Efficient Ensemble Simulation Methods for In-Silico Trials of Endovascular Medical Devices

Michael MacRaild

The University of Leeds

Centre for Doctoral Training in Fluid Dynamics
School of Computing

Submitted in accordance with the requirements for the degree of

Doctor of Philosophy

June 2024

Intellectual Property Statement

I confirm that the work submitted is my own, except where work which has formed part of jointly authored publications has been included. My contribution and the other authors' to this work has been explicitly indicated below. I confirm that appropriate credit has been given within the thesis where reference has been made to the work of others.

Work from this thesis will appear in three publications. Chapters 2 and 3 include work from the following publication: *MacRaild, M., Sarrami-Foroushani, A., Lassila, T. and Frangi, A.F., 2024. Accelerated simulation methodologies for computational vascular flow modelling. Journal of the Royal Society Interface, 21(211), p.20230565.* I performed the literature review, wrote the original manuscript draft and revised the manuscript. The other authors contributed to conceptualisation, supervision and review and editing of the manuscript.

Chapter 4 includes work from the following publication that is under review: *MacRaild, M., Sarrami-Foroushani, A., Song, S., Liu, Q., Ravikumar, N., Kelly, C., Patankar T., Taylor, Z.A., Lassila, T., Frangi, A.F., 2023. Off-label in-silico flow diverter performance assessment in posterior communicating artery aneurysms. Journal of Neurointerventional Surgery.* I contributed to the in-silico trial design, developed boundary conditions specific to the in-silico trial, processed the majority of the data (i.e. performed surface mesh corrections, device deployment, volume meshing, simulation set-up), executed and post-processed the simulations, analysed the results, wrote the original manuscript draft and revised the manuscript. A. Sarrami-Foroushani contributed to conceptualisation, methodology, software, and review and editing of the manuscript. S. Song, Q. Liu, C. Kelly and N. Ravikumar contributed to methodology and software. T. Patankar,

T. Lassila, Z.A. Taylor and A.F. Frangi contributed to conceptualisation and review and editing of the manuscript.

Chapter 5 includes work from the following publication that is under review: *MacRaild, M., Sarrami-Foroushani, A., Lassila, T., Frangi, A.F., 2024. Reduced order modelling of intracranial aneurysm flow using proper orthogonal decomposition and neural networks. International Journal for Numerical Methods in Biomedical Engineering, submitted January 2024.* I prepared, performed and post-processed simulations, wrote all code for the reduced order model, trained and tested the reduced order model, analysed the results, wrote the original manuscript draft, and revised the manuscript. The other authors contributed to conceptualisation, supervision and review and editing of the manuscript.

Acknowledgements

To my supervisory team: Alex, Toni and Ali. Thank you for all of your support and guidance. You have shaped me into the researcher I am today.

To my other collaborators and friends from our research group: Zeike, Nishant, Chris, Shuang, Fergus, Qiongyao, Ben, Fengming and Ning. Thank you for your shared passion and for making my PhD experience more enjoyable for it.

I am grateful to the Centre for Doctoral Training in Fluid Dynamics for giving me the opportunity to undertake this PhD and I am thankful for the friendships I have made through this: Greg, Imran, Jen, Joe, Molly, Rose.

To my dear friends Isabella, Jack, Jamie, Luke, Sam, Evie, Hannah and Rose, you have all supported me throughout the PhD and given me treasured memories outside of work.

To Dad, your belief in me gives me determination; to Mum, your selflessness inspires me; to my younger siblings, Katie and Andrew, you make sure that I don't take myself too seriously; and to Nanny Shirley, Papa, Nanny Wendy and Auntie Peg, you always show interest and excitement for all that I do. Thank you to you all.

Lastly and most importantly, Pascale. Your encouragement and support have been constant throughout my PhD from back when I was preparing for my first PhD interview to now as I prepare to submit my thesis. Your belief in me has never wavered and I am incredibly grateful to have you by my side.

Abstract

In-silico trials (ISTs) use computational modelling and simulation in virtual patients to evaluate medical device performance. Despite early promise, various challenges prevent the use of ISTs from becoming common practice in medical device development. Three significant challenges are: (i) Prohibitive costs due to complex computational models that require excessive resources and time to execute. (ii) Lack of exemplar ISTs demonstrating their effectiveness in generating evidence for medical device performance. (iii) Lack of scalability and reproducibility due to computational modelling pipelines requiring significant expert manual input.

In this thesis, challenge (i) was addressed through a comprehensive literature review into reduced order modelling and machine learning techniques that can accelerate the computational models that are essential in ISTs. Challenge (ii) was addressed by performing the FD-PComA IST into flow diversion (FD) of posterior communicating artery (PComA) aneurysms, which are a common sub-group currently not approved for treatment with the most widely used flow diverter. PComA aneurysm treatment is complicated by the presence of fetal posterior circulation (FPC), which has an estimated prevalence of 4–29% and is more common in black than white ethnicities. Given these factors, FD-PComA demonstrates the effectiveness of ISTs in generating evidence for less-studied scenarios and demographics. Challenge (iii) was addressed in FD-PComA through automation of the modelling steps. The results of FD-PComA demonstrate that flow diversion is less effective in FPC patients and that PComA and aneurysm morphology do not influence treatment performance. Challenge (i) was addressed further through the development of a machine learning reduced order model (ML-ROM) for evaluating aneurysm blood flow subject to physiological variation, which is a relevant problem for IST applications. The ML-ROM was approximately 98% accurate in evaluating the velocity solution and provided an acceleration of five orders of magnitude relative to a computational fluid dynamics model for the same problem.

Contents

LIST OF FIGURES

List of Tables

Abbreviations

ND, N-Dimensional, $N \in \{0, 1, 2, 3\}$

ACA, Anterior Cerebral Artery

AE, Autoencoder

AW, Area-weighted

CAE, Convolutional Autoencoder

CFD, Computational Fluid Dynamics

CNN, Convolutional Neural Network

CoW, Circle of Willis

cPINN, Conservative PINN

CPU, Central Processing Unit

CVRC, Continuously Variable Resonance Combustor

DeepONet, Deep Operator Network

DEIM, Discrete Empirical Interpolation Method

DL, Deep Learning

DMD, Dynamic Mode Decomposition

DMDc, DMD with Control

DMDho, High-order DMD

DPM, Dynamic Pulling Method

ECAP, Endothelial Cell Activation Potential

EDMD, Extended DMD

EIM, Empirical Interpolation Method

FCNN, Fully-Connected Neural Network

FDA, Food and Drug Administration

FDS, Flow-Diverting Stent

FEM, Finite Element Model

FFR, Fractional Flow Reserve

FOM, Full Order Model

FPC, Fetal Posterior Circulation

FSI, Fluid-Structure Interaction

GP, Galerkin Projection

GPU, Graphical Processing Unit

ICA, Internal Carotid Artery

IST, In-Silico Trial

LAA, Left Atrial Appendage

LSTM, Long Short-Term Memory

LVAD, Left Ventricular Assist Device

LH, Latin Hypercube

MCA, Middle Cerebral Artery

mDMD, Multi-stage DMD

mDMDc, Multi-stage DMD with Control

mrDMD, Multi-resolution DMD

ML, Machine Learning

MTAV, Maximum Time-Averaged Velocity

NN, Neural Network

NT, Normalised Time

OSI, Oscillatory Shear Index

PA, Pulmonary Artery

PCA, Principal Component Analysis

PCA, Posterior Cerebral Artery

PComA, Posterior Communicating Artery

PED, Pipeline Embolization Device

PDE, Partial Differential Equation

PGD, Proper Generalised Decomposition

PI, Pulsatility Index

PI-CNN, Physics-Informed CNN

PINN, Physics-Informed Neural Network

PIPN, Physics-Informed Point-Net

PPINN, Parallel-in-time PINN

POD, Proper Orthogonal Decomposition

POD-GP, POD with Galerkin Projection

POD-I, POD with Interpolation

POD-P, POD with Projection

RBF, Radial Basis Functions

RB, Reduced Basis

RN, Residual Network

RNN, Recurrent Neural Network

ROM, Reduced Order Model/Modelling

SBP, Systolic Blood Pressure

SDR, Spatial Dimension Reduction

SFH, Scale Factor for Heart Rate

SFP, Scale Factor for Pulsatility Index

SFQ, Scale Factor for Flow Magnitude

SN, Sequential Network

SR, Super-Resolution

SST, Sea Surface Temperature

STAV, Space-and-Time-Averaged Velocity

ToF, Tetralogy of Fallot

WSS, Wall Shear Stress

WCT, Wall Clock Time

XPINN, Extended PINN

Chapter 1

Introduction

1.1 In-Silico Trials of Medical Devices

Clinical trials have revolutionised healthcare by providing patients with a myriad of new treatments such as vaccines, drugs and medical devices. Despite this success, clinical trials are afflicted by numerous issues in their current state. These issues include the following:

1. Human clinical trials are the most expensive and time-consuming stage in the development of a medical device [5]. Using data collected from seven major biopharma companies across a five year period, Martin et al. [6] found that the median cost from protocol approval to final clinical trial report was \$3.4M USD for phase I trials, \$8.6M USD for phase II trials and \$21.4M USD for phase III trials.
2. Clinical trials are not always successful. Wong et al. [7] found the probability of success (i.e. likelihood of moving from phase 1 to approval) for all indications across over 40,000 pharmaceutical trials to be only 13.8%. While identifying treatments that will not work before they are approved is useful, failed trials cost money and take time which can delay the development of suitable treatments.
3. Long time-to-market. A survey of over 200 manufacturers found that a Food and Drug Administration (FDA) pre-market approval certificate took an average of 4.5 years to acquire [8].
4. Clinical trials are not exhaustive. Safety and efficacy can not be established for the entire population or for long-time use of the device/drug, which can lead to complications once the device/drug reaches the market [9].
5. Inability to explain underlying mechanisms behind treatment failure. Clinical trials are phenomenological and often cannot improve our understanding of why treatments fail in particular patients or under certain physiological conditions.
6. There can be ethical and environmental issues associated with clinical trials [10].

In-silico trials (ISTs) offer an alternative strategy towards the development of medical devices and other treatments that can help to mitigate the above issues [11, 12]. In-silico trials refer to pre-clinical trials performed using individualised computer modelling and simulation to virtually test some aspect of a medical device, drug or clinical procedure across a large range of anatomies and physiologies. Performing trials in-silico can facilitate more comprehensive judgement of the safety and efficacy of a treatment across the population. As a result, in-silico methodologies are being adopted in drug and medical device development [13–15] in the treatment of pathologies such as diabetes [16], acute ischemic stroke [17, 18] and COVID-19 [19].

The most relevant exemplar trial for this thesis is the Flow Diverter Performance Assessment (FD-PASS) IST [20]. In this study, Sarrami-Foroushani et al. [20] used computer modelling and simulation to perform an IST investigating the performance of the Pipeline Embolization Device (PED), a flow diverter device used for endovascular treatment of intracranial aneurysms. Fluid dynamics and biochemical reaction modelling were used to assess the safety and efficacy of the flow diverter in a cohort of 82 virtual patients, with normotensive and hypertensive physiology modelled for each patient through the use of boundary conditions calibrated by patient data. By defining haemodynamic metrics for assessing treatment performance, the results of the IST were able to be compared with the results from three conventional clinical trials. Sarrami-Foroushani et al. [20] found that the IST replicated the results from the conventional trials and was able to expand upon them by performing sub-group analyses and additional experiments that would not be feasible or ethical in conventional trials. The results from FD-PASS demonstrate the power of ISTs, but the methodology used to generate the results can be improved upon. Developing an automated simulation pipeline would reduce the manual input required to prepare and execute simulations, thus improving the scalability of the trial. Simulation acceleration techniques could also be used to decrease the computational resources required for each model evaluation. The run-time for one patient, one physiology and one device in FD-PASS was approximately one day, which could lead to prohibitively large run-times and costs when increasing the scale of the IST towards thousands of patients, hundreds of physiologies and multiple devices.

There are numerous challenges to overcome before widespread adoption of ISTs in medical device development becomes common practice. These challenges can largely be categorised as relating to regulatory uncertainties, financial/leadership issues and problems relating to modelling/simulation [5]. The aim of my PhD is to address some of the aspects of modelling and simulation that currently hinder widespread adoption of ISTs. Specifically, these challenges are:

1. Prohibitively large simulation costs and run-times.
2. Lack of exemplars demonstrating the usefulness of in-silico trials.
3. Highly complex modelling and simulation pipelines that require significant expert input to execute.

The first challenge will be addressed through the review and development of accelerated simulation methods using data-driven modelling techniques. The second and third challenges will be addressed by performing an exemplar IST into flow diversion of posterior communicating artery aneurysms using a semi-automated simulation pipeline. Figure 1.1.1 highlights the structure of the thesis chapters in relation to these challenges.

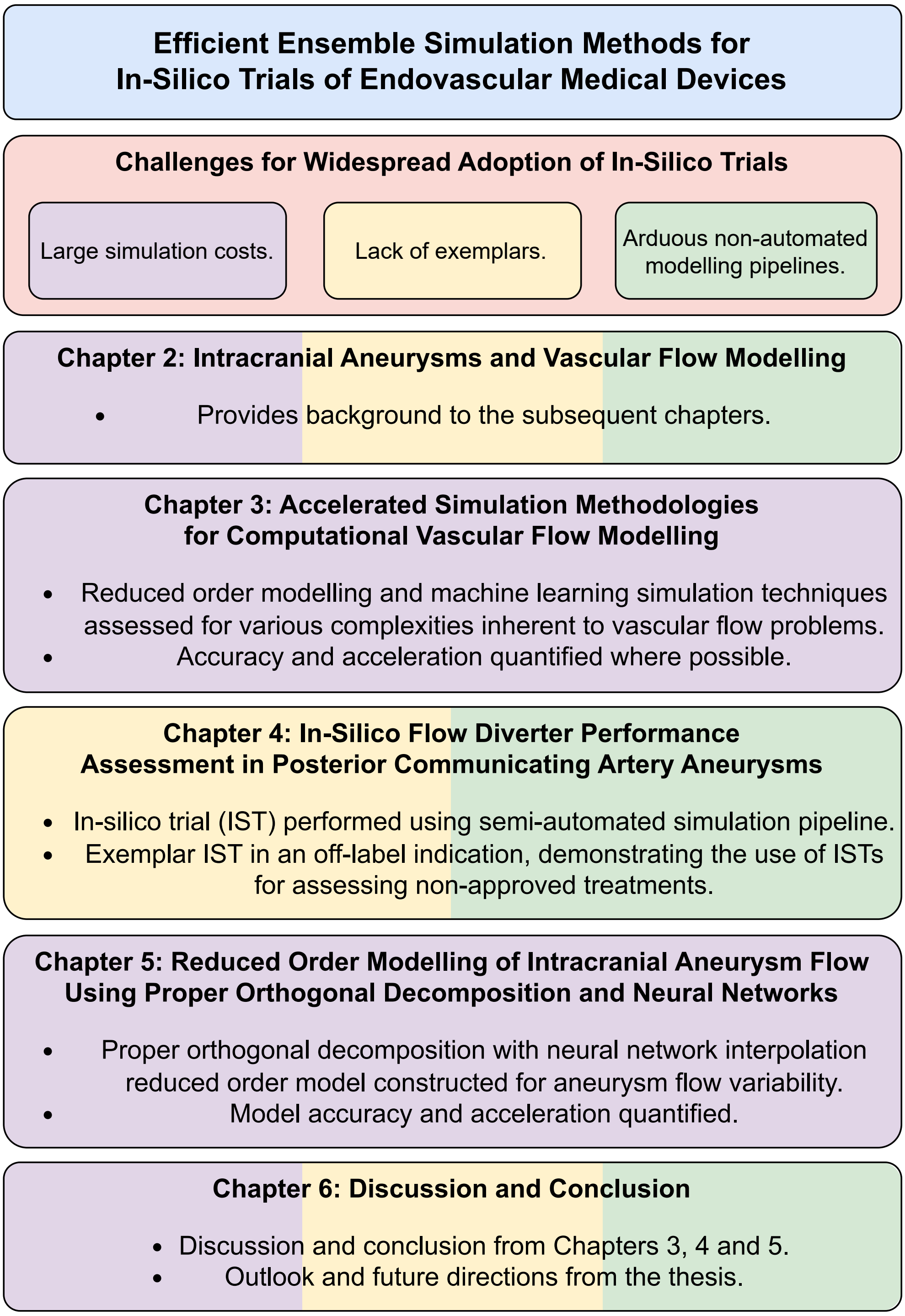


Figure 1.1.1: Thesis outline. The colour used for each chapter corresponds to the challenge that was tackled in that work.

1.2 Research Aims

1. Critically review accelerated simulation methodologies for vascular flow problems, identifying the types of application each method is suitable for and directions for future research.
2. Perform an in-silico trial investigating flow diverter treatment of intracranial aneurysms and automate the steps involved in the simulation pipeline.
3. Develop accelerated simulation methodology to facilitate efficient in-silico trials of endovascular medical devices and demonstrate the application of the methodology to a relevant problem.

1.3 Contributions

Various chapters within the thesis represent a distinct study that is published, under review or soon-to-be submitted. Below I state the contributions of each chapter and any associated publications.

- *Chapter 1* – I briefly discussed some of the issues relating to clinical trials, introduced in-silico trials for medical devices, overviewed a key paper by Sarrami-Foroushani et al. [20] and presented my research aims.

  Sarrami-Foroushani, A., Lassila, T., ***MacRaild, M.****, Asquith, J., Roes, K.C., Byrne, J.V. and Frangi, A.F., 2021. In-silico trial of intracranial flow diverters replicates and expands insights from conventional clinical trials. Nature communications, 12(1), pp.1-12.*

- *Chapter 2* – I provide background into intracranial aneurysms, treatment of intracranial aneurysms, vascular flow modelling and the use of simulation and modelling for understanding intracranial aneurysm treatment.

- *Chapter 3* – I perform an extensive literature review into the acceleration of computational vascular flow models. The review focuses on: (i) Reduced order modelling for accelerating vascular flow simulations; (ii) Machine learning-based approaches for accelerating vascular flow simulations.

  Chapters 2 and 3 both contributed to the following publication.

MacRaild, M., *Sarrami-Foroushani, A., Lassila, T. and Frangi, A.F., 2024. Accelerated simulation methodologies for computational vascular flow modelling. Journal of the Royal Society Interface, 21(211), p.20230565.*

- *Chapter 4* – I perform an in-silico trial assessing the performance of the Pipeline Embolization Device (PED) in posterior communicating artery (PComA) aneurysms using a semi-automated simulation pipeline. Treating PComA aneurysms with PED represents an off-label use of the flow diverter device with an uncertain outcome, particularly for patients with fetal posterior circulation. The trial establishes the usefulness of modelling and simulation in assessing treatment scenarios that may not be approved by the FDA.

  MacRaild, M., *Sarrami-Foroushani, A., Song, S., Liu, Q., Kelly, C., Ravikumar, N., Patankar, T., Lassila, T., Taylor, Z.T., and Frangi, A.F., 2023. Off-label in-silico flow diverter performance assessment in posterior communicating artery aneurysms. Journal of NeuroInterventional Surgery, under review.*

- *Chapter 5* – I develop a machine learning reduced order model for capturing physiological flow variation in intracranial aneurysm models. Computational fluid dynamics data was generated in a 3D aneurysm model, a reduced order model using proper orthogonal decomposition and neural network interpolation was constructed, and the model's accuracy and acceleration was assessed.

  MacRaild, M., *Sarrami-Foroushani, A., Lassila, T., and Frangi, A.F., 2023. Reduced order modelling of physiological flow variation in intracranial aneurysms using proper orthogonal decomposition and neural networks. International Journal for Numerical Methods in Biomedical Engineering, under review.*

- *Chapter 6* – I present a discussion of each thesis chapter individually, draw final conclusions from the thesis and present ideas for future studies.

CHAPTER 2

Intracranial Aneurysms and Vascular Flow Modelling

2.1 Introduction

The first and third research aims of this thesis are to review and develop accelerated simulation methodology. When developing methodology for accelerating vascular flow simulations, it is useful to have a target application that is suitably complex and interesting. Previous work in my research group has focused on modelling blood flow and treatments of intracranial aneurysms [20–23]. As such, this will be the target application used when developing the methodology. The second aim of this thesis is to perform an exemplar in-silico trial to generate evidence on medical device performance. It similarly makes sense to use intracranial aneurysm treatment modelling as the target application upon which to base the trial. This chapter provides clinical background on intracranial aneurysms, techniques for treating them, and modelling of their pre- and post-treatment haemodynamics. It also introduces vascular flow modelling and the specific complexities that arise in intracranial aneurysm modelling. Vascular flow modelling and its complexities form the background to Chapter 3 and aneurysm pathophysiology, treatment and modelling form the background to Chapters 4 and 5.

2.2 Intracranial Aneurysms

Intracranial aneurysm are pathological dilations in the arterial walls of the brain vasculature that form when weakness in the vessel wall causes it to balloon or bulge outwards [24]. The exact cause of this weakness is unknown, however it has been linked to a host of inherited and acquired risk factors, such as increasing age, female sex, smoking and head trauma [25]. It has been estimated that intracranial aneurysms occur in 5–8% of the general population, but the exact prevalence is unknown because most are asymptomatic [26, 27]. When an aneurysm ruptures, it causes subarachnoid haemorrhage, an event which carries high rates of mortality, morbidity and disability [28]. Despite improvement in clinical care of patients with subarachnoid haemorrhage, a quarter still die and roughly half live with persistent neurological deficits [29]. The estimated annual cost to the UK's National Health Service for aneurysmal subarachnoid haemorrhage was £168.2 million in 2005. Accounting for informal care and production losses the total economic burden was estimated to be £510 million annually [30].

Aneurysms can occur in all vessels of the brain vasculature, but are most commonly found on the Circle of Willis, a loop-like vessel structure that supplies blood to the brain and surrounding tissue (Figure 2.2.1). In the Circle of Willis, blood flow enters the system through the internal carotid and vertebral arteries and flow into the other surrounding arteries shown in Figure 2.2.1. The structure of the Circle of Willis means that each efferent artery can potentially be supplied by either the anterior (internal carotid arteries) or posterior (vertebral arteries) blood inflow. The posterior communicating arteries (PComAs) connect the anterior and posterior circulation and as a result their physiological function can vary depending on the structure of the surrounding vasculature. One example of this is the presence of fetal posterior circulation, a condition where atrophy or absence of the P1 segment of the posterior cerebral artery increases the blood flow demand on the PComA. This raises questions regarding treatment of aneurysms that originate from the PComA – this will be investigated further in Chapter 4 [2].

The most frequent aneurysm location is the anterior communicating artery (35%), then the internal carotid artery including the posterior communicating and ophthalmic arteries (30%), the middle cerebral artery (22%) and finally the posterior circulation locations [31]. Aneurysms also vary in pathogenesis. When a blood vessel balloons outwards on one side of the vessel wall, the resulting aneurysm is referred to as saccular. When the vessel bulges radially outwards around the entire wall, the aneurysm is referred to as fusiform. Saccular aneurysms are typically more spherical in shape and have a distinct neck, whereas fusiform aneurysms have no distinct neck. Aneurysms can be further classified as either side wall or terminal, with terminal aneurysms occurring at a vessel bifurcation.

Aneurysms are commonly categorised as small (maximum diameter < 10 mm), large (> 10 mm, < 20 mm), and giant (> 20 mm). Aneurysm morphology can also be described by parameters such as neck diameter, aspect ratio and non-sphericity index. Aspect ratio is typically the ratio between the distance from the neck to the aneurysm tip and the neck width. Non-sphericity index is derived from the volume and surface area of the aneurysm and is used to quantify the complexity of its shape [32]. Figure 2.2.2 illustrates these morphological quantities.

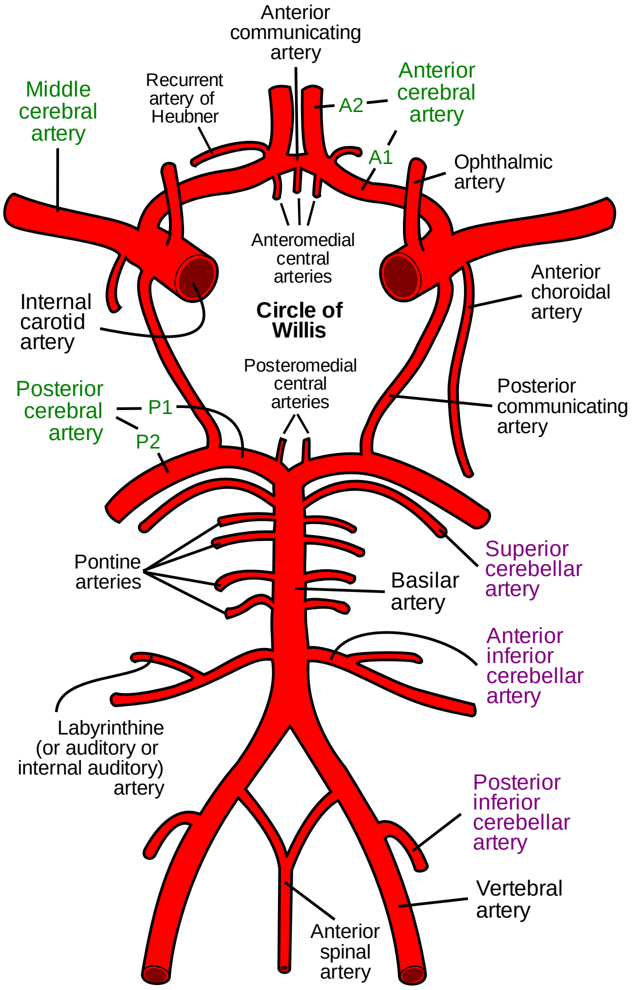


Figure 2.2.1: The Circle of Willis. Blood flows into the system through the internal carotid and vertebral arteries. The posterior communicating arteries, which are the subject of the in-silico trial in Chapter 4, connect the anterior and posterior circulation. This figure was adapted from a file freely available in the public domain (Wikimedia Commons).

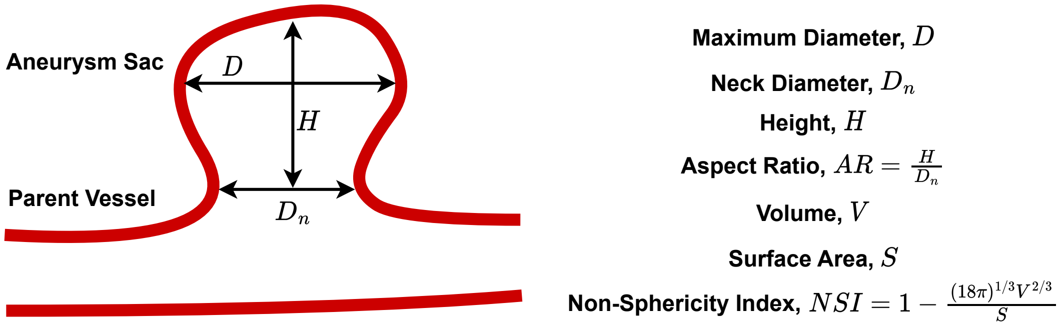


Figure 2.2.2: Common morphological parameters used to describe intracranial aneurysms.

Within a given aneurysm location and type (e.g. PComA, saccular, bifurcation), there is still significant variation in aneurysm size and shape. Figure 2.2.3 shows six PComA aneurysms and Table 2.2.1 gives morphological parameters for the aneurysms and the posterior communicating arteries for each case. These six cases demonstrate the large variation in shape and size possible amongst aneurysms arising from the same location. Aneurysm (a) is spherical in shape with a maximum diameter of 3.5 mm, an aspect ratio of 1.2, and a non-sphericity index of 0.15. Aneurysm (b) is at the upper end of the large category with a maximum diameter of 18.5 mm and an aspect ratio of 1.0. Aneurysm (c) has a high aspect ratio of 2.1 and a complex shape due to the formation of a daughter aneurysm on the original aneurysm. Aneurysm (c) has the largest non-sphericity index among these six aneurysms with a value of 0.30. Aneurysm (d) is similar in size and aspect ratio to aneurysm (a), but has a more complex shape and a greater non-sphericity index of 0.22 compared to (a). Aneurysm (e) has a low aspect ratio of 0.5 and a low non-sphericity index of 0.05 due to its uniform rounded shape. Aneurysm (f) has a complex-shape with a non-sphericity index of 0.28. Differences are also apparent in the size and shapes of the vessels surrounding the aneurysms. Of particular interest in this thesis is the posterior communicating artery, which is drastically different in size for cases (a) and (d), for example. The morphology of the aneurysm and the surrounding vessels are an important factor in treatment selection. The physiology of the vessels can also affect the treatment efficacy, but this will be discussed further in Chapter 4 [2].

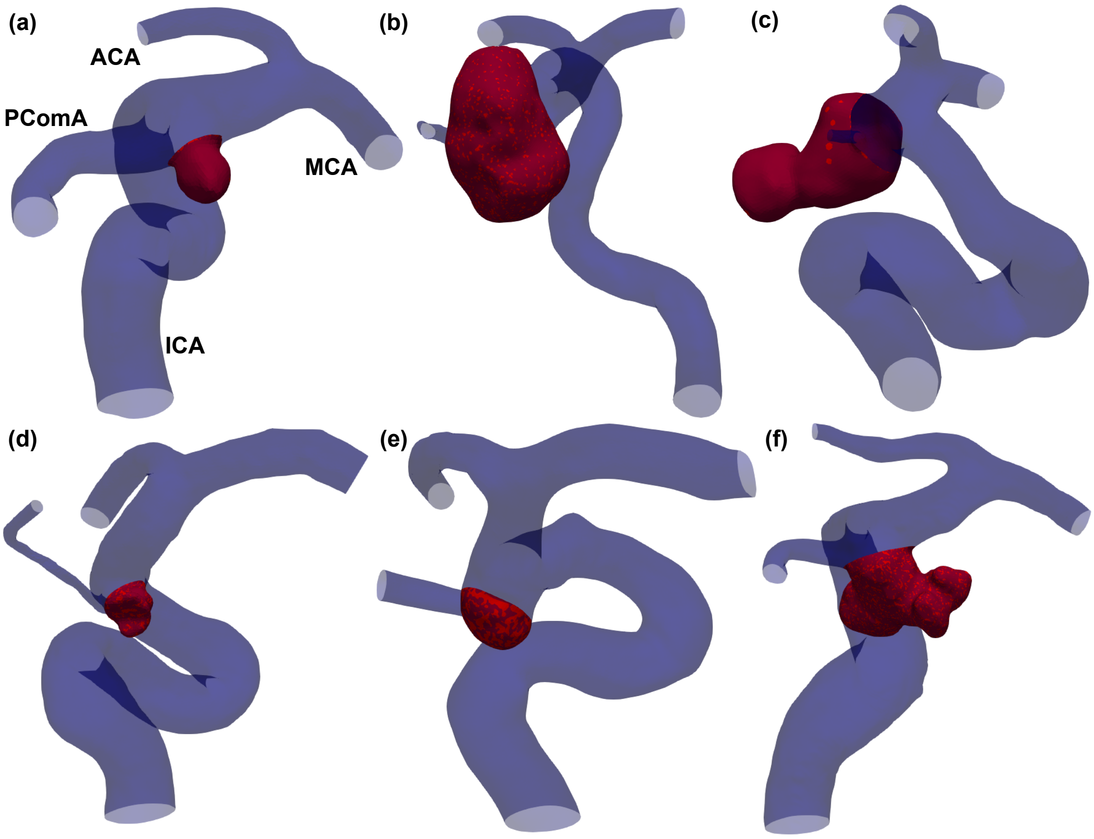


Figure 2.2.3: Six posterior communicating artery (PComA) aneurysm geometries from the patient cohort used in Chapter 4. ICA, internal carotid artery; MCA, middle cerebral artery; ACA, anterior cerebral artery.

Table 2.2.1: Six posterior communicating artery aneurysms and their morphological parameters. Aneurysm morphological parameters are provided by the AneuX database. PComA radius is taken as the radius of the PComA outlet surface used for the flow simulations in Chapter 4.

Figure 2.2.3	Max. diameter [mm]	Aspect ratio	Neck diameter [mm]	NSI	PComA radius [mm]
(a)	3.5	1.2	3.0	0.15	1.1
(b)	18.5	1.0	10.0	0.14	0.8
(c)	7.8	2.1	4.7	0.30	0.3
(d)	3.4	1.2	2.7	0.22	0.4
(e)	4.0	0.5	3.5	0.05	0.9
(f)	9.6	1.4	4.6	0.28	0.7

NSI, non-sphericity index; PComA, posterior communicating artery.

2.3 Intracranial Aneurysm Treatment

Improvements in diagnostic techniques using medical image analysis have increased the detection rate of unruptured intracranial aneurysms [33]. Despite occurring in 5–8% of the population, only about 1% of intracranial aneurysms actually rupture [34]. This means that clinicians are faced with the question of whether to treat an aneurysm upon discovery and how best to treat the aneurysm if treatment is deemed necessary.

It is thought that aneurysm rupture is a consequence of the vessel wall being unable to contain the haemodynamic load, with rupture occurring when wall stress exceeds wall strength [35]. The haemodynamic load stems from the pressure exerted by the blood flow upon the wall of the vessel. Interventional treatments of aneurysms aim to isolate the aneurysm from the circulatory system, mitigating the impact of blood flow variability on the weakened vessel wall. Isolating the aneurysm can be achieved through parent vessel occlusion, surgical clipping or endovascular techniques [1].

Parent vessel occlusion is a viable option for treatment of intracranial aneurysms where sufficient collateral circulation exists [24]. The procedure involves occluding the parent vessel through surgery or using an endovascular technique, such as balloons or coils, to prevent further blood flow to the vessel and therefore to the aneurysm. Without adequate collateral circulation that can supply the distal regions of the brain in place of the occluded vessel, this treatment method can result in a stroke.

Surgical clipping was developed in the 1930s and was the standard treatment for intracranial aneurysms for many years [36]. The procedure involves a craniotomy followed by placing a clip across the neck of the aneurysm, thereby excluding it from the circulation [24]. This treatment method has good long-term efficacy, but it is highly invasive and in some cases the morbidity and mortality rates associated with the surgery are greater than for rupture of the aneurysm without intervention [25].

An alternative to surgical clipping is endovascular coiling. This procedure involves catheter insertion of an electrolytically detachable platinum coil that adapts to different sizes and shapes of aneurysm [37]. This coil can be guided into the aneurysm and detached once a sufficient packing density has been achieved, as shown in Figure 2.3.1. Filling the aneurysm sac with coils reduces the blood flow velocity, which reduces the risk of rupture and promotes the growth of stasis-induced thrombus in the sac. This procedure raises additional questions, such as how densely to pack the coil [38]. Coiling is shown to be reasonably safe, with moderate effectiveness in terms of complete

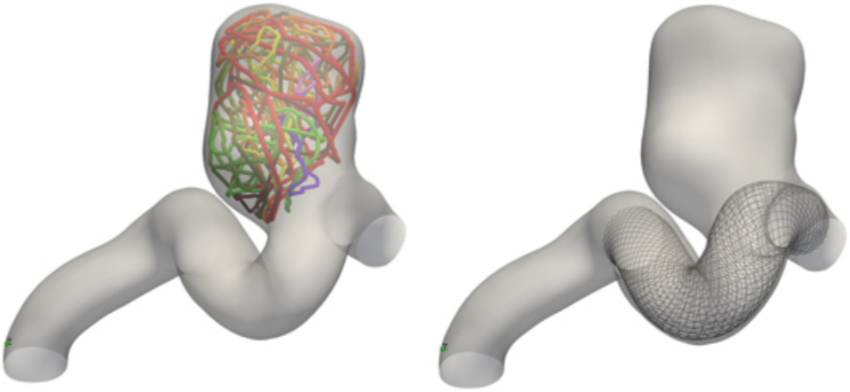


Figure 2.3.1: Virtual treatment of an aneurysm using coiling (left) and a flow diverter (right) [1].

occlusion of the aneurysm [39]. However, as with aneurysms treated through surgical clipping, the most difficult to treat with coil embolisation are those with a wide neck, defined as greater than 4 mm in diameter [40]. In wide neck aneurysms, there is a risk that the coil will migrate into the parent vessel and lead to an ischaemic stroke.

Another endovascular treatment technique is stenting, which has been used since the 1990s [41]. Early stent designs had a high porosity and so were rarely able to redirect flow away from the aneurysm sac of their own accord, but they can be used in conjunction with coiling for wide neck aneurysms to mitigate the risk of coil migration. Between the 1990s and 2008, improvements were made to the initial stent design to create what is called a flow diverter [42]. A flow diverter is a braided metallic cylindrical device that typically has a much lower porosity than a stent. Flