to modelling discontinuous parts of the flow, such as solid bodies. The combination of the limitations of conventional techniques, and success in applying SPH to blood flow and bodies in pipe flow problems, support the potential benefits of modelling emboli with SPH.

In this section I propose use of SPH for modelling emboli in the bloodstream and perform an exploratory 'proof of concept' study to develop SPH code specifically for embolus modelling.

### 3.3 SPH methodology

#### 3.3.1 SPH methodology theory

The essential equations of SPH can be derived using Lagrangian mechanics and a density estimate. We use this method here as it clearly links the underlying physics to the simulation. Additional physics can be introduced by modifying the Lagrangian, which is used to derive the simulation equations. The method here is adapted from Price (2012), Monaghan (2005) and Violeau (2012), which should be referred to for more comprehensive derivations.

SPH is a particle description of a fluid and, as such, the particles must contain all the information needed to represent the fluid. (i.e. velocity, pressure and density). Velocity is stored for each particle in SPH and evolves due to the density and pressure in the surrounding fluid. Pressure depends only on density in water-like fluids under normal conditions. Density is therefore the root property that changes all other properties with time. This is true for a non-dissipative system; entropy and other properties are usually added as additional terms during the derivation, if required.

#### Derivation of equations of motion

The key question for a particle method is then, how do we get the density? The density is defined at points located at the particles, but a single particle cannot give the density at that location. We interpolate the density at a particle from the masses and locations of the surrounding particles using a weighting function. Since all particles have equal mass, the density depends entirely on relative particle positions. This function reduces the contribution of neighbour particles to the density with increasing distance. The equation which computes the density  $\rho$  about a point  $\mathbf{r}$  is

$$\rho(\mathbf{r}) = \sum_b m_b W(\mathbf{r} - \mathbf{r}_b, h) \quad (3.1)$$

### Section 3

where the sum is over all neighbour particles,  $m$  is mass, which is constant for all particles,  $W$  is the weighting function which depends on distance and the “smoothing length”  $h$ .  $W$  is known as the “smoothing kernel”, for which smoothed-particle hydrodynamics is named, and has units of  $m^{-3}$ . To conserve total mass,

$$\int \rho \, dV = \sum_b m_b \quad (3.2)$$

to ensure this is satisfied we impose the normalisation condition

$$\int_V W(\mathbf{r}' - \mathbf{r}_b, h) dV' = 1. \quad (3.3)$$

The smoothing kernel must satisfy several conditions: The normalisation condition above, it must be symmetrical, decrease monotonically with distance and preferably the central peak should be flat and wide so near neighbour weightings vary little with position. A Gaussian kernel fulfils these requirements,

$$W(\mathbf{r} - \mathbf{r}', h) = \alpha e^{-q^2} \quad (3.4)$$

where  $\alpha$  is a normalisation factor which is a function of  $h$  and the number of simulation dimensions  $n$ ,

$$\alpha(n, h) = \frac{1}{(n\sqrt{\pi})h^n} \quad (3.5)$$

and  $q$  is the distance as a factor  $h$ ,

$$q(\mathbf{r} - \mathbf{r}', h) = \frac{\mathbf{r} - \mathbf{r}'}{h}. \quad (3.6)$$

In simulations, it is computationally expensive to calculate the Gaussian due to the exponent and non-zero value at large distances. Simpler approximate representations are used instead, which reduce to zero beyond a certain

### Section 3

distance. This avoids calculating interactions with particles whose contribution will be negligible. A commonly used kernel is the cubic B-spline,

$$W = \alpha \begin{cases} 1 - \frac{3}{2}q^2 + \frac{3}{4}q^3 & 0 \leq q \leq 1 \\ \frac{1}{4}(2 - q)^3 & 1 \leq q \leq 2 \\ 0 & 2 \leq q \end{cases} \quad (3.7)$$

Where  $\alpha = 2/(3h)$ ,  $10/(7\pi h^2)$ , and  $1/(\pi h^3)$  in 1D, 2D, and 3D respectively. Past  $\mathbf{r} - \mathbf{r}' = 2h$  the kernel is zero and therefore any particles outside of this radius are excluded from the computation, greatly speeding it up compared with a Gaussian kernel. The gradient is then

$$\nabla W = \alpha \begin{cases} -3q + \frac{9}{4}q^2 & 0 \leq q \leq 1 \\ -\frac{3}{4}(2 - q)^2 & 1 \leq q \leq 2 \\ 0 & 2 \leq q \end{cases} \quad (3.8)$$

For an incompressible fluid, the smoothing length  $h$  does not need to vary as the density is restricted to small perturbations from the mean. It is computationally cheaper to use a fixed smoothing length, which is normally around a factor of 1.25 larger than the initial particle separation. Particles are usually arrayed on a grid for 2D simulations and in a face-centred cubic lattice in 3D simulations for optimum packing density.

To progress from the density estimate to the equations of motion we start with the Lagrangian  $L$ , which describes the total energy of the system and is expressed as

$$L = T - V \quad (3.9)$$

where  $T$  and  $V$  are respectively the kinetic and potential energies. For our particle system this becomes

Section 3

$$L = \sum_b m_b \left[ \frac{1}{2} v_b^2 - u_b(\rho_b, s_b) \right] \quad (3.10)$$

where  $v_b$  are the particle velocities,  $u_b$  the specific internal energies (energy per unit mass) and  $s_b$  entropy.

We use the least action principle to get the equations of motion, using the Euler-Lagrange equation:

$$\frac{d}{dt} \left( \frac{\partial L}{\partial \mathbf{v}_a} \right) - \frac{\partial L}{\partial \mathbf{r}_a} = 0 \quad (3.11)$$

where we use Eq. (3.10) to get

$$\frac{\partial L}{\partial \mathbf{v}_a} = m_a \mathbf{v}_a \quad (3.12)$$

$$\frac{\partial L}{\partial \mathbf{r}_a} = - \sum_b m_b \left. \frac{\partial u_b}{\partial \rho_b} \right|_s \frac{\partial \rho_b}{\partial \mathbf{r}_a}. \quad (3.13)$$

Since we are assuming a dissipation-free case the derivative can be taken at constant entropy. This derivative is taken from the 1st law of thermodynamics at constant entropy to get

$$\left. \frac{\partial u_b}{\partial \rho_b} \right|_s = \frac{P}{\rho^2}. \quad (3.14)$$

The density gradient for constant  $h$  is (from Eq. (1.7))

$$\frac{\partial \rho_b}{\partial \mathbf{r}_a} = \sum_c m_c \frac{\partial W_{bc}(h_b)}{\partial \mathbf{r}_a} (\delta_{ba} - \delta_{ca}) \quad (3.15)$$

where  $W_{bc} \equiv W(\mathbf{r}_b - \mathbf{r}_c, h_b)$  and  $\delta_{ab}$  is a Kronecker delta function (valued 1 at  $a = b$  only and 0 elsewhere).

We substitute Eq. (3.14) and (3.15) into (3.13) to get

Section 3

$$\frac{\partial L}{\partial \mathbf{r}_a} = - \sum_b m_b \frac{P_b}{\rho_b^2} \sum_c m_c \frac{\partial W_{bc}(h_a)}{\partial \mathbf{r}_a} (\delta_{ba} - \delta_{ca}). \quad (3.16)$$

Putting this and Eq. (3.12) into (3.11), we have the equations of motion

$$\frac{\partial \mathbf{v}_a}{\partial t} = - \sum_b m_b \left( \frac{P_a}{\rho_a^2} + \frac{P_b}{\rho_b^2} \right) \nabla_a W_{ab} \quad (3.17)$$

which are used to calculate the acceleration on each particle according to the surrounding pressures and densities.

At this stage we must choose an equation of state to relate the pressure and density. There are many options in the literature, but we highlight two common “stiff” equations of state for incompressible fluids here. The first (Monaghan, 1994) is often used in free surface flows for problems similar to dam-break or coastal simulation,

$$P = \frac{c_0^2 \rho_0}{\gamma} \left[ \left( \frac{\rho}{\rho_0} \right)^\gamma - 1 \right] \quad (3.18)$$

where  $c_0$  is the speed of sound,  $\rho_0$  the reference fluid density and  $\gamma \sim 7$  is the fluid polytropic constant. The second (Morris et al., 1997) is suitable for internal flows where fluid fills the whole space and is therefore more appropriate for our research:

$$P = c_0^2 \rho. \quad (3.19)$$

The sound speed  $c_0$  in SPH must be much lower than the physical value as the computational time step becomes short, leading to long simulation times.

Usually a sound speed is chosen which is a factor of 10 larger than the bulk motion in the fluid (Monaghan and Kos, 1999).

To introduce viscosity, commonly an “artificial” viscosity term (Monaghan, 1992) is added to Eq. (3.17) at this stage:

### Section 3

$$\frac{d\mathbf{v}_a}{dt} = - \sum_b m_b \left( \frac{P_b}{\rho_b^2} + \frac{P_a}{\rho_a^2} + \Pi_{ab} \right) \nabla_a W_{ab} + \mathbf{g} \quad (3.20)$$

where

$$\Pi_{ab} = \begin{cases} \frac{-\alpha \overline{c_{ab}} \mu_{ab}}{\overline{\rho_{ab}}} & \mathbf{v}_{ab} \cdot \mathbf{r}_{ab} < 0 \\ 0 & \text{otherwise} \end{cases} \quad (3.21)$$

where averages are represented as  $\overline{A_{ab}} = 0.5(A_a + A_b)$ , vector differences as  $\mathbf{A}_{ab} = \mathbf{A}_a - \mathbf{A}_b$ , and  $\alpha = 1$ , although this can be varied depending on the problem, also

$$c_a = \sqrt{\frac{\partial P_a}{\partial \rho_a}} = c_0 \left( \frac{\rho_a}{\rho_0} \right)^{(\gamma-1)/2} \quad (3.22)$$

$$\mu_{ab} = \frac{h \mathbf{v}_{ab} \cdot \mathbf{r}_{ab}}{r_{ab}^2 + \eta^2} \quad (3.23)$$

where  $\eta^2 = 0.01h^2$ . Although useful, this viscosity approximation is too dissipative and affects shear forces in the fluid, which can be corrected by using a different formulation (Dalrymple and Rogers, 2006).

Finally we need to evolve the density, for which we use the continuity equation, which is simply the time derivative of Eq. (1.7),

$$\frac{d\rho_a}{dt} = \sum_b m_b \mathbf{v}_{ab} \cdot \nabla_a W_{ab}. \quad (3.24)$$

### Time stepping

We can express the full equations of motion needed to evolve the system as

$$\frac{d\mathbf{v}_a}{dt} = \mathbf{F}_a \quad (3.25)$$

$$\frac{d\rho_a}{dt} = D_a \quad (3.26)$$

Section 3

$$\frac{d\mathbf{r}_a}{dt} = \mathbf{v}_a \quad (3.27)$$

where  $\mathbf{F}$  is force per unit mass and  $\mathbf{D}$  is the density rate of change. These equations describe the change of all relevant particle properties through time. The properties of every particle are evolved by time stepping, a process which iterates the particle motion at a sufficiently slow rate that errors are minimised and the simulation approximates the physical system. The size of the time step  $\Delta t$  is controlled by the Courant-Freidrichs-Lewy (CFL) condition (Courant et al., 1928) (As cited in Morris et al. (1997).), with additional viscous controls implemented in Monaghan and Kos (1999):

$$\Delta t = 0.3 \min(\Delta t_f, \Delta t_{cv}) \quad (3.28)$$

where  $\Delta t_f$  is the step based on force per unit mass

$$\Delta t_f = \min_a(\sqrt{h/|\mathbf{F}_a|}) \quad (3.29)$$

and  $\Delta t_{cv}$  is the step based on CFL and viscous controls

$$\Delta t_{cv} = \min_a \left( \frac{h}{c_a + \max_b \left| \frac{h(\mathbf{v}_{ab} \cdot \mathbf{r}_{ab})}{r_{ab}^2 + \eta^2} \right|} \right) \quad (3.30)$$

The properties can be time-stepped using a number of methods, among which the Verlet integrator is popular (Monaghan, 2005). For the purposes of simplicity, our work currently uses a predictor-corrector algorithm, which is still popular in modern simulations (Akinci et al., 2012; Monaghan and Kos, 1999; Gomez-Gesteira et al., 2012b).

The algorithm begins by incrementing particle properties by half a time step according to the starting properties, then calculating the pressure,

### Section 3

$$\mathbf{v}_a^{n+1/2} = \mathbf{v}_a^n + \frac{\Delta t}{2} \mathbf{F}_a^n \quad (3.31)$$

$$\mathbf{r}_a^{n+1/2} = \mathbf{r}_a^n + \frac{\Delta t}{2} \mathbf{v}_a^n \quad (3.32)$$

$$\rho_a^{n+1/2} = \rho_a^n + \frac{\Delta t}{2} D_a^n \quad (3.33)$$

$$P_a^{n+1/2} = f(\rho_a^{n+1/2}) \quad (3.34)$$

at half-step the iteration is refined by recalculating using the new half-step properties,

$$\mathbf{v}_a^{n+1/2} = \mathbf{v}_a^n + \frac{\Delta t}{2} \mathbf{F}_a^{n+1/2} \quad (3.35)$$

$$\mathbf{r}_a^{n+1/2} = \mathbf{r}_a^n + \frac{\Delta t}{2} \mathbf{v}_a^{n+1/2} \quad (3.36)$$

$$\rho_a^{n+1/2} = \rho_a^n + \frac{\Delta t}{2} D_a^{n+1/2} \quad (3.37)$$

and at end of the step the final properties are calculated using

$$\mathbf{v}_a^{n+1} = 2\mathbf{v}_a^{n+1/2} - \mathbf{v}_a^n \quad (3.38)$$

$$\mathbf{r}_a^{n+1} = 2\mathbf{r}_a^{n+1/2} - \mathbf{r}_a^n \quad (3.39)$$

$$\rho_a^{n+1} = 2\rho_a^{n+1/2} - \rho_a^n \quad (3.40)$$

$$P_a^{n+1} = f(\rho_a^{n+1}). \quad (3.41)$$

This covers all of the theory required to create an SPH simulation. We have by no means covered all of the methods which have been developed to date, and much of the theory here requires additions specific to the simulation problem. This description of the theory has been kept intentionally short to give an overview of how, from a few simple principles, all of the basic equations of SPH can be derived. It is this simplicity and the intuitive nature of a particle

### *Section 3*

approach to fluid modelling that makes adding physics simpler than with conventional CFD methods.

#### **3.3.2 Use of DualSPHysics for embolus simulation**

The aim of this task was to assess the feasibility of using SPH to model the motion of emboli in blood flow. The open source software DualSPHysics (Crespo et al., 2011; Gomez-Gesteira et al., 2012b; Gomez-Gesteira et al., 2012a) was used to test the capability of SPH to perform simulations of solid particles carried by flow through a tube. The treatment of floating bodies by this algorithm is not accurate, but was considered sufficient for a preliminary test.

DualSPHysics is limited for internal flows as it currently has no boundary conditions for either periodic or inflow/outflow boundaries. The SPH implementation itself, however, is well developed and highly optimised, so it is useful for testing purposes. To create flow in a tube, two reservoirs were connected by a hollow cylinder, with outflow from the bottom reservoir handled by removing particles at the boundary.

Geometries were created from simple structures such as cylinders using the DualSPHysics GenCase tool. The code supports both 2D and 3D simulation, but since floating bodies are only currently supported in 3D, all simulations were 3D. Simulations were performed using the GPU implementation on NVIDIA Tesla M2050 graphics cards. The full “experiment” setup was 1 metre high, using around 1 million particles, particle spacing was 0.5mm. A cubic B-spline kernel was used with a Verlet stepping algorithm. Viscosity was artificial, using a “viscosity” value of 0.3.

### 3.3.3 Feasibility study of implementation of SPH code for embolus applications

Blood flow is internal and therefore must be modelled differently to the problems SPH is traditionally used for, such as astrophysical flows with open boundaries, and terrestrial flows where water forms a free surface at the air interface. Internal flows are usually the domain of conventional CFD solvers, since, given a closed boundary, solving for the flow is now simple with modern simulation packages and methods. When adding complexities to the simulation such as solids in the flow or flexible walls, conventional CFD becomes more difficult to work with, which is where the benefits of SPH become apparent.

Modelling steady flow in a pipe or network of pipes in SPH requires two additions to the theory already described: 1) Inlet/outlet boundary conditions at the pipe ends, and 2) Solid wall boundary conditions on the insides of the pipe. Inlets and outlets are relatively simple to implement. Although approaches vary, in all methods particles are introduced at the inlet and removed at the outlet to create a driving pressure through the pipe network. Solids are usually modelled as SPH particles placed over the boundary geometry. Fluid particles can interact with solid particles, but either the solid particles are fixed in place, or are governed by different physics to the fluid system.

As a preliminary study to investigate the feasibility of using SPH in blood flow problems, an SPH code was written in C++. A cubic b-spline kernel is used (Eq. (3.7)) with gradient Eq. (3.8). Although there are more accurate kernels, it is popular in the literature and sufficient for most simulations. All other equations are as described in section 3.3.1.

The inlet/outlet boundary condition is currently periodic. Periodic boundaries are simple to implement; particle interactions and motion are “looped” from

### *Section 3*

each edge of the simulation domain to the opposite edge. Flow is then generated by applying a force to all particles in the system, this acts as a pressure.

Solid wall boundaries are implemented by fixing some particles in place. Forces are calculated as normal for these particles, but the particle velocities are never updated. Since the boundary particles are mostly treated as normal SPH particles the boundary is weak and sufficient pressure will allow fluid particles to penetrate.

#### **3.3.4 Development of an in-house SPH code for embolus simulation**

Given the success of the DualSPHysics test and the SPH feasibility study, we continued by developing an in-house SPH code targeted at simulating emboli in the blood in 2D. Creating an in-house code was necessary since there are no publicly available SPH codes which are suited to simulation of internal flows such as in pipes or blood vessels, and most floating-body implementations in codes which are available are not suitable for emboli.

The in-house code will need to fulfil the following requirements:

1. Flow in a straight tube must be parabolic and match theory as resolution tends to infinity.
2. Emboli should migrate in the flow according to the Segre-Silberberg effect.
3. The code should be extendable to 3D in future.
4. The code should include flexible vessel walls or be able to incorporate flexible walls in future.

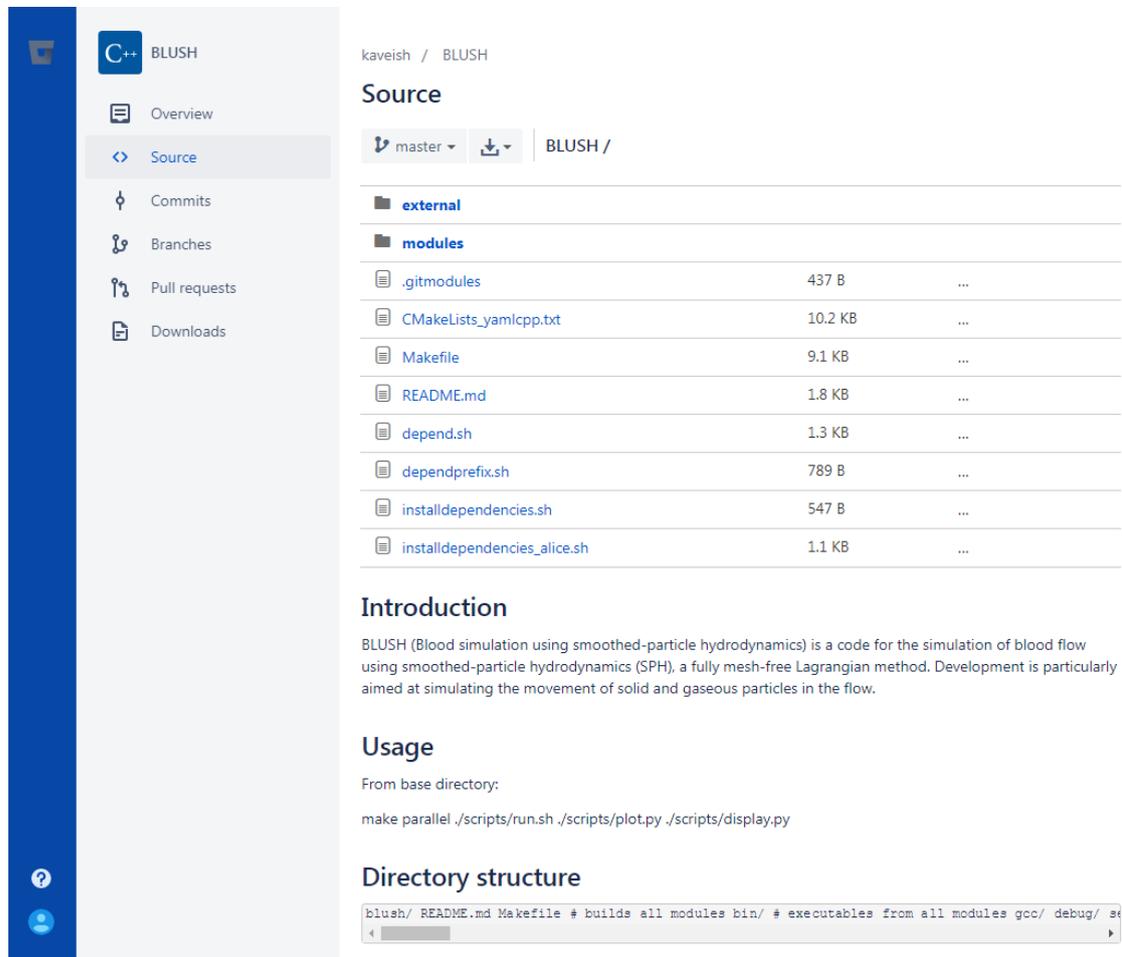
#### **BLUSH**

BLUSH (Blood simulation using smoothed-particle hydrodynamics) is the in-house code which was developed. The code comprises over 10,000 lines of C++

### Section 3

and python and is hosted on a publicly accessible website:

<https://bitbucket.org/kaveish/blush> (Figure 3.4).<sup>2</sup>



The screenshot shows the Bitbucket source page for the BLUSH repository. The page is titled "kaveish / BLUSH" and "Source". The left sidebar contains navigation links: Overview, Source (selected), Commits, Branches, Pull requests, and Downloads. The main content area shows the "Source" view for the "master" branch. It lists files and folders: "external" (folder), "modules" (folder), ".gitmodules" (437 B), "CMakeLists\_yamlicpp.txt" (10.2 KB), "Makefile" (9.1 KB), "README.md" (1.8 KB), "depend.sh" (1.3 KB), "dependprefix.sh" (789 B), "installdependencies.sh" (547 B), and "installdependencies\_alice.sh" (1.1 KB). Below the file list, there are sections for "Introduction" and "Usage". The "Introduction" section describes BLUSH as a code for simulating blood flow using smoothed-particle hydrodynamics (SPH). The "Usage" section provides instructions on how to build and run the code from the base directory. The "Directory structure" section shows a tree view of the repository files.

Figure 3.4 The BLUSH code is hosted on a public webpage at <https://bitbucket.org/kaveish/blush>. The site contains instructions on how to build and run BLUSH.

The SPH theory used in BLUSH is largely the same as described in section 0, but with replacements for the different equations, which are given below.

<sup>2</sup> The code is not included in this thesis since it would fill 140 pages if printed.

### Section 3

#### Kernels

##### Fluid interaction

For fluid-fluid interaction, a Wendland 2D C2 kernel is used:

$$W = \frac{7}{\pi h^2} (1-r)_+^4 (1+4r) \quad (3.42)$$

$$\nabla W = -\frac{140}{\pi h^2} (1-r)_+^3 r \quad (3.43)$$

where  $(\cdot)_+ \equiv \max\{0, \cdot\}$ .

##### Boundary interaction

For fluid-boundary interactions using the Monaghan and Kajtar (2009) force model, the kernel below is used.

Wendland 1D C4 (quintic):

$$W = \frac{3}{2} (1-r)_+^5 (1+5r+8r^2) \quad (3.44)$$

Monaghan and Kajtar (2009) normalise this to 1.77 at  $q = 0$  for the boundary force, which gives:

$$W = 1.77 (1-r)_+^5 (1+5r+8r^2) \quad (3.45)$$

#### Equation of state

We implement an equation of state for free-surface flows as well as internal flows. For our applications the internal flow equation is used, but since the free-surface equation is included in BLUSH, it is listed here for completeness.

For free surface flows we use the form of Monaghan (1994):

$$P = \frac{c_0^2 \rho_0}{\gamma} \left[ \left( \frac{\rho}{\rho_0} \right)^\gamma - 1 \right] \quad (3.46)$$

### Section 3

where  $c_0$  is the speed of sound,  $\rho_0$  the reference fluid density and  $\gamma \sim 7$  is the fluid polytropic constant.

For internal flows we use the equation of Morris et al. (1997):

$$P = c_0^2 \rho. \quad (3.47)$$

#### Continuity equation

We use the standard continuity equation below in most cases:

$$\frac{d\rho_a}{dt} = \sum_b m_b \vec{v}_{ab} \cdot \nabla_a W_{ab}. \quad (3.48)$$

#### Equation of motion (Momentum equation)

The equation of motion includes an additional term for the boundary force, if used, which is taken from Monaghan and Kajtar (2009).

$$\begin{aligned} \frac{d\vec{v}_a}{dt} = & - \sum_b m_b \left( \frac{P_b}{\rho_b^2} + \frac{P_a}{\rho_a^2} + \Pi_{ab} \right) \nabla_a W_{ab} \\ & + \sum_j (\vec{f}_{aj} - m_j \Pi_{aj} \nabla_a W_{aj}) \\ & + \vec{g} \end{aligned} \quad (3.49)$$

#### Artificial viscosity

We use the standard form of artificial viscosity:

$$\Pi_{ab} = \begin{cases} \frac{-\alpha \overline{c_{ab}} \mu_{ab}}{\rho_{ab}} & \vec{v}_{ab} \cdot \vec{r}_{ab} < 0 \\ 0 & \text{otherwise} \end{cases} \quad (3.50)$$

$\alpha = 1$  but can be varied depending on the problem,

### Section 3

$$\mu_{ab} = \frac{h\vec{v}_{ab} \cdot \vec{r}_{ab}}{r_{ab}^2 + \eta^2} \quad (3.51)$$

which gives a kinematic viscosity

$$\nu = \frac{1}{8} \alpha h \overline{c_{ab}} \quad (3.52)$$

#### Local sound speed

For local sound speed we use the standard form below:

$$c_a = \sqrt{\frac{\partial P_a}{\partial \rho_a}} = c_0 \left(\frac{\rho_a}{\rho_0}\right)^{(\gamma-1)/2} \quad (3.53)$$

#### Boundary force terms (Monaghan & Kajtar)

The  $\vec{f}_{ab}$  term in the equation of state is given by:

$$\vec{f}_{ab} = \frac{K \vec{r}_{ab}}{\beta r_{ab}^2} W(r_{ab}/h) \frac{2m_b}{m_a + m_b} \quad (3.54)$$

where the constants  $K$  and  $\alpha$  are chosen to provide the correct boundary force and viscosity depending on the Reynolds number of the problem as follows.

For example, for a static tank Reynolds number is given in terms of viscosity  $\nu$  and tank depth  $D$  by

$$\text{Re} = (gD)^{1/2} \frac{D}{\nu} \quad (3.55)$$

The Reynolds number depends on a characteristic maximum velocity, which in this case is

$$v_{\max} = \sqrt{gD} \quad (3.56)$$

which is used to get sound speed

### Section 3

$$c_0 = 10v_{\max} = 10\sqrt{gD} \quad (3.57)$$

which gives us  $K$

$$K = v_{\max}^2 = 0.01c_0^2 \quad (3.58)$$

Kinematic viscosity for a chosen Reynolds number is expressed by rearranging (3.55):

$$\nu = (gD)^{1/2} \frac{D}{\text{Re}} \quad (3.59)$$

$\alpha$  is then chosen to produce the correct viscosity for this sound speed and  $h$ .

$$\nu = \frac{1}{8} \alpha h c_0 \quad (3.60)$$

or

$$\alpha = \frac{8\nu}{h c_0} \quad (3.61)$$

### Time Integration

#### Notation

We use the following notation to describe time integration techniques:

$$\frac{d\vec{v}_a}{dt} = \vec{F}_a \quad (3.62)$$

$$\frac{d\rho_a}{dt} = D_a \quad (3.63)$$

$$\frac{d\vec{r}_a}{dt} = \vec{v}_a \quad (3.64)$$

### Section 3

#### Verlet Symplectic Integrator

This method is used by Monaghan and Kajtar (2008). The particle properties are first brought up to the half step, at which point pressures and force are recalculated, and used to progress the properties to the full step.

$$\vec{v}_a^{1/2} = \vec{v}_a^0 + \frac{\delta t}{2} \vec{F}_a^0 \quad (3.65)$$

$$\vec{r}_a^{1/2} = \vec{r}_a^0 + \frac{\delta t}{2} \vec{v}_a^0 \quad (3.66)$$

$$\rho_a^{1/2} = \rho_a^0 + \frac{\delta t}{2} D_a^0 \quad (3.67)$$

$$P_a^{1/2} = f(\rho_a^{1/2}) \quad (3.68)$$

$$\vec{v}_a^1 = \vec{v}_a^0 + \delta t \vec{F}_a^{1/2} \quad (3.69)$$

$$\vec{r}_a^1 = \vec{r}_a^0 + \frac{\delta t}{2} (\vec{v}_a^0 + \vec{v}_a^1) \quad (3.70)$$

$$\rho_a^1 = \rho_a^{1/2} + \frac{\delta t}{2} D_a^1 \quad (3.71)$$

$$P_a^1 = f(\rho_a^1). \quad (3.72)$$

(3.70) can also be written as:

$$\vec{r}_a^1 = \vec{r}_a^{1/2} + \frac{\delta t}{2} \vec{v}_a^1 \quad (3.73)$$

#### Time step

The length of the time step  $\delta t$  is given by

$$\delta t = \min(\delta t_c, \delta t_b) \quad (3.74)$$

where  $\delta t_c$  is a Courant condition (Courant et al., 1928; Monaghan, 1992)

$$\delta t_c = \min\left(\frac{h}{2c_a}\right) \quad (3.75)$$

and  $\delta t_b$  depends on boundary proximity

$$\delta t_b = \min\left(\frac{|r_{ab} - d|}{\sqrt{K}}\right) \quad (3.76)$$

### Implementation

The implementation is divided into initialisation and run stages. In initialisation, the particle properties are initialised and particles are added to cells in the computational domain. In the run stage, steps are followed in a repeating loop to calculate particle forces and evolve the particle properties.

The steps in the implementation are given below, with functions in the time-stepping algorithm listed with their input arguments.

### Initialisation

1.  $\vec{v}_0$ (inputfile)
2.  $\vec{r}_0$ (inputfile)
3.  $\rho_0$ (overdensity,  $\rho_{\text{ref}}$ )
4.  $P_0(\rho_0)$
5.  $t_0 = 0$
6.  $D_0 = 0$
7. Assign particles to cells

### Main simulation run

1.  $F_0(\vec{v}_0, \vec{r}_0, \rho_0, P_0)$
2.  $\vec{v}_{1/2}(\vec{v}_0, t_0, \vec{F}_0)$
3.  $\vec{r}_{1/2}(\vec{r}_0, t_0, \vec{v}_0)$
4.  $\rho_{1/2}(\rho_0, t_0, D_0)$
5.  $P_{1/2}(\rho_{1/2})$

Section 3

6. Loop particles for periodic boundary
7. Assign particles to cells
8.  $F_{1/2}(\vec{v}_{1/2}, \vec{r}_{1/2}, \rho_{1/2}, P_{1/2})$
9.  $\rightarrow dt(\min(c_{a1/2}), \min(\vec{r}_{ab1/2}))$
10.  $\vec{v}_1(\vec{v}_0, t_0, \vec{F}_{1/2})$
11.  $\vec{r}_1(\vec{r}_{1/2}, t_0, \vec{v}_1)$
12. Loop particles for periodic boundary
13. Assign particles to cells
14.  $D_1(\vec{r}_1, v_1)$
15.  $\rho_1(\rho_{1/2}, t_0, D_1)$
16.  $P_1(\rho_1)$

## **3.4 Results**

### **3.4.1 Introduction**

In this section we present the results of the DualSPHysics feasibility study, followed by the in-house SPH code feasibility study, and finally the results of the full in-house SPH code, BLUSH.

### **3.4.2 Results from DualSPHysics simulation of embolus**

Preliminary tests to determine the suitability of SPH for modelling the motion of objects in blood flow were performed using DualSPHysics, the results of which are shown in Figure 3.5. This generated regular flow for sufficient time to test the motion of a sphere in the fluid. The sphere was observed to spin, as is expected for an object in a flow gradient. The sphere also migrated from the initial position to the centre. The sphere migrated in all tests further than expected, without reaching an equilibrium position consistent with the tubular “pinch” expected from the Segré-Silberberg effect. This is probably due to the erroneous floating body algorithm used in the current version of DualSPHysics which is known to produce non-physical results.

### Section 3

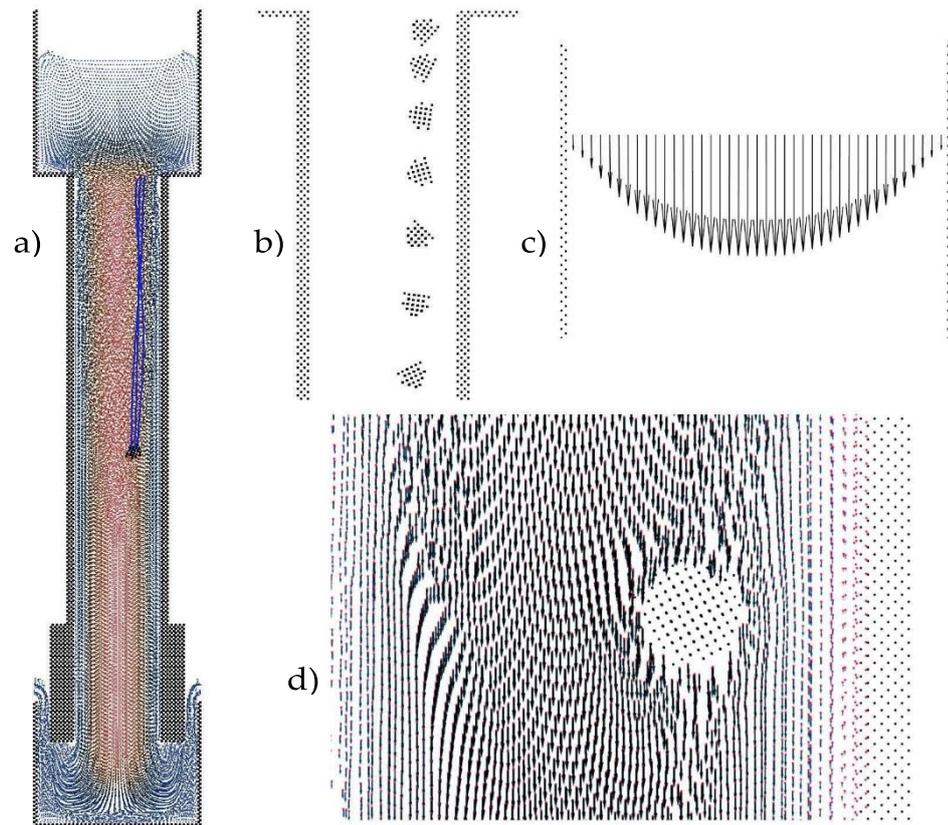


Figure 3.5 2D sections of 3D flow simulations. a) The initial low resolution case. Fluid flows from an upper to a lower reservoir with a neutrally buoyant sphere comprised of particles. The sphere migrates in the flow towards the centre of the tube. b) Magnified view of the sphere motion shows rotation in the flow. c) Flow vectors across the tube; the flow is parabolic as expected. d) High resolution case displaying flow around the sphere.

#### **Findings from DualSPHysics feasibility study**

SPH tests using DualSPHysics demonstrate the ease with which floating bodies can be included in SPH simulations. The motion of emboli within the flow appeared to be qualitatively realistic. With improved handling of fluid-solid interactions in our in-house code, accurate simulation of solids in internal flows should be possible.

Next steps include running these test simulations using our in-house code, improving implementation of embolus interactions, and simulating embolus trajectory at bifurcations to study behaviour.

### 3.4.3 Results from feasibility study of SPH code implementation

Fig. 2.2 shows a “dam break” simulated by our code. This example is commonly used to test SPH codes, as the results are intuitive and performance can be checked by eye. A body of fluid begins as a stationary block before collapsing into the empty space to one side. Some particle penetration is apparent as we used a flawed solid boundary treatment. As the edge boundary is periodic, the right side boundary also acts as the left boundary.

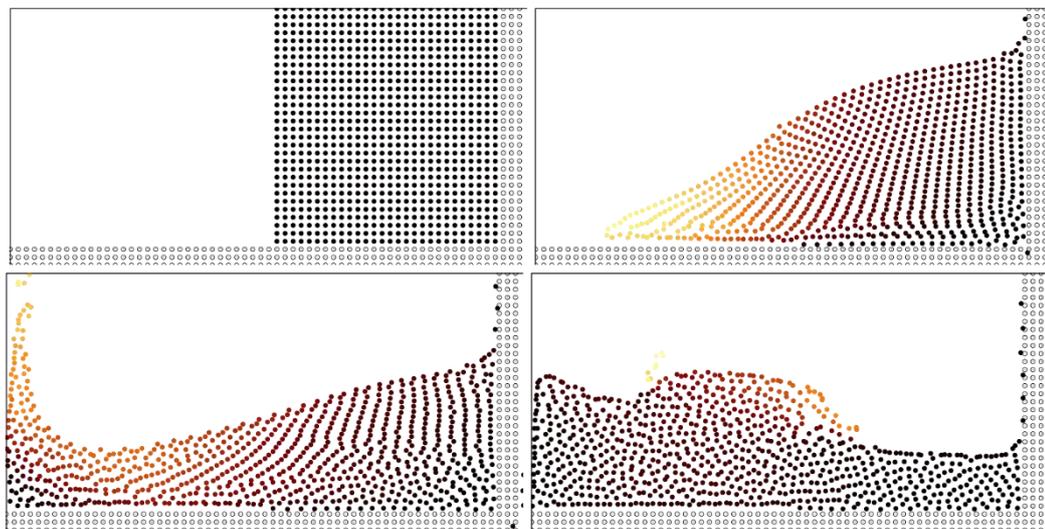


Figure 2.2: 2D SPH simulation of a “dam break”. Colour represents particle speed; orange particles are fast.

Laminar parabolic flow in a tube, developing under constant force, was simulated and the results are shown in Fig. 2.3. This test confirms that Poiseuille flow forms, even with a simple SPH implementation. The fluid ‘loops’ between edges using a periodic boundary condition.

### Section 3

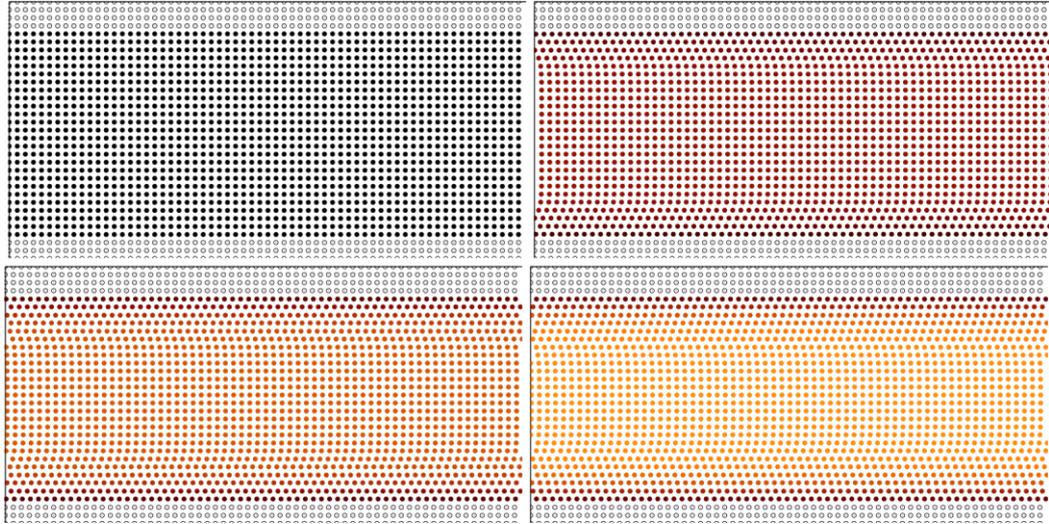


Figure 2.3: Development of parabolic flow in a tube due to a constant acceleration.

#### **Findings from the feasibility study**

Both simulations ran for 2-3 minutes; at this level of detail longer run times are unnecessary. If complexity and resolution were increased, simulation time would become an issue and the code would then need to be optimised.

The wall boundary implementation used in these tests was observed to lead to particle penetration through the vessel walls, which is a common problem in SPH simulations. There are various techniques for improving solid boundary interactions such as adding “mirror” particles that repel fluid particles using an opposing artificial velocity.

The periodic boundary condition used in our initial simulations was useful for this simple flow simulation, but does not allow proper control of flows at the boundary, and only works for a straight tube.

We have presented results from our preliminary SPH simulation showing a simple test simulation. This study shows that we are able to implement a working SPH code which can be expanded for investigating emboli in blood flow.

### 3.4.4 Results from in-house SPH code for embolus simulation

#### Validating against Poiseuille flow

The in-house SPH code was first tested to ensure that it could replicate Poiseuille flow. The tests performed are similar to those by Pazouki & Negrut (2015), who use test flow between two infinite parallel plates with Reynolds number of 12.73.

The transient expression of the velocity profile is given by (Morris, 1997)

$$v_x(y, t) = \frac{F}{2\nu} y(y - L) + \sum_{n=0}^{\infty} \frac{4FL^2}{\nu\pi^3(2n+1)^3} \sin\left(\frac{\pi y}{L}(2n+1)\right) \exp\left(-\frac{(2n+1)^2\pi^2\nu}{L^2}t\right) \quad (3.77)$$

where  $v_x$  is velocity,  $y$  is in the direction perpendicular to the plates,  $t$  is time,  $L$  is the distance between the plates,  $F$  is body force, and  $\nu$  is kinematic viscosity.

In the case that  $t \rightarrow \infty$ , the equation becomes

$$v_{x(y)} = \frac{F}{2\nu} y(y - L) \quad (3.78)$$

which is the steady flow solution. We can use this to calculate the mean velocity

$$v_{mean} = \int_L^0 v_x dy = \frac{FL^3}{12\nu} = \frac{FL^3\rho}{12\mu} \quad (3.79)$$

where  $\mu$  is dynamic viscosity.

Reynolds number for this case is expressed as

$$Re = \frac{\nu\rho L}{\mu} \quad (3.80)$$

which means the body force necessary to create a particular Reynolds number is

### Section 3

$$F = \frac{12Re\mu^2}{L^3\rho^2} \quad (3.81)$$

To emulate the test by Pazouki & Negrut (2015), we need  $Re=12.73$ . Using  $\rho = 1$ ,  $L = 1$ , and  $\mu = 0.01$ , we need a body force  $F = 0.0153$ , which would give  $v_{mean} = 0.127$ .

#### Particle placement issues

Currently particles must be placed on a grid. The actual boundary occurs  $dp/2$  away from the boundary particles, where  $dp$  is particle spacing. This means the boundary width varies according to  $dp$ . To complicate matters further, for a hex grid, the  $y$  spacing is defined, not  $dp$ . This means the actual boundary occurs  $2dp/\sqrt{3}$  away from the boundary.

We have investigated automatically moving the boundary such that it is consistent between values of  $dp$ , but this is complicated by only being applicable to boundaries that are fully aligned.

We resolved that, since at high resolution these effects become smaller, it would be best to persevere with the current method, but tweak inputs for each  $dp$  such that parameters remain constant e.g. Reynolds number.

Given this, the body force above must be adjusted due to the change in  $L$  caused by  $dp$ .

#### Adjusting input parameters

Since the true boundary created by boundary particles is offset from the boundary particle positions, we must use a vessel length equal to the distance between the true boundaries in our Poiseuille flow calculations. We recalculate the vessel length as  $L' = L - 4dp/\sqrt{3}$  which for  $dp = 0.025$  is 0.942 and for  $dp = 0.0125$  is 0.971. This then gives body forces of 0.0183 and 0.0167 for  $dp = 0.025$  and 0.0125, respectively.

### Section 3

The results for the full simulation are shown in Figure 3.6. Average flow varies by 0.3% from the theoretical value for  $dp = 0.0125$ . The evolving flow shows good agreement with theory.

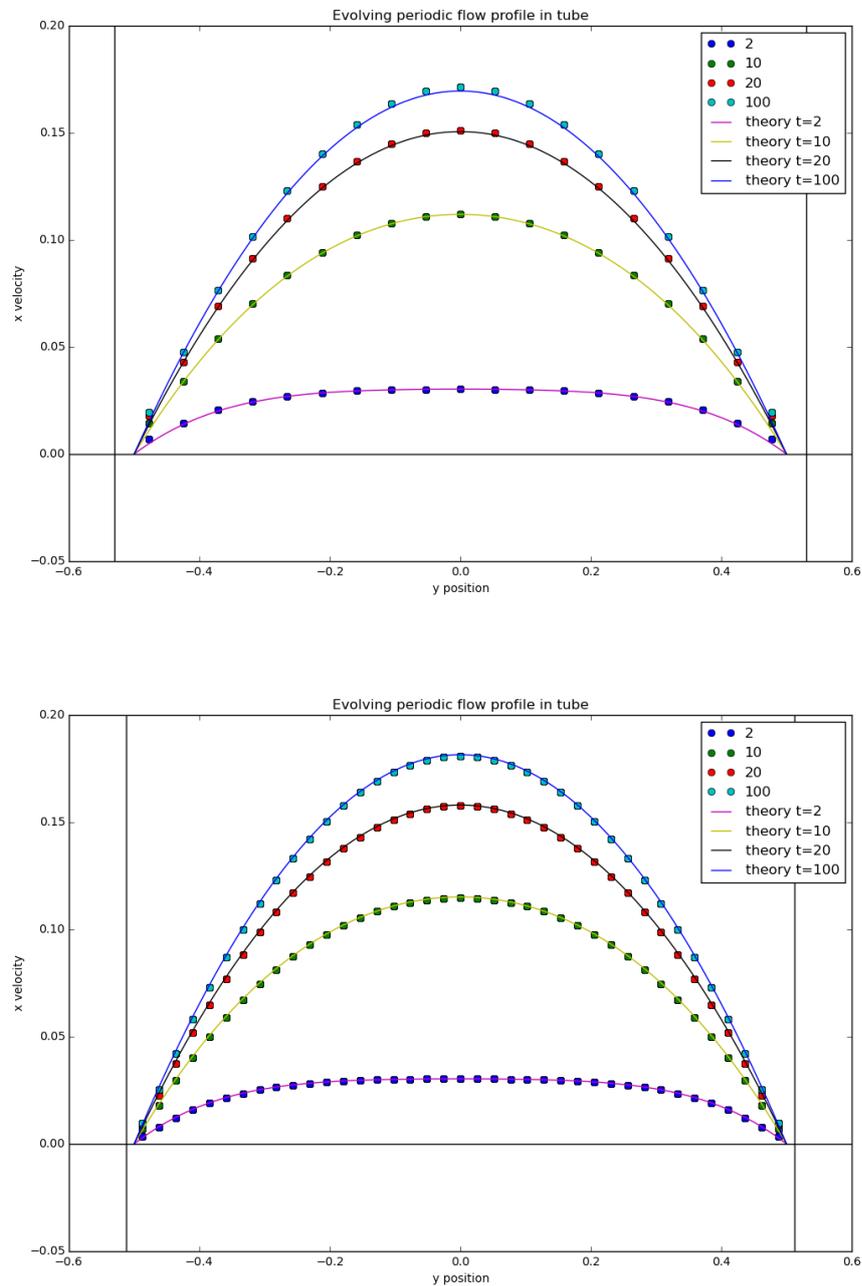


Figure 3.6 Validating the in-house code against Poiseuille flow evolution of flow between two infinite parallel plates. In the top plot, particle separation is 0.025 with 19 particles across the channel width. In the bottom plot, particle separation is 0.0125 with 39 particles across the channel width.

### Section 3

#### Testing body migration in flow

The in-house code was tested to determine whether it could reproduce the Segre-Silberberg effect. In 2D, we modelled an infinite cylinder travelling in the flow, between two infinite parallel plates. The fluid and cylinder were released from rest, with the fluid under the influence of a constant body force of 0.1. The cylinder was initially placed at multiple starting positions between the centre-plane and one parallel plate. The flow was symmetric to avoid the need to place the cylinder on both sides of the centre-plane. Simulations were run with Reynolds number 73.8; the cylinder diameter to channel width ratio was 0.125; tests were performed with  $dp=0.0125$  and  $dp=0.00625$  and channel width 1.

Results from one of these simulations are shown in Figure 3.7 and Figure 3.8. We can see that the presence of the cylinder only mildly perturbs the flow, but close to the cylinder viscous forces dominate and result in the fluid rotating with the cylinder at the cylinder boundary.

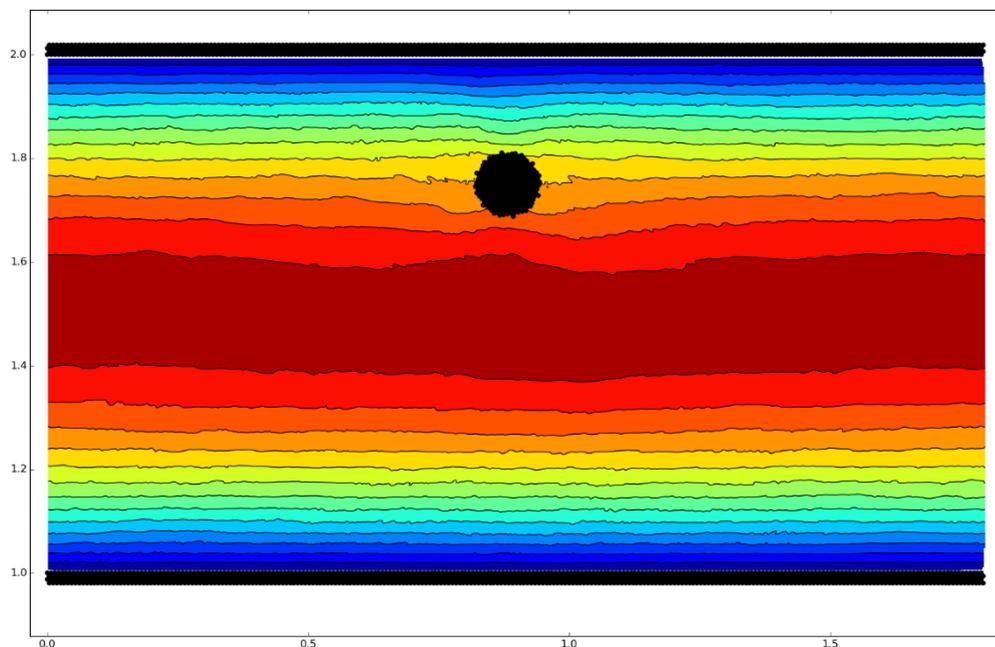


Figure 3.7 Contour plot of particle velocity in a simulation of an infinite cylinder moving in flow between two infinite parallel plates. The cylinder has reached its

### Section 3

equilibrium position in the y direction, but continues to rotate. This rotation causes local changes to the fluid.

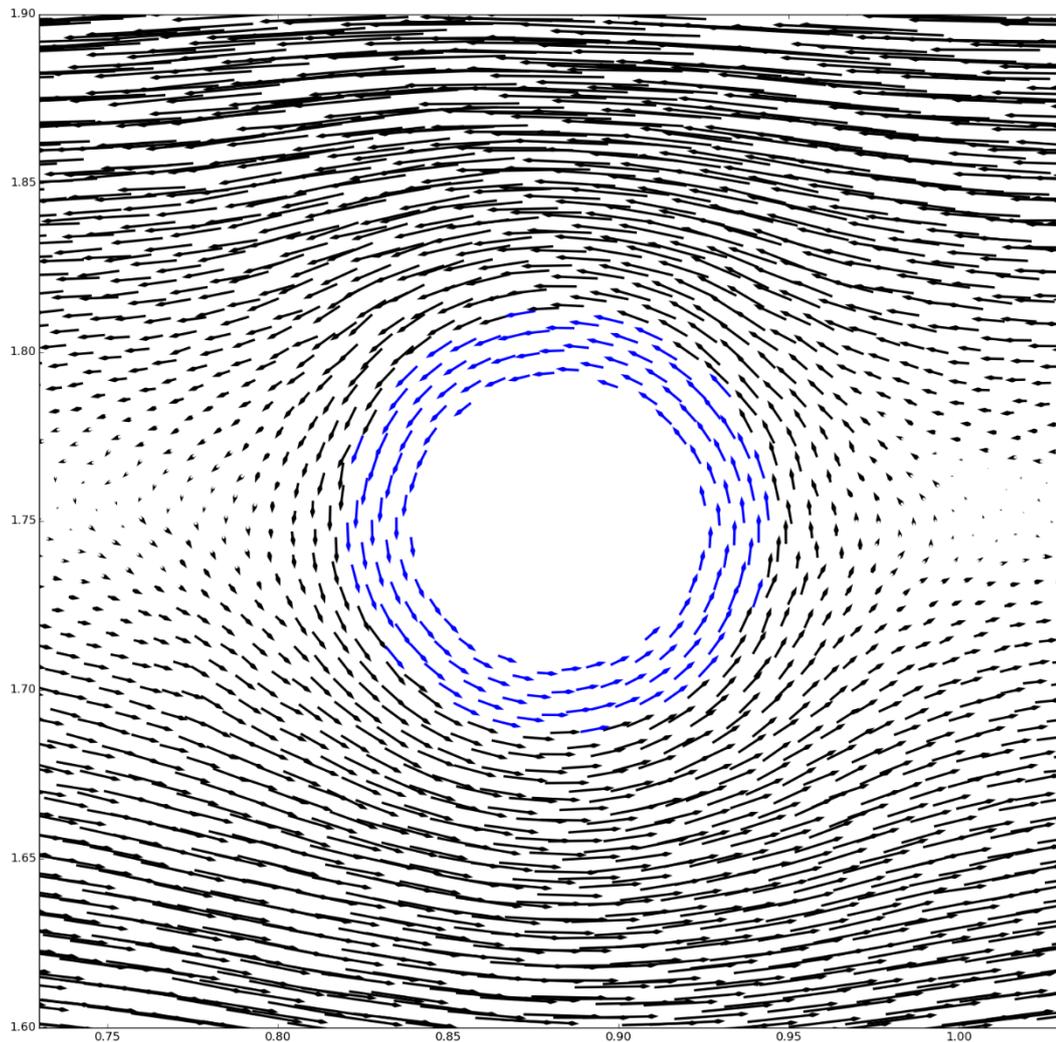


Figure 3.8 Close-up of an infinite cylinder travelling in flow between two parallel plates. Velocities are relative to the velocity of the cylinder and are plotted as arrows scaled according to their magnitude. Flow above the cylinder is travelling slower than the cylinder and flow below is travelling faster. This velocity difference causes the cylinder to rotate. The cylinder's rotation causes the local fluid to circulate around the cylinder.

To assess the effect of the cylinder's presence on flow evolution, the fluid's mean flow speed is shown in Figure 3.9 for the smallest  $dp$  simulation. The flow evolves smoothly and does not greatly change depending on released

### Section 3

cylinder position. This is to be expected from the figures above showing little disturbance to the flow from the cylinder.

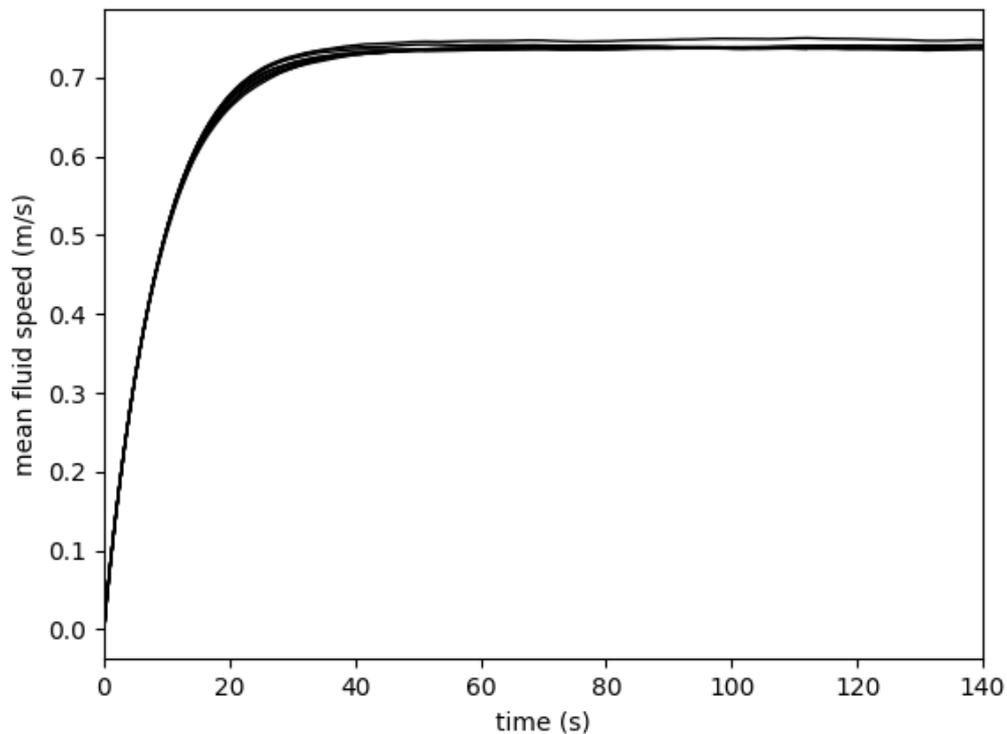


Figure 3.9 Development of steady flow in the high resolution simulations of a cylinder between two parallel plates. Results from all cylinder release locations are plotted, showing that generally the flow speed evolves independently of the cylinder location. For all the simulations the flow reaches equilibrium at around 40 seconds.

Finally, the results of all simulations are plotted in Figure 3.10. The simulations at largest particle spacing resulted in large oscillations of the cylinder's position compared with the smaller particle spacing. This is likely caused by a combination of reduced simulation fidelity with respect to the fluid, as well as the boundary becoming less circular due to the lower spatial resolution. At  $dp=0.0125$ , the cylinder appears hexagonal since the cylinder particles are also placed on a hexagonal grid. Despite the increased oscillations, our results compare well with each other and the results of Paxouki & Negrut (2015). The

Section 3

equilibrium position is 1.740 at 0.0125 dp and 1.747 at 0.00625 dp, where the maximum deviation from the mean is 1.4% of the channel width.

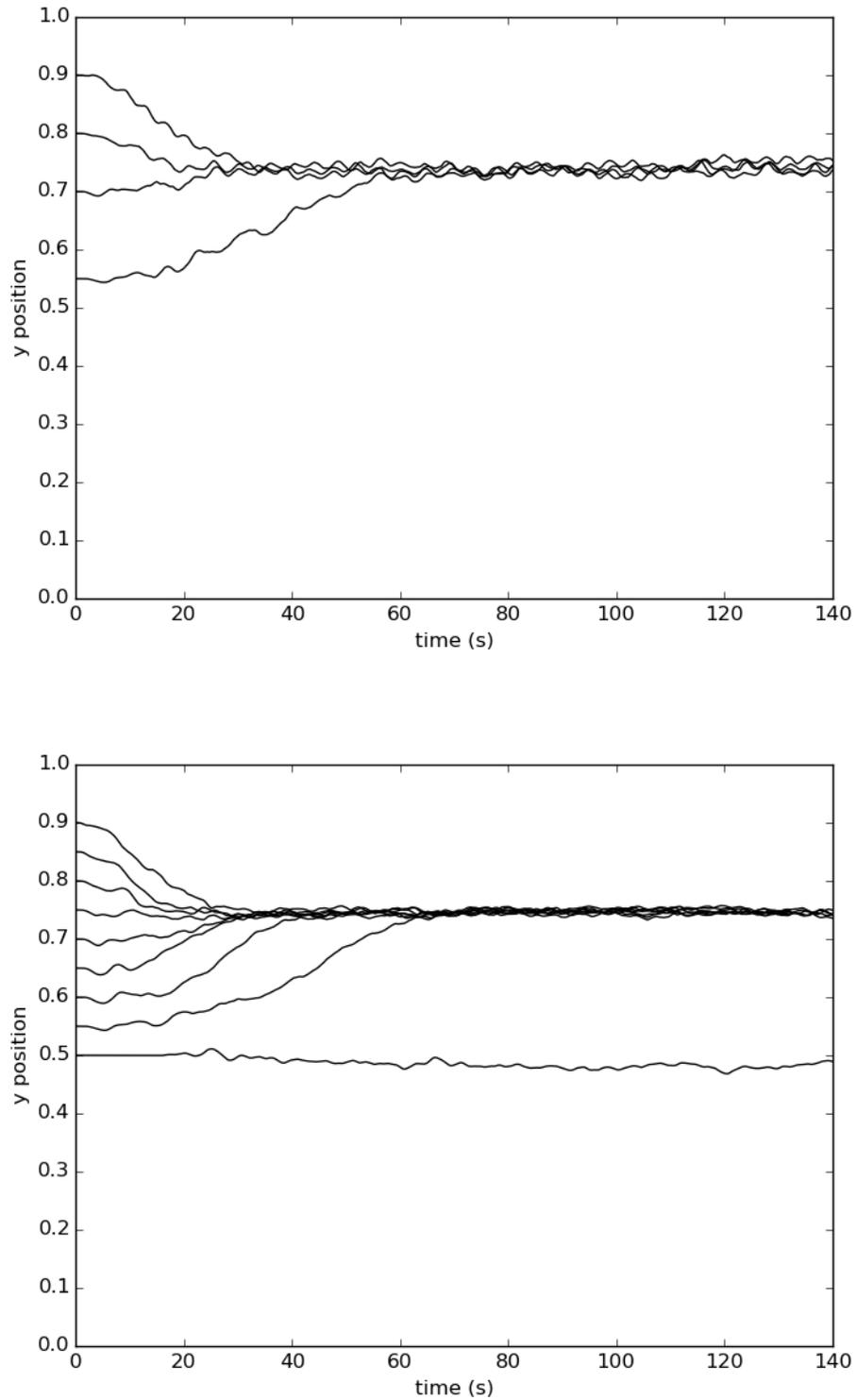


Figure 3.10 Migration of infinite cylinders between two infinite parallel plates released at different distances from the centre-plane. The top and bottom simulations used a

### *Section 3*

particle spacing of 1.25% and 0.625% of the channel width, respectively. The two resolutions agree, but the higher resolution simulations result in a more stable equilibrium position. The cylinder released at the centre-plane did not migrate, which is not surprising since the centre is a position of unstable equilibrium.

The lower resolution simulations involved 10,900 particles and took 2 hours to run on 8 cores of a 2.6GHz Intel Xeon Skylake CPU. The higher resolution simulations used 41,500 particles and took 16 hours. Discussion

#### **3.4.5 Discussion of DualSPHysics and preliminary in-house code feasibility studies**

The feasibility studies were successful and led to us continuing to use SPH as our main simulation method.

The feasibility of using SPH to simulate emboli in blood flow was tested using a DualSPHysics simulation, since that software is mature and well documented. The main drawback of this software is that it is designed for use with free-surface flows, and not internal flows. The problems with this are twofold, 1) The tools are not in place in the software to create a pipe flow simulation case, and 2) The SPH formulation is designed for free-surface flows, and therefore the results for internal flows are less accurate.

Due to the above limitations, the only way to test flow in a pipe was to create a “virtual experiment”, using a reservoir and gravity to generate flow. This is unnecessarily expensive computationally, but the results showed some migration of the sphere placed in the flow, which was sufficient for this test.

It was initially hoped that DualSPHysics could be adapted for simulating emboli since it already had a full SPH implementation, and there were few open-source SPH codes available at this time. Unfortunately, the code-base is extensive and heavily designed for free-surface flows, which is why we made the decision to develop an in-house code instead of adapting existing software.

### *Section 3*

The in-house code feasibility study was used to demonstrate that SPH simulations could be run using custom code before incorporating further advanced features needed for accurate simulation of emboli. We used a similar SPH formulation to that used in DualSPHysics for this code, and successfully ran validation tests including a dam-break and Poiseuille flow simulations. The results were sufficient to support continuing this line of development.

#### **3.4.6 Discussion of in-house SPH code for simulating emboli in the flow**

The completed SPH code has been tested to replicate both Poiseuille flow and the Segre-Silberberg effect successfully. It has a number of extensible features including different boundary implementations and SPH formulations. It also includes a flexible input structure to create simulation cases from generic shapes. The code is ready to be applied to embolus problems and is publicly available online for others to utilise and learn from.

The strengths of the code are speed, and features targeted explicitly at simulating of internal flows. The code has been designed from the ground-up to perform rapid simulations on a standard PC. The codebase makes heavy use of C++ templates to switch between features and move computation to compile-time wherever possible. Algorithms were adjusted against benchmarks to ensure optimum performance. In addition, the code is parallelised using OpenMP and includes vectorisation to optimise running the code using modern multi-core CPUs.

The code contains many useful features. Input is defined using human readable text files, where simulation properties as well as case geometry can be defined. Many shapes such as rectangles, circles, and triangles, as well as arbitrary polygons can be added and filled with fluid, boundary or object particles. Boundaries can be specified as either inward or outward facing. Particles

### *Section 3*

themselves can be placed on several types of grid at an arbitrary spacing. The case can include body forces, initial damping, background pressure, velocity constraints, and other options. The SPH formulation used here can also be customised from a number of included options for the equation of state, density calculation method, continuity equation, time integrator, and boundary particles.

Finally, the code has been designed for internal flows. Most publicly available SPH codes are designed for free-surface flows since those are the main SPH applications outside of astrophysics. These codes are difficult to use for internal flows, as we found in the DualSPHysics case study. Our in-house code both includes appropriate SPH formulations for internal flows, as well as providing easier case design methods to aid setting-up internal flow cases.

The code has several limitations, chiefly that it currently only supports 2D simulations. Many algorithms in the code were designed to allow an arbitrary number of dimensions, but significant work would be needed to extend this code to support 3D simulation. The code also has limited particle placement options since particles must be placed on a grid. This is common in SPH codes, but can cause issues when representing curved structures and creating boundaries. The output format of the code is not compliant with any SPH file formats as I found some of these formats prohibitive. However, it would be useful to standardise the file format in order to allow more generic tools to be used with our code. For instance, DualSPHysics output can be displayed in ParaView, which can make analysis easier.

In future it would be beneficial to address the limitations above, primarily making the code 3D. A 3D code would need new tools to create simulation cases and performance would be an increasing concern. Primarily, the code needs to be applied to new embolus problems. Currently we have a validated

### *Section 3*

code, but there was insufficient time to apply this to any clinically relevant scenarios. We are interested in studying, in particular, the trajectory of emboli at bifurcations, which is a problem this code would be well suited to investigating.

During my progress in this PhD, the study by Pazouki and Negrut (2015) was published, demonstrating the Segre-Silberberg effect using SPH, not only in 2D but also 3D. Although it was promising to see other authors interested in these problems, we were disappointed not to be able to be the first to publish this result. It should be recognised that the study by Pazouki and Negrut (2015) uses very different SPH and boundary formulations to our code, so it is good to see agreement between our two methods.

While developing the SPH code, I learned a great deal about the process of developing large software applications. Getting the code to produce correct results took many months, leading to years, of work. With time, I found that results I had believed to be accurate were not behaving correctly due to bugs in the code. I eventually learned about unit testing and implemented tests to ensure the algorithms I had implemented were producing the expected results. This led to fixing a number of issues, some of which were causing hard to detect, but real differences in physical behaviour. The code grew to over 10,000 lines, and because of this it has been difficult to get feedback on my code from others; since I was the sole developer and my supervisors were not familiar with the code base.

Personally, I enjoyed spending a significant length of time working on a single programming project. Doing so gave me many opportunities to learn skills I would not have learned otherwise. Since it was my first large application, there are certainly things I would change were I to do it again; but overall I am pleased with the final result.

### *Section 3*

I hope to see more medical applications of SPH appearing in the literature. It is a technique well suited to problems involving biological factors, but currently the technique is in its infancy, making other, more mature, methods more appealing. Regarding medical research, there is a tendency to focus on application rather than method, but in this case SPH offers real benefits to both which need transferring from current SPH applications into medical applications..

# Conclusion

In this thesis I have investigated the modelling of cerebral blood flow and solid and gaseous emboli in generic and patient-specific scenarios.

I began by studying fluid dynamic models of the cerebral vasculature, creating a 0D model of the cerebral arteries and validating it against a physical flow phantom created from MR data. The model showed good agreement with experimental results and was demonstrated to be a useful tool for studying patient-specific cerebral blood flow.

I demonstrated segmentation and processing of patient MRIs to extract quantitative geometric information in a robust manner. These techniques were applied to create a semi-automated set of scripts to measure MCAs from a set of 47 patients.

In the following section, I reviewed computational modelling of embolus transport before applying embolus sizing and trajectory simulation models to patient data. This process required use of the robust MCA measurement process developed in the previous section. The results of these simulations were briefly discussed.

In the final section, I studied smoothed-particle hydrodynamics as a potential method for modelling the movement of emboli in the blood. I performed feasibility studies to assess its suitability for this application, before creating my own code. The produced code was able to reproduce the Segré-Silberberg effect in two dimensions, demonstrating bodies in tube flow migrating to equilibrium positions. Accurately modelling this phenomenon is a necessity for any method which is used to model emboli in flow in 2D or 3D. This validation demonstrates the suitability of my code for application to emboli, and I hope future work can apply the model to embolus studies.

The main outputs of my PhD work are the validated generalised 0D model of the circle of Willis, the cerebral vasculature flow phantom which has already been applied in other studies, a robust and non-labour-intensive method for analysing the cerebral arteries from MR angiograms and measuring vessel diameters, measurement of MCAs in 47 patients and application of this to embolus sizing and trajectory models which I applied to these patients as part of a larger study, and finally the validated fluid dynamics code which can be applied in future to studying the motion of emboli. Regarding the last contribution, this is the only publicly available SPH code which has been designed specifically for embolus and blood flow studies.

I hope that the tools and methods I have developed will continue to be applied to the study of emboli, which is an interesting and complex field of study where research can have real impact on clinical outcomes for patients.

# Appendix A. Vasculature processing scripts.

## convert.py

```
import argparse
import subprocess
import logging
import string
import os
import re
import sys
import csv
from subprocess import Popen, PIPE
from time import time
from numpy import array, genfromtxt, arange

#vpath = '/opt/vmtk/bin/'
#vpath = 'D:/programs/vmtk/Install/bin/vmtk-exe.pyc '
vpath = ''

def setupLogging():
    global log
    log = logging.getLogger(__name__)
    log.setLevel(logging.DEBUG)
    sth = logging.StreamHandler()
    sth.setLevel(logging.INFO)
    log.addHandler(sth)
    fhnd = logging.FileHandler('vmtkscript.log', mode='w')
    fhnd.setLevel(logging.DEBUG)
    log.addHandler(fhnd)

def vmtkRun(cmd, t=False):
    t0 = time()
    p = subprocess.Popen(cmd, stdout=PIPE, stderr=PIPE)
    out, err = p.communicate()
    log.debug(out+err)
    checkError(out+err)
    if t:
        printTime(t0)

def checkError(out):
    err = False
    for line in out.split("\n"):
        if "error" in line or "Error" in line:
            print line.strip()
            err = True
    if err is True:
        raise Exception("VMTK error")

def printTime(t0):
    t1 = time()
    log.info("time = %.2f minutes." %((t1-t0)/60))

def run():
    #
```

```

# Read in input arguments.
#
parser = argparse.ArgumentParser(description='''
    Converts a DICOM image to vti.

    Example:
        convert.py dicomfolder -n file

    This segments the image found in "dicomfolder", and outputs
    file named "file-1-source.vti".
    ''')
)
parser.add_argument(
    'dicomfolder', metavar='DICOMFOLDER', help='''
    folder containing input dicom files to use.
    ''')
)
parser.add_argument(
    '-n', '--name', metavar='NAME', dest='filename',
default='out',
    help='''
    filename to use for output, format will be [name]-1-src.vti
    etc. (default: out).
    ''')
)
args = parser.parse_args()

#
# Process input arguments.
#

filename = args.filename

# Check DICOM folder exists
if not os.path.exists(args.dicomfolder):
    parser.error('Invalid DICOM folder path.')

# File naming scheme.
suffix = ['',
          '-1-src.vti',
          '-2-levelsets.vti',
          '-3-levelsetssurface.vtp',
          ]

#
# Convert to vti
#
log.info("Converting input to vti format and renaming
(vmtkimagereader).")
vmtkRun([(vpath+'vmtkimagereader'),
        '-f', 'dicom',
        '-d', args.dicomfolder,
        '-ofile', (filename+suffix[1])],
        True)

def main():
    setupLogging()

    try:
        run()
        return 0

```

```

    except Exception, err:
        log.error(str(err), exc_info=True)
        return 1

if __name__ == '__main__':
    main()

```

## segmentation.py

```

import argparse
import subprocess
import logging
import string
import os
import re
import sys
import csv
from subprocess import Popen, PIPE
from time import time
from numpy import array, genfromtxt, arange

vpath = ''

def setupLogging():
    global log
    log = logging.getLogger(__name__)
    log.setLevel(logging.DEBUG)
    sth = logging.StreamHandler()
    sth.setLevel(logging.INFO)
    log.addHandler(sth)
    fhnd = logging.FileHandler('vmtkscript.log', mode='w')
    fhnd.setLevel(logging.DEBUG)
    log.addHandler(fhnd)

def vmtkRun(cmd, t=False):
    t0 = time()
    p = subprocess.Popen(cmd, stdout=PIPE, stderr=PIPE)
    out, err = p.communicate()
    log.debug(out+err)
    checkError(out+err)
    if t:
        printTime(t0)

def checkError(out):
    err = False
    for line in out.split("\n"):
        if "error" in line or "Error" in line:
            print line.strip()
            err = True
    if err is True:
        raise Exception("VMTK error")

def printTime(t0):
    t1 = time()
    log.info("time = %.2f minutes." %((t1-t0)/60))

def run():
    #
    # Read in input arguments.
    #
    parser = argparse.ArgumentParser(description='')

```

Segments a .vti image of blood vessels using level sets.

Example:

```
segmentation.py -n file
```

This segments the image found in "file-1-source.vti", and outputs

```
files named like "file-2-levelsets.vti" etc.
'''
)
parser.add_argument(
    '-n', '--name', metavar='NAME', dest='filename',
default='out',
    help='''
    filename to use for output, format will be [name]-1-src.vti
    etc. (default: out).
    '''
)
args = parser.parse_args()

#
# Process input arguments.
#

filename = args.filename

# File naming scheme.
suffix = ['',
          '-1-src.vti',
          '-2-levelsets.vti',
          '-3-levelsetssurface.vtp',
          ]

#
# Begin segmentation pipeline
#

log.info("Creating surface with level sets
(vmtklevelsetsegmentation).")
print '''Recommended settings:
Initialisation type: 1, fast marching works well
Lower threshold: 200 (tested), raise if accepting too much noise
Upper threshold: none
Seeds: Place at ICA/MCA/ACA bifurcations
Targets: Place at MCA ends
Level set parameters: 300 steps, default others'''
vmtkRun([(vpath+'vmtklevelsetsegmentation'),
        '-ifile', (filename+suffix[1]),
        '-ofile', (filename+suffix[2])],
        True)

log.info("Convert level sets to surface with marching cubes
(vmtkmarchingcubes).")
vmtkRun([(vpath+'vmtkmarchingcubes'),
        '-ifile', (filename+suffix[2]),
        '-ofile', (filename+suffix[3])],
        True)

def main():
    setupLogging()
```

```

try:
    run()
    return 0
except Exception, err:
    log.error(str(err), exc_info=True)
    return 1

if __name__ == '__main__':
    main()

```

## clipmesh.py

```

import argparse
import subprocess
import logging
import string
import os
import re
import sys
import csv
from subprocess import Popen, PIPE, call
from time import time
from numpy import array, genfromtxt, arange

#vpath = '/opt/vmtk/bin/'
#vpath = 'D:/programs/vmtk/Install/bin/vmtk-exe.pyc '
vpath = ''

def setupLogging():
    global log
    log = logging.getLogger(__name__)
    log.setLevel(logging.DEBUG)
    sth = logging.StreamHandler()
    sth.setLevel(logging.INFO)
    log.addHandler(sth)
    fhnd = logging.FileHandler('vmtkscript.log', mode='w')
    fhnd.setLevel(logging.DEBUG)
    log.addHandler(fhnd)

def vmtkRun(cmd, t=False):
    t0 = time()
    p = subprocess.Popen(cmd, stdout=PIPE, stderr=PIPE)
    out, err = p.communicate()
    log.debug(out+err)
    checkError(out+err)
    if t:
        printTime(t0)

def checkError(out):
    err = False
    for line in out.split("\n"):
        if "error" in line or "Error" in line:
            print line.strip()
            err = True
    if err is True:
        raise Exception("VMTK error")

def printTime(t0):
    t1 = time()
    log.info("time = %.2f minutes." %((t1-t0)/60))

```

```

def run():
    #
    # Read in input arguments.
    #
    parser = argparse.ArgumentParser(description='''
        Clips mesh using vmtksurfaceclipper.

        Example:
            clipmesh.py -n filename

        This processes the file named filename-3-levelsetssurface.vtp.
        A backup is made to filename-3-levelsetssurface.vtp.bak.
        ''')
    parser.add_argument(
        '-n', '--name', metavar='NAME', dest='filename',
        default='out',
        help='''
            filename to use for output, format will be [name]-1-src.vti
            etc. (default: out).
            ''')
    args = parser.parse_args()

    # Check file exists
    if not os.path.exists(args.filename + '-3-levelsetssurface.vtp'):
        parser.error('Invalid file root name, path not found.')

    filename = args.filename

    # File naming scheme.
    suffix = [' ',
              '-1-src.vti',
              '-2-levelsets.vti',
              '-3-levelsetssurface.vtp',
              '-4-centerlines.vtp',
              '-5-brextr.vtp',
              '-6-brgeom.dat',
              '-7-brclip.vtp',
              '-8-brsect.vtp',
              '-9-brsect.dat'
             ]

    #
    # Backup mesh
    #
    call(["cp", (filename+suffix[3]), (filename+suffix[3]+'.bak')])

    #
    # Clip mesh
    #
    log.info("Running surface clipper.")
    vmtkRun([(vpath+'vmtksurfaceclipper'),
             '-ifile', (filename+suffix[3]),
             '-ofile', (filename+suffix[3])],
            True)

def main():
    setupLogging()

    try:

```

```

        run()
        return 0
    except Exception, err:
        log.error(str(err), exc_info=True)
        return 1

if __name__ == '__main__':
    main()

```

## geometry.py

```

import argparse
import subprocess
import logging
import string
import os
import re
import sys
import csv
from subprocess import Popen, PIPE, call
from time import time
from numpy import array, genfromtxt, arange

#vpath = '/opt/vmtk/bin/'
#vpath = 'D:/programs/vmtk/Install/bin/vmtk-exe.pyc '
vpath = ''

def setupLogging():
    global log
    log = logging.getLogger(__name__)
    log.setLevel(logging.DEBUG)
    sth = logging.StreamHandler()
    sth.setLevel(logging.INFO)
    log.addHandler(sth)
    fhnd = logging.FileHandler('vmtkscrip.log', mode='w')
    fhnd.setLevel(logging.DEBUG)
    log.addHandler(fhnd)

def vmtkRun(cmd, t=False):
    t0 = time()
    p = subprocess.Popen(cmd, stdout=PIPE, stderr=PIPE)
    out, err = p.communicate()
    log.debug(out+err)
    checkError(out+err)
    if t:
        printTime(t0)

def checkError(out):
    err = False
    for line in out.split("\n"):
        if "error" in line or "Error" in line:
            print line.strip()
            err = True
    if err is True:
        raise Exception("VMTK error")

def printTime(t0):
    t1 = time()
    log.info("time = %.2f minutes." %((t1-t0)/60))

def run():

```

```

#
# Read in input arguments.
#
parser = argparse.ArgumentParser(description='''
    Process 3d model of patient cerebral vasculature with VMTK for
    extraction of vessel geometrical data.

    Example:
        geometry.py -n filename

    This processes the file named filename-3-levelsetssurface.vtp.
    '''
)
parser.add_argument(
    '-n', '--name', metavar='NAME', dest='filename',
default='out',
    help='''
    filename to use for output, format will be [name]-1-src.vti
    etc. (default: out).
    '''
)
args = parser.parse_args()

#
# Process input arguments.
#

# Check file exists
if not os.path.exists(args.filename + '-3-levelsetssurface.vtp'):
    parser.error('Invalid file root name, path not found.')

filename = args.filename

# File naming scheme.
suffix = ['',
    '-1-src.vti',
    '-2-levelsets.vti',
    '-3-levelsetssurface.vtp',
    '-4-centerlines.vtp',
    '-5-brextr.vtp',
    '-6-brgeom.dat',
    '-7-brclip.vtp',
    '-8-brsect.vtp',
    '-9-brsect.dat'
]

#
# Begin geometric analysis pipeline
#

# Compute centerlines file (requires user input)
log.info("Computing centerlines file (vmtkcenterlines)")
print '''Recommended options:
Seeds: Single seed on one ICA shortly before ICA bifurcation.
Targets: On other ICA at same position, at end of all branches
        from MCAs, a short way up each ACA.'''
vmtkRun([(vpath+'vmtkcenterlines'),
    '-ifile', (filename+suffix[3]),
    '-ofile', (filename+suffix[4])],
    True)

```

```

    log.info(''Extracting branches and branch geometries
(vmtkbranchextractor,
    branchgeometry,
    branchclipper,
    branchsections)''')
    vmtkRun([(vpath+'vmtkbranchextractor'),
        '-ifile', (filename+suffix[4]),
        '-ofile', (filename+suffix[5]),
        '-radiusarray@', 'MaximumInscribedSphereRadius',
        '--pipe',
        (vpath+'vmtkbranchgeometry'),
        '-ofile', (filename+suffix[6]),
        '--pipe',
        (vpath+'vmtkbranchclipper'),
        '-ifile', (filename+suffix[3]),
        '-ofile', (filename+suffix[7]),
        '--pipe',
        (vpath+'vmtkbranchsections'),
        '-ofile', (filename+suffix[8])],
        True)

    log.info('Outputting geometric data as .dat file
(vmtksurfacewriter)')
    vmtkRun([(vpath+'vmtksurfacereader'),
        '-ifile', (filename+suffix[8]),
        '--pipe',
        (vpath+'vmtksurfacewriter'),
        '-ofile', (filename+suffix[9]),
        '-celldata', '1'],
        True)

    call('scriptdone.py')

def main():
    setupLogging()

    try:
        run()
        return 0
    except Exception, err:
        log.error(str(err), exc_info=True)
        return 1

if __name__ == '__main__':
    main()

```

## displaygeometry.py

```

import argparse
import subprocess
import logging
import string
import os
import re
import sys
import csv
from subprocess import Popen, PIPE
from time import time
from numpy import array, genfromtxt, arange

#vpath = '/opt/vmtk/bin/'

```

```

#vpath = 'D:/programs/vmtk/Install/bin/vmtk-exe.pyc '
vpath = ''

def setupLogging():
    global log
    log = logging.getLogger(__name__)
    log.setLevel(logging.DEBUG)
    sth = logging.StreamHandler()
    sth.setLevel(logging.INFO)
    log.addHandler(sth)
    fhnd = logging.FileHandler('vmtkscript.log', mode='w')
    fhnd.setLevel(logging.DEBUG)
    log.addHandler(fhnd)

def vmtkRun(cmd, t=False):
    t0 = time()
    p = subprocess.Popen(cmd, stdout=PIPE, stderr=PIPE)
    out, err = p.communicate()
    log.debug(out+err)
    checkError(out+err)
    if t:
        printTime(t0)

def checkError(out):
    err = False
    for line in out.split("\n"):
        if "error" in line or "Error" in line:
            print line.strip()
            err = True
    if err is True:
        raise Exception("VMTK error")

def printTime(t0):
    t1 = time()
    log.info("time = %.2f minutes." %((t1-t0)/60))

def run():
    #
    # Read in input arguments.
    #
    parser = argparse.ArgumentParser(description='''
        Display geometric information computed using VMTK.

        Example:
            displaygeometry.py -n filename

        This displays files named in the form filename-4-
        centerlines.vtp etc.
        ''')
    parser.add_argument(
        '-n', '--name', metavar='NAME', dest='filename',
        default='out',
        help='''
            filename to use for output, format will be [name]-4-
            centerlines.vtp
            etc. (default: out).
        ''')
    args = parser.parse_args()

```

```

#
# Process input arguments.
#

# Check file exists
if not os.path.exists(args.filename + '-3-levelsetssurface.vtp'):
    parser.error('Invalid file root name, path not found.')

filename = args.filename

# File naming scheme.
suffix = [' ',
          '-1-src.vti',
          '-2-levelsets.vti',
          '-3-levelsetssurface.vtp',
          '-4-centerlines.vtp',
          '-5-brextr.vtp',
          '-6-brgeom.dat',
          '-7-brclip.vtp',
          '-8-brsect.vtp',
          '-9-brsect.dat'
         ]

#
# Display geometric info
#

log.info("Displaying geometric information.")
log.info("Press q twice in window to view all information.")
vmtkRun([(vpath+'vmtkrenderer'),
         '--pipe',
         (vpath+'vmtkcenterlineviewer'),
         '-ifile', (filename+suffix[5]),
         '-cellarray', 'GroupIds',
         '--pipe',
         (vpath+'vmtksurfaceviewer'),
         '-ifile', (filename+suffix[8]),
         '-array', 'BranchSectionDistanceSpheres',
         '-scalarrange', '0.0', '4.0',
         '-celldata', '1',
         '--pipe',
         (vpath+'vmtksurfaceviewer'),
         '-ifile', (filename+suffix[3]),
         '-opacity', '0.25'],
        True)

def main():
    setupLogging()

    try:
        run()
        return 0
    except Exception, err:
        log.error(str(err), exc_info=True)
        return 1

if __name__ == '__main__':
    main()

```

## branchdiameters.py

```
import argparse
import csv
import numpy
import sys

def main():
    # Read in input arguments.
    parser = argparse.ArgumentParser(description='''
    Extract branch diameters from .dat output of
    vmtkbranchsections.

    .dat must have been produced using:
        vmtksurfacewriter -ifile branchsections.vtp
                        -ofile out.dat
                        -celldata 1

    Example:
        branchdiameters.py -n file.dat
    ''')
    parser.add_argument(
        '-n', '--name', metavar='NAME', dest='filename',
        default='out-9-brsect.dat',
        help='''
        input dat source file to use, must be created from
        vmtkbranchsections output (Default: out-9-brsect.dat).
    ''')
    parser.add_argument(
        '-r', nargs='*', metavar='RMCAGROUPS', dest='rmcagroups', help='''
        groups belonging to the RMCA.
    ''')
    parser.add_argument(
        '-l', nargs='*', metavar='LMCAGROUPS', dest='lmcagroups', help='''
        groups belonging to the LMCA.
    ''')
    args = parser.parse_args()

    if args.rmcagroups:
        args.rmcagroups = map(int, args.rmcagroups)
    if args.lmcagroups:
        args.lmcagroups = map(int, args.lmcagroups)

    filename = args.filename

    # Process .dat file
    with open(filename, 'rb') as csvfile:
        reader = csv.reader(csvfile, delimiter=' ')
        group = []
        area = []
        minimum = []
        maximum = []
        next(reader, None) # skip headers
        for row in reader:
            group.append(int(float(row[3]))) # convert to integer
            area.append(float(row[4]))
```

```

# Create list of lists containing areas for each group
groups = [[] for i in range(max(group)+1)]
for i in range(len(group)):
    groups[group[i]].append(area[i])

if args.rmccgroups or args.lmccgroups:
    print '{0:>5s} {1:>5s} {2:>9s} {3:>6s} {4:>16s}'.format(
        'Vessel',
        'Area',
        'Diameter',
        'Stdev',
        'NumCrossSections')

if args.rmccgroups:
    rmcclist = []
    for grp in args.rmccgroups:
        rmcclist = rmcclist + groups[grp]

    print rmcclist
    area = numpy.mean(rmcclist)
    diameter = numpy.sqrt(4*numpy.mean(rmcclist)/numpy.pi)
    stdev = numpy.std([numpy.sqrt(4*x/numpy.pi) for x in rmcclist])

    print '{0:5s} {1:6.2f} {2:9.2f} {3:6.2f} {4:16d}'.format(
        'RMCA',
        round(area,2),
        round(diameter,2),
        round(stdev,2),
        len(rmcclist))

if args.lmccgroups:
    lmcclist = []
    for grp in args.lmccgroups:
        lmcclist = lmcclist + groups[grp]

    area = numpy.mean(lmcclist)
    diameter = numpy.sqrt(4*numpy.mean(lmcclist)/numpy.pi)
    stdev = numpy.std([numpy.sqrt(4*x/numpy.pi) for x in lmcclist])

    print '{0:5s} {1:6.2f} {2:9.2f} {3:6.2f} {4:16d}'.format(
        'LMCA',
        round(area,2),
        round(diameter,2),
        round(stdev,2),
        len(lmcclist))

print

print '{0:>5s} {1:>6s} {2:>9s} {3:>6s} {4:>16s}'.format(
    'Group',
    'Area',
    'Diameter',
    'Stdev',
    'NumCrossSections')
for i in range(0, len(groups)):
    if len(groups[i]) != 0:
        area = numpy.mean(groups[i])
        diameter = numpy.sqrt(4*numpy.mean(groups[i])/numpy.pi)
        stdev = numpy.std([numpy.sqrt(4*x/numpy.pi) for x in groups[i]])
        print '{0:5d} {1:6.2f} {2:9.2f} {3:6.2f} {4:16d}'.format(

```

```
        i,  
        round(area,2),  
        round(diameter,2),  
        round(stdev,2),  
        len(groups[i]))  
  
if __name__ == '__main__':  
    main()
```

# Appendix B. Batch embolus propagation simulation scripts.

A script was used to submit batch jobs for all patients to be run on a computing cluster. The jobs were based on a template job script shown below:

```
#PBS -o RUNDIR/out/BICI_stdout_output.txt
#PBS -e RUNDIR/out/BICI_stdout_error.txt
#PBS -N emboli_BICI
#PBS -l walltime=02:00:00:00
#PBS -l vmem=1gb

set -x

path=~/bubblesim-out/
runpath=RUNDIR/
outpath="$runpath"/out/
inpath="$runpath"/in/
binpath="$path"/bin/
jobpath="$runpath"/jobs/

mkdir -p "$runpath" "$outpath" "$inpath"/running "$inpath"/done "$jobpath"/running
"$jobpath"/done

mv "$inpath"/todo/BICI.txt "$inpath"/running/

runtime=$((RUNTIME+7200))

"$binpath"/EmbolusSimulationBifurcationInputFileGas.exe \
0.1 0.1 0.0 0.0 "$runtime" 30 "$runtime" 1 1 \
"$outpath"/BICI_perc.txt "$outpath"/BICI_time.txt "$outpath"/BICI_space.txt
"$outpath"/BICI_equil.txt \
"$outpath"/BICI_node.txt \
20 3.0 RADIUS 1e-1 1 100 \
"$inpath"/running/BICI.txt

mv "$inpath"/running/BICI.txt "$inpath"/done/
```

The template script was modified for each patient by another script which replaced “RUNDIR” with the target output directory for that patient, “RUNTIME” with the time of the last embolus introduced to the vessel tree, and “RADIUS” with the MCA radius for that patient. This script is also provided as supplemental material:

```
#!/bin/bash

#set -x
set -e

# get run directory
if [ ! $# -eq 1 ]; then
    echo "Must supply run directory as argument"
    exit 1
fi

rundir=$1
```

```

if [[ ! -d "$rundir" ]]; then
    echo "Run directory does not exist"
    exit 1
fi

# expand run directory
rundir=$(cd $rundir; pwd)/
echo $rundir

# get job directory
jobdir="$rundir"/jobs/todo/
mkdir -p "$jobdir"

# change to script directory
dir=`dirname $0`
cd $dir

# lookup MCA diameters, store in array
while IFS=$'\t' read -r -a tmpArray
do
    echo ${tmpArray[0]}
    rightArray[${tmpArray[0]}]=${tmpArray[1]}
    leftArray[${tmpArray[0]}]=${tmpArray[2]}
done < "$rundir/in/mca.txt"

# loop through input files and extract names
for f in $(find "$rundir/in/todo" -name '*.txt' | sort); do
    name=$(basename $f .txt)
    sed s/BICI/$name/g <job-base.sub >"$jobdir"/"$name".sub
    # get number and left/right from name
    side=$(echo $name | cut -d'_' -f 2)
    number=$(echo $name | cut -d'_' -f 1)
    number=$(echo $number | cut -c 5,6,7)
    # store current diameter
    if [ $side = "R" ]; then
        diameter=${rightArray[$number]}
    elif [ $side = "L" ]; then
        diameter=${leftArray[$number]}
    else
        echo "ERROR: Filename must be in wrong format"
    fi
    radius=`echo "scale=3; $diameter / 2" | bc -l`
    # put radius in job file
    sed -i s:RADIUS:"$radius": "$jobdir"/"$name".sub
    # put run directory in job file
    sed -i s:RUNDIR:"$rundir": "$jobdir"/"$name".sub
    # get time of last embolus in each input file
    totaltime=$(tac "$f" |egrep -m 1 . | cut -f1)
    # cast float to integer
    totaltime=${totaltime%.*}
    # put time in job file
    sed -i s:RUNTIME:$totaltime: "$jobdir"/"$name".sub
    echo $name $number $side $diameter $totaltime
done

# make output folder (required for stdout stderr in job.sub)
mkdir -p "$rundir"/out

```

Once the job scripts were generated, all jobs were submitted with the following script:

```

#!/bin/bash

set -e
#set -x

# get run directory
if [ ! $# -eq 1 ]; then

```

```
    echo "Must supply run directory as argument"
    exit 1
fi

rundir=$1

if [[ ! -d "$rundir" ]]; then
    echo "Run directory does not exist"
    exit 1
fi

# expand run directory
rundir=$(cd $rundir; pwd)/
echo $rundir

# get job directory
jobdir="$rundir"/jobs/todo/

# loop through job files and extract names
for f in $(find "$jobdir" -name '*.sub'); do
    qsub $f
done
```

The output from each job is a text file containing, among other information, the instantaneous number of blocked end arterioles at each time during and after surgery. Each job took up to 2 days to complete, running on a single CPU core.

# Bibliography

3D Slicer. (n.d.). Retrieved from <http://www.slicer.org>

Abolfazli, E., Fatouraee, N., & Vahidi, B. (2014). Dynamics of motion of a clot through an arterial bifurcation: a finite element analysis. *Fluid Dynamics Research*, 46(5), 55505. <http://doi.org/10.1088/0169-5983/46/5/055505>

Adami, S., Hu, X. Y., & Adams, N. a. (2012). A generalized wall boundary condition for smoothed particle hydrodynamics. *Journal of Computational Physics*, 231(21), 7057–7075. <http://doi.org/10.1016/j.jcp.2012.05.005>

Akinci, N., Ihmsen, M., Akinci, G., Solenthaler, B., & Teschner, M. (2012). Versatile rigid-fluid coupling for incompressible SPH. *ACM Transactions on Graphics*, 31(4), 1–8. <http://doi.org/10.1145/2185520.2185558>

Alastruey, J., Parker, K. H., Peiró, J., Byrd, S. M., & Sherwin, S. J. (2007). Modelling the circle of Willis to assess the effects of anatomical variations and occlusions on cerebral flows. *Journal of Biomechanics*, 40(8), 1794–1805. Retrieved from <http://www.ncbi.nlm.nih.gov/pubmed/17045276>

Alnaes, M. S., Isaksen, J., Mardal, K.-A., Romner, B., Morgan, M. K., & Ingebrigtsen, T. (2007). Computation of hemodynamics in the circle of Willis. *Stroke*, 38(9), 2500–5. <http://doi.org/10.1161/STROKEAHA.107.482471>

Anor, Grinberg, Anor, T., Grinberg, L., Baek, H., Madsen, J. R., ... Karniadakis, G. E. (2010). Modeling of blood flow in arterial trees. *Wiley Interdisciplinary Reviews: Systems Biology and Medicine*, 2(5), 612–623. <http://doi.org/10.1002/wsbm.90>

Antiga, L., & Steinman, D. A. (2004). Robust and objective decomposition and mapping of bifurcating vessels. *IEEE Transactions on Medical Imaging*, 23(6), 704–13. <http://doi.org/10.1109/TMI.2004.826946>

Antiga, L., Piccinelli, M., Botti, L., Ene-Iordache, B., Remuzzi, A., & Steinman, D. A. (2008). An image-based modeling framework for patient-specific computational hemodynamics. *Medical & Biological Engineering & Computing*, 46(11), 1097–112. <http://doi.org/10.1007/s11517-008-0420-1>

Artmann, G. M., & Chien, S. (Eds.). (2008). *Bioengineering in Cell and Tissue Research*. Berlin, Heidelberg: Springer Berlin Heidelberg.

<http://doi.org/10.1007/978-3-540-75409-1>

Autodesk. (2011). 3ds Max.

Avman, N., & Bering, E. A. (1961). A Plastic Model for the Study of Pressure Changes in the Circle of Willis and Major Cerebral Arteries Following Arterial Occlusion. *Journal of Neurosurgery*, 18(3), 361–365.

<http://doi.org/10.3171/jns.1961.18.3.0361>

Baaijens, J. P. W. (1993). Numerical analysis of steady generalized Newtonian blood flow in a 2D model of the carotid artery bifurcation. *Biorheology*, 30(1), 63–74. Retrieved from

<http://www.ncbi.nlm.nih.gov/pubmed/8374103><http://alexandria.tue.nl/repository/freearticles/605420.pdf>

Banahan, C., Hague, J. P., Evans, D. H., Patel, R., Ramnarine, K. V., & Chung, E. M. L. (2012). Sizing gaseous emboli using Doppler embolic signal intensity. *Ultrasound in Medicine & Biology*, 38(5), 824–33.

<http://doi.org/10.1016/j.ultrasmedbio.2012.01.008>

Barak, M., & Katz, Y. (2005). Microbubbles: pathophysiology and clinical implications. *Chest*, 128(4), 2918–32. <http://doi.org/10.1378/chest.128.4.2918>

Bernabeu, M. O., Nash, R. W., Groen, D., Carver, H. B., Hetherington, J., Kruger, T., & Coveney, P. V. (2013). Impact of blood rheology on wall shear stress in a model of the middle cerebral artery. *Interface Focus*, 3(2), 20120094.

<http://doi.org/10.1098/rsfs.2012.0094>

Baumbach, A., Mullen, M., Brickman, A. M., Aggarwal, S. K., Pietras, C. G., Forrest, J. K., ... Lansky, A. J. (2015). Safety and performance of a novel embolic deflection device in patients undergoing transcatheter aortic valve replacement: results from the DEFLECT I study. *EuroIntervention*, 11(1), 75–84.

[http://doi.org/10.4244/EIJY15M04\\_01](http://doi.org/10.4244/EIJY15M04_01)

Carr, I. a, Nemoto, N., Schwartz, R. S., & Shadden, S. C. (2013). Size-dependent predilections of cardiogenic embolic transport. *AJP: Heart and Circulatory Physiology*, 305(5), H732–H739. <http://doi.org/10.1152/ajpheart.00320.2013>

Cassot, F., Zagzoule, M., & Marc-Vergnes, J. P. (2000). Hemodynamic role of the circle of Willis in stenoses of internal carotid arteries. An analytical solution

of a linear model. *Journal of Biomechanics*, 33(4), 395–405.

[http://doi.org/10.1016/S0021-9290\(99\)00193-1](http://doi.org/10.1016/S0021-9290(99)00193-1)

Cebral, J. R., Castro, M. a., Soto, O., Löhner, R., & Alperin, N. (2003). Blood-flow models of the circle of Willis from magnetic resonance data. *Journal of Engineering Mathematics*, 47(3/4), 369–386.

<http://doi.org/10.1023/B:ENGI.0000007977.02652.02>

Cenova, I., Kauzlarić, D., Greiner, A., & Korvink, J. G. (2011). Constrained simulations of flow in haemodynamic devices: towards a computational assistance of magnetic resonance imaging measurements. *Philosophical Transactions. Series A, Mathematical, Physical, and Engineering Sciences*, 369(1945), 2494–501. <http://doi.org/10.1098/rsta.2011.0028>

Chatziprodromou, I., Tricoli, a, Poulidakos, D., & Ventikos, Y. (2007). Haemodynamics and wall remodelling of a growing cerebral aneurysm: a computational model. *Journal of Biomechanics*, 40(2), 412–26.

<http://doi.org/10.1016/j.jbiomech.2005.12.009>

Chaudhuri, K., & Marasco, S. F. (2011). The effect of carbon dioxide insufflation on cognitive function during cardiac surgery. *Journal of Cardiac Surgery*, 26(2), 189–96. <http://doi.org/10.1111/j.1540-8191.2011.01217.x>

Chaudhuri, K., Storey, E., Lee, G. A., Bailey, M., Chan, J., Rosenfeldt, F. L., ... Marasco, S. F. (2012). Carbon dioxide insufflation in open-chamber cardiac surgery: a double-blind, randomized clinical trial of neurocognitive effects. *The Journal of Thoracic and Cardiovascular Surgery*, 144(3), 646–653.e1.

<http://doi.org/10.1016/j.jtcvs.2012.04.010>

Chen, C. H., Nevo, E., Fetics, B., Pak, P. H., Yin, F. C., Maughan, W. L., & Kass, D. A. (1997). Estimation of central aortic pressure waveform by mathematical transformation of radial tonometry pressure. Validation of generalized transfer function. *Circulation*, 95(7), 1827–36. Retrieved from

<http://www.ncbi.nlm.nih.gov/pubmed/9107170>

Chui, Y.-P., & Heng, P.-A. (2010). A meshless rheological model for blood-vessel interaction in endovascular simulation. *Progress in Biophysics and Molecular Biology*, 103(2–3), 252–61.

<http://doi.org/10.1016/j.pbiomolbio.2010.09.003>

Chung, E. M. L., Hague, J. P., & Evans, D. H. (2007). Revealing the mechanisms underlying embolic stroke using computational modelling. *Physics in Medicine and Biology*, 52(23), 7153–66. <http://doi.org/10.1088/0031-9155/52/23/025>

Chung, E. M. L., Hague, J. P., Chanrion, M.-A., Ramnarine, K. V, Katsogridakis, E., & Evans, D. H. (2010). Embolus trajectory through a physical replica of the major cerebral arteries. *Stroke*, 41(4), 647–52.  
<http://doi.org/10.1161/STROKEAHA.109.574400>

Cieslicki, K., & Ciesla, D. (2005). Investigations of flow and pressure distributions in physical model of the circle of Willis. *Journal of Biomechanics*, 38(11), 2302–2310. <http://doi.org/10.1016/j.jbiomech.2004.07.038>

Courant, R., Friedrichs, K., & Lewy, H. (1928). Über die partiellen Differenzgleichungen der mathematischen Physik. *Mathematische Annalen*, 100(1), 32–74. <http://doi.org/10.1007/BF01448839>

Dalrymple, R. A., & Rogers, B. D. (2006). Numerical modeling of water waves with the SPH method. *Coastal Engineering*, 53(2–3), 141–147.  
<http://doi.org/10.1016/j.coastaleng.2005.10.004>

David, T., Brown, M., & Ferrandez, A. (2003). Auto-regulation and blood flow in the cerebral circulation. *International Journal for Numerical Methods in Fluids*, 43(6–7), 701–713. <http://doi.org/10.1002/flf.495>

Fabbri, D., Long, Q., Das, S., & Pinelli, M. (2014). Computational modelling of emboli travel trajectories in cerebral arteries: influence of microembolic particle size and density. *Biomechanics and Modeling in Mechanobiology*, 13(2), 289–302. <http://doi.org/10.1007/s10237-014-0561-0>

Farahani, M. H., Amanifard, N., & Pouryoussefi, G. (2008). Numerical Simulation of a Pulsatory Flow Moving Through Flexible Walls Using Smoothed Particle Hydrodynamics. In *Proceedings of the World Congress on Engineering 2008 Vol II*.

Fasano, V. A., Portalupi, A., & Broggi, G. (1966). Study of cerebral hemodynamics by analogic models. *Vascular Diseases*, 3(2), 89–99. Retrieved from <http://www.ncbi.nlm.nih.gov/pubmed/5933763>

Fedorov, A., Beichel, R., Kalpathy-Cramer, J., Finet, J., Fillion-Robin, J.-C., Pujol, S., ... Kikinis, R. (2012). 3D Slicer as an image computing platform for the

Quantitative Imaging Network. *Magnetic Resonance Imaging*, 30(9), 1323–1341.  
<http://doi.org/10.1016/j.mri.2012.05.001>

Ferrandez, a, David, T., & Brown, M. D. (2002). Numerical models of auto-regulation and blood flow in the cerebral circulation. *Computer Methods in Biomechanics and Biomedical Engineering*, 5(1), 7–19.  
<http://doi.org/10.1080/10255840290032171>

Galassi, M., Davies, J., Theiler, J., Gough, B., Jungman, G., Alken, P., ... Rossi, F. (2009). *GNU Scientific Library Reference Manual* (3rd ed.). Network Theory Ltd. Retrieved from <http://www.gnu.org/software/gsl/>

Gerriets, T., Schwarz, N., Sammer, G., Baehr, J., Stolz, E., Kaps, M., ... Schönburg, M. (2010). Protecting the brain from gaseous and solid micro-emboli during coronary artery bypass grafting: a randomized controlled trial. *European Heart Journal*, 31(3), 360–8. <http://doi.org/10.1093/eurheartj/ehp178>

Gholami, B., Comerford, A., & Ellero, M. (2014). A multiscale SPH particle model of the near-wall dynamics of leukocytes in flow. *International Journal for Numerical Methods in Biomedical Engineering*, 30(1), 83–102.  
<http://doi.org/10.1002/cnm.2591>

Gizzi, A., Bernaschi, M., Bini, D., Cherubini, C., Filippi, S., Melchionna, S., & Succi, S. (2011). Three-band decomposition analysis of wall shear stress in pulsatile flows. *Physical Review E*, 83(3), 31902.  
<http://doi.org/10.1103/PhysRevE.83.031902>

Gomez-Gesteira, M., Crespo, A. J. C., Rogers, B. D., Dalrymple, R. a., Dominguez, J. M., & Barreiro, A. (2012a). SPHysics - development of a free-surface fluid solver - Part 2: Efficiency and test cases. *Computers & Geosciences*, 48, 300–307. <http://doi.org/10.1016/j.cageo.2012.02.028>

Gomez-Gesteira, M., Rogers, B. D., Crespo, A. J. C., Dalrymple, R. a., Narayanaswamy, M., & Dominguez, J. M. (2012b). SPHysics - development of a free-surface fluid solver - Part 1: Theory and formulations. *Computers & Geosciences*, 48, 289–299. <http://doi.org/10.1016/j.cageo.2012.02.029>

Grinberg, L., Anor, T., Madsen, J. R., Yakhov, a, & Karniadakis, G. E. (2009). Large-scale simulation of the human arterial tree. *Clinical and Experimental Pharmacology & Physiology*, 36(2), 194–205. <http://doi.org/10.1111/j.1440-1681.2008.05010.x>

- Grinberg, L. (2012). Proper orthogonal decomposition of atomistic flow simulations. *Journal of Computational Physics*, 231(16), 5542–5556.  
<http://doi.org/10.1016/j.jcp.2012.05.007>
- Grinberg, L., Fedosov, D. A., & Karniadakis, G. E. (2013). Parallel multiscale simulations of a brain aneurysm. *Journal of Computational Physics*, 244, 131–147. <http://doi.org/10.1016/j.jcp.2012.08.023>
- Hague, J., & Chung, E. (2009). Statistical physics of cerebral embolization leading to stroke. *Physical Review E*, 80(5), 1–9.  
<http://doi.org/10.1103/PhysRevE.80.051912>
- Hague, J. P., Banahan, C., & Chung, E. M. L. (2013). Modelling of impaired cerebral blood flow due to gaseous emboli. *Physics in Medicine and Biology*, 58(13), 4381–4394. <http://doi.org/10.1088/0031-9155/58/13/4381>
- Haines, D. E., Stewart, M. T., Barka, N. D., Kirchhof, N., Lentz, L. R., Reinking, N. M., ... Kanal, E. (2013). Microembolism and catheter ablation II: effects of cerebral microemboli injection in a canine model. *Circulation. Arrhythmia and Electrophysiology*, 6(1), 23–30. <http://doi.org/10.1161/CIRCEP.112.973461>
- Hillen, B., Drinkenburg, B. A. H., Hoogstraten, H. W., & Post, L. (1988). Analysis of flow and vascular resistance in a model of the cricle of Willis. *Journal of Biomechanics*, 21(10), 807–814. [http://doi.org/10.1016/0021-9290\(88\)90013-9](http://doi.org/10.1016/0021-9290(88)90013-9)
- Himwich, W. A., & Clark, M. E. (1971). Cerebral blood flow comparisons between model and prototype. *Journal of Applied Physiology*, 31(6), 873–9. Retrieved from <http://www.ncbi.nlm.nih.gov/pubmed/5123667>
- Hosseini, S. M., & Feng, J. J. (2009). A particle-based model for the transport of erythrocytes in capillaries. *Chemical Engineering Science*, 64(22), 4488–4497. <http://doi.org/10.1016/j.ces.2008.11.028>
- Hosseini, S. M., & Feng, J. J. (2012). How malaria parasites reduce the deformability of infected red blood cells. *Biophysical Journal*, 103(1), 1–10. <http://doi.org/10.1016/j.bpj.2012.05.026>
- Hu, Y., Sui, J., Yang, L., & Hua, Z. (2009). A Nonlinear Model for the Circle of Willis Based on Hemodynamics. 2009 3rd International Conference on Bioinformatics and Biomedical Engineering, 1(2), 1–4. <http://doi.org/10.1109/ICBBE.2009.5162208>

- Jou, L.-D., Lee, D. H., & Mawad, M. E. (2010). Cross-flow at the anterior communicating artery and its implication in cerebral aneurysm formation. *Journal of Biomechanics*, 43(11), 2189–95.  
<http://doi.org/10.1016/j.jbiomech.2010.03.039>
- Karch, R., Neumann, F., Neumann, M., & Schreiner, W. (2000). Staged Growth of Optimized Arterial Model Trees. *Annals of Biomedical Engineering*, 28(5), 495–511. <http://doi.org/10.1114/1.290>
- Kassab, G. S., Rider, C. a, Tang, N. J., & Fung, Y. C. (1993). Morphometry of pig coronary arterial trees. *The American Journal of Physiology*, 265(1 Pt 2), H350-65. Retrieved from <http://www.ncbi.nlm.nih.gov/pubmed/8342652>
- Kazhdan, M., & Hoppe, H. (2013). Screened poisson surface reconstruction. *ACM Transactions on Graphics*, 32(3), 1–13.  
<http://doi.org/10.1145/2487228.2487237>
- Keelan, J., Chung, E. M. L., & Hague, J. P. (2016). Simulated annealing approach to vascular structure with application to the coronary arteries. *Royal Society Open Science*, 3(2), 150431. <http://doi.org/10.1098/rsos.150431>
- Kerwin, T., Shen, H.-W., & Stredney, D. (2009). Enhancing realism of wet surfaces in temporal bone surgical simulation. *IEEE Transactions on Visualization and Computer Graphics*, 15(5), 747–58.  
<http://doi.org/10.1109/TVCG.2009.31>
- Khodae, F., Vahidi, B., & Fatourae, N. (2016). Analysis of mechanical parameters on the thromboembolism using a patient-specific computational model. *Biomechanics and Modeling in Mechanobiology*, 15(5), 1–11.  
<http://doi.org/10.1007/s10237-016-0762-9>
- Krabbe-Hartkamp, M. J., van der Grond, J., de Leeuw, F. E., de Groot, J. C., Algra, A., Hillen, B., ... Mali, W. P. (1998). Circle of Willis: morphologic variation on three-dimensional time-of-flight MR angiograms. *Radiology*, 207(1), 103–11. Retrieved from <http://radiology.rsna.org/content/207/1/103.short>
- Kruis, R. W. J., Vlasveld, F. A. E., & Van Dijk, D. (2010). The (un)importance of cerebral microemboli. *Seminars in Cardiothoracic and Vascular Anesthesia*, 14(2), 111–8. <http://doi.org/10.1177/1089253210370903>

- Lynch, J. E., & Riley, J. B. (2008). Microemboli detection on extracorporeal bypass circuits. *Perfusion*, 23(1), 23–32.  
<http://doi.org/10.1177/0267659108094350>
- Liang, C. X., Zhang, C. Y., Liu, H. Y., Qin, Z. R., & Wang, Q. (2012). Modeling low Reynolds number incompressible flows with curved boundaries using SPH. *International Journal for Numerical Methods in Fluids*, 68(9), 1173–1188.  
<http://doi.org/10.1002/fld.2600>
- Lippert, H., & Pabst, R. (1985). Arterial variations in man: classification and frequency. J.F. Bergmann Verlag. Retrieved from  
[https://books.google.co.uk/books/about/Arterial\\_variations\\_in\\_man.html?id=SbxqAAAAMAAJ&redir\\_esc=y](https://books.google.co.uk/books/about/Arterial_variations_in_man.html?id=SbxqAAAAMAAJ&redir_esc=y)
- Liu, W., Sewell, C., Blevins, N., Salisbury, K., Bodin, K., & Hjelte, N. (2008). Representing fluid with smoothed particle hydrodynamics in a cranial base simulator. In *Studies in Health Technology and Informatics* (Vol. 132, pp. 257–9). Retrieved from <http://www.ncbi.nlm.nih.gov/pubmed/18391299>
- Matas, J.-P., Morris, J. F., & Guazzelli, É. (2004). Inertial migration of rigid spherical particles in Poiseuille flow. *Journal of Fluid Mechanics*, 515, 171–195.  
<http://doi.org/10.1017/S0022112004000254>
- Matthys, K. S., Alastruey, J., Peiró, J., Khir, A. W., Segers, P., Verdonck, P. R., ... Sherwin, S. J. (2007). Pulse wave propagation in a model human arterial network: assessment of 1-D numerical simulations against in vitro measurements. *Journal of Biomechanics*, 40(15), 3476–86.  
<http://doi.org/10.1016/j.jbiomech.2007.05.027>
- Mitchell, S., & Gorman, D. (2002). The pathophysiology of cerebral arterial gas embolism. *The Journal of Extra-Corporeal Technology*, 34(1), 18–23. Retrieved from <http://www.ncbi.nlm.nih.gov/pubmed/11911624>
- Moehring, M. a, & Klepper, J. R. (1994). Pulse Doppler ultrasound detection, characterization and size estimation of emboli in flowing blood. *IEEE Transactions on Bio-Medical Engineering*, 41(1), 35–44.  
<http://doi.org/10.1109/10.277269>
- Moore, S. M., Moorhead, K. T., Chase, J. G., David, T., & Fink, J. (2005). One-Dimensional and Three-Dimensional Models of Cerebrovascular Flow. *Journal of Biomechanical Engineering*, 127(3), 440. <http://doi.org/10.1115/1.1894350>

Moore, S., David, T., Chase, J. G., Arnold, J., & Fink, J. (2006). 3D models of blood flow in the cerebral vasculature. *Journal of Biomechanics*, 39(8), 1454–63. <http://doi.org/10.1016/j.jbiomech.2005.04.005>

Moorhead, K. T., Doran, C. V., Chase, J. G., & David, T. (2004). Lumped parameter and feedback control models of the auto-regulatory response in the Circle of Willis. *Computer Methods in Biomechanics and Biomedical Engineering*, 7(3), 121–30. <http://doi.org/10.1080/10255840410001710894>

Monaghan, J. (1992). Smoothed Particle Hydrodynamics. *Annual Review of Astronomy and Astrophysics*, 30(1), 543–574. <http://doi.org/10.1146/annurev.astro.30.1.543>

Monaghan, J. J. (1994). Simulating Free Surface Flows with SPH. *Journal of Computational Physics*, 110(2), 399–406. <http://doi.org/10.1006/jcph.1994.1034>

Monaghan, J. J., & Kos, A. (1999). Solitary Waves on a Cretan Beach. *Journal of Waterway, Port, Coastal, and Ocean Engineering*, 125(3), 145–155. [http://doi.org/10.1061/\(ASCE\)0733-950X\(1999\)125:3\(145\)](http://doi.org/10.1061/(ASCE)0733-950X(1999)125:3(145))

Monaghan, J. J. (2005). Smoothed particle hydrodynamics. *Reports on Progress in Physics*, 68(8), 1703–1759. <http://doi.org/10.1088/0034-4885/68/8/R01>

Monaghan, J. J. (2012). Smoothed Particle Hydrodynamics and Its Diverse Applications. *Annual Review of Fluid Mechanics*, 44(1), 323–346. <http://doi.org/10.1146/annurev-fluid-120710-101220>

Moody, D. M., Brown, W. R., Challa, V. R., Stump, D. A., Reboussin, D. M., & Legault, C. (1995). Brain microemboli associated with cardiopulmonary bypass: a histologic and magnetic resonance imaging study. *The Annals of Thoracic Surgery*, 59(5), 1304–7. Retrieved from <http://www.ncbi.nlm.nih.gov/pubmed/7733757>

Moorhead, K. T., Doran, C. V., Chase, J. G., & David, T. (2004). Lumped parameter and feedback control models of the auto-regulatory response in the Circle of Willis. *Computer Methods in Biomechanics and Biomedical Engineering*, 7(3), 121–30. <http://doi.org/10.1080/10255840410001710894>

Morris, J. P., Fox, P. J., & Zhu, Y. (1997). Modeling Low Reynolds Number Incompressible Flows Using SPH. *Journal of Computational Physics*, 136(1), 214–226. <http://doi.org/10.1006/jcph.1997.5776>

Mukherjee, D., Padilla, J., & Shadden, S. C. (2016 (1)). Numerical investigation of fluid–particle interactions for embolic stroke. *Theoretical and Computational Fluid Dynamics*, 30(1–2), 23–39. <http://doi.org/10.1007/s00162-015-0359-4>

Mukherjee, D., Jani, N. D., Selvaganesan, K., Weng, C. L., & Shadden, S. C. (2016 (2)). Computational Assessment of the Relation Between Embolism Source and Embolus Distribution to the Circle of Willis for Improved Understanding of Stroke Etiology. *Journal of Biomechanical Engineering*, 138(8), 81008. <http://doi.org/10.1115/1.4033986>

Mulder, G., Bogaerds, a C. B., Rongen, P., & van de Vosse, F. N. (2011). The influence of contrast agent injection on physiological flow in the circle of Willis. *Medical Engineering & Physics*, 33(2), 195–203. <http://doi.org/10.1016/j.medengphy.2010.09.021>

Müller, M., Schirm, S., & Teschner, M. (2004). Interactive blood simulation for virtual surgery based on smoothed particle hydrodynamics. *Technology and Health Care: Official Journal of the European Society for Engineering and Medicine*, 12(1), 25–31. Retrieved from <http://www.ncbi.nlm.nih.gov/pubmed/15096684>

Murray, C. D. (1926). The Physiological Principle of Minimum Work: I. The Vascular System and the Cost of Blood Volume. *Proceedings of the National Academy of Sciences*, 12(3), 207–214. <http://doi.org/10.1073/pnas.12.3.207>

Murray, K. D. (1964). Dimensions of the Circle of Willis and Dynamic Studies Using Electrical Analogy. *Journal of Neurosurgery*, 21(1), 26–34. <http://doi.org/10.3171/jns.1964.21.1.0026>

Nietlispach, F., Wijesinghe, N., Gurvitch, R., Tay, E., Carpenter, J. P., Burns, C., ... Webb, J. G. (2010). An embolic deflection device for aortic valve interventions. *JACC: Cardiovascular Interventions*, 3(11), 1133–8. <http://doi.org/10.1016/j.jcin.2010.05.022>

Orosz, L., Hoksbergen, A. W., Molnár, C., Siró, P., Cassot, F., Marc-Vergnes, J.-P., & Fülesdi, B. (2009). Clinical applicability of a mathematical model in assessing the functional ability of the communicating arteries of the circle of Willis. *Journal of the Neurological Sciences*, 287(1–2), 94–9. <http://doi.org/10.1016/j.jns.2009.08.049>

- Oshima, M. (2004). A new approach to cerebral hemodynamics. Patient-specific modelling and numerical simulation of blood and arterial wall interaction. *IACM Expressions*, 16, 4–9.
- Patel, N., Horsfield, M. A., Banahan, C., Janus, J., Masters, K., Morlese, J., ... Chung, E. M. L. (2015). Impact of perioperative infarcts after cardiac surgery. *Stroke* (Vol. 46). <http://doi.org/10.1161/STROKEAHA.114.007533>
- Pazouki, A., & Negrut, D. (2015). A numerical study of the effect of particle properties on the radial distribution of suspensions in pipe flow. *Computers & Fluids*, 108, 1–12. <http://doi.org/10.1016/j.compfluid.2014.11.027>
- Price, D. J. (2012). Smoothed particle hydrodynamics and magnetohydrodynamics. *Journal of Computational Physics*, 231(3), 759–794. <http://doi.org/10.1016/j.jcp.2010.12.011>
- Raines, J. K., Jaffrin, M. Y., & Shapiro, A. H. (1974). A computer simulation of arterial dynamics in the human leg. *Journal of Biomechanics*, 7(1), 77–91. [http://doi.org/10.1016/0021-9290\(74\)90072-4](http://doi.org/10.1016/0021-9290(74)90072-4)
- Ringelstein, E. B., Droste, D. W., Babikian, V. L., Evans, D. H., Grosset, D. G., Kaps, M., ... Siebler, M. (1998). Consensus on microembolus detection by TCD. International Consensus Group on Microembolus Detection. *Stroke*, 29(3), 725–9. Retrieved from <http://www.ncbi.nlm.nih.gov/pubmed/9506619>
- Rogers, L. (1947). The function of the circulus arteriosus of Willis. *Brain*, 70(Pt 2), 171–8. Retrieved from <http://www.ncbi.nlm.nih.gov/pubmed/20261819>
- Schaaf, B. W., & Abbrecht, P. H. (1972). Digital computer simulation of human systemic arterial pulse wave transmission: A nonlinear model. *Journal of Biomechanics*, 5(4), 345–364. [http://doi.org/10.1016/0021-9290\(72\)90064-4](http://doi.org/10.1016/0021-9290(72)90064-4)
- Scheel, P., Ruge, C., Petruch, U. R., & Schoning, M. (2000). Color Duplex Measurement of Cerebral Blood Flow Volume in Healthy Adults. *Stroke*, 31(1), 147–150. <http://doi.org/10.1161/01.STR.31.1.147>
- Schomer, D. F., Marks, M. P., Steinberg, G. K., Johnstone, I. M., Boothroyd, D. B., Ross, M. R., ... Enzmann, D. R. (1994). The anatomy of the posterior communicating artery as a risk factor for ischemic cerebral infarction. *The New England Journal of Medicine*, 330(22), 1565–1570.

Segré, G., & Silberberg, A. (1962). Behaviour of macroscopic rigid spheres in Poiseuille flow, Parts 1 and 2. *Journal of Fluid Mechanics*, 14(1), 115–157.

<http://doi.org/10.1017/S0022112062001111>

Shahriari, S. (2011). *Computational Modeling of Cardiovascular Flows using Smoothed Particle Hydrodynamics*. Concordia University.

Shahriari, S., Kadem, L., Rogers, B. D., & Hassan, I. (2012a). Smoothed particle hydrodynamics method applied to pulsatile flow inside a rigid two-dimensional model of left heart cavity. *International Journal for Numerical Methods in Biomedical Engineering*, 28(11), 1121–43. Retrieved from

<http://www.ncbi.nlm.nih.gov/pubmed/23109382>

Shahriari, S., Maleki, H., Hassan, I., & Kadem, L. (2012b). Evaluation of shear stress accumulation on blood components in normal and dysfunctional bileaflet mechanical heart valves using smoothed particle hydrodynamics. *Journal of Biomechanics*, 45(15), 2637–44. <http://doi.org/10.1016/j.jbiomech.2012.08.009>

Shahriari, S., Hassan, I. G., & Kadem, L. (2013). Modeling unsteady flow characteristics using smoothed particle hydrodynamics. *Applied Mathematical Modelling*, 37(3), 1431–1450. <http://doi.org/10.1016/j.apm.2012.04.017>

Sherman, T. F. (1981). On connecting large vessels to small. The meaning of Murray's law. *The Journal of General Physiology*, 78(4), 431–53. Retrieved from <http://www.uvm.edu/pdodds/files/papers/others/1981/sherman1981a.pdf>

Sherwin, S. J., Formaggia, L., Peiró, J., & Franke, V. (2003a). Computational modelling of 1D blood flow with variable mechanical properties and its application to the simulation of wave propagation in the human arterial system. *International Journal for Numerical Methods in Fluids*, 43(6–7), 673–700. <http://doi.org/10.1002/flid.543>

Sherwin, S. J., Franke, V., Peiró, J., & Parker, K. (2003b). One-dimensional modelling of a vascular network in space-time variables. *Journal of Engineering Mathematics*, 47(3/4), 217–250.

<http://doi.org/10.1023/B:ENGI.0000007979.32871.e2>

Shi, Y., Lawford, P., & Hose, R. (2011). Review of zero-D and 1-D models of blood flow in the cardiovascular system. *Biomedical Engineering Online*, 10(1), 33. article. <http://doi.org/10.1186/1475-925X-10-33>

Simvascular, 2016, "SimVascular," Stanford University, Stanford, CA  
<http://simvascular.github.io>

Sinnott, M., Cleary, P. W., & Prakash, M. (2006). An investigation of pulsatile blood flow in a bifurcation artery using a grid-free method. In Proceedings of the Fifth International Conference on CFD in the Process Industries. Melbourne, Australia.

Sinnott, M. D., & Cleary, P. W. (2010). Effect of rotor blade angle and clearance on blood flow through a non-pulsatile, axial, heart pump. *Progress in Computational Fluid Dynamics, An International Journal*, 10(5/6), 300.  
<http://doi.org/10.1504/PCFD.2010.035363>

Sinnott, M. D., Cleary, P. W., Arkwright, J. W., & Dinning, P. G. (2012). Investigating the relationships between peristaltic contraction and fluid transport in the human colon using Smoothed Particle Hydrodynamics. *Computers in Biology and Medicine*, 42(4), 492–503.  
<http://doi.org/10.1016/j.compbiomed.2012.01.002>

Šutalo, I. D., Bui, a., Ahmed, S., Liffman, K., & Manasseh, R. (2009). Modelling of flow through the circle of Willis and cerebral vasculature, 13, 83–92.  
<http://doi.org/10.2495/BIO090081>

Tanaka, N., & Takano, T. (2005). Microscopic-scale simulation of blood flow using SPH method. *International Journal of Computational Methods*, 2(4), 555–568. <http://doi.org/10.1142/S021987620500065X>

Taylor, C. a, & Figueroa, C. a. (2009). Patient-specific modeling of cardiovascular mechanics. *Annual Review of Biomedical Engineering*, 11, 109–34. <http://doi.org/10.1146/annurev.bioeng.10.061807.160521>

Tian, F. B., Zhu, L., Fok, P. W., & Lu, X. Y. (2013). Simulation of a pulsatile non-Newtonian flow past a stenosed 2D artery with atherosclerosis. *Computers in Biology and Medicine*, 43(9), 1098–1113.  
<http://doi.org/10.1016/j.compbiomed.2013.05.023>

Vahidi, B., & Fatourae, N. (2012). Large deforming buoyant embolus passing through a stenotic common carotid artery: a computational simulation. *Journal of Biomechanics*, 45(7), 1312–22. <http://doi.org/10.1016/j.jbiomech.2012.01.020>

Viedma, A., Jiménez-Ortiz, C., & Marco, V. (1997). Extended Willis circle model to explain clinical observations in periorbital arterial flow. *Journal of Biomechanics*, 30(3), 265–272. [http://doi.org/10.1016/S0021-9290\(96\)00143-1](http://doi.org/10.1016/S0021-9290(96)00143-1)

Violeau, D. (2012). *Fluid Mechanics and the SPH method: theory and applications* (1st ed.). Oxford: Oxford University Press.

Visual Computing Lab ISTI - CNR. (2012). MeshLab. Retrieved from <http://meshlab.sourceforge.net/>

Westerhof, N., Bosman, F., De Vries, C. J., & Noordergraaf, A. (1969). Analog studies of the human systemic arterial tree. *Journal of Biomechanics*, 2(2), 121–143. [http://doi.org/10.1016/0021-9290\(69\)90024-4](http://doi.org/10.1016/0021-9290(69)90024-4)

Xinapse Systems. (n.d.). Jim. Retrieved from <http://www.xinapse.com/>

Young, P. A., Young, P. H. (Paul H., Tolbert, D. L. (Daniel L., & Young, P. A. (2008). *Basic clinical neuroscience* (2nd ed.). Maryland: Wolters Kluwer Health/Lippincott Williams & Wilkins.

Zamir, M. (1999). On fractal properties of arterial trees. *Journal of Theoretical Biology*, 197(4), 517–26. <http://doi.org/10.1006/jtbi.1998.0892>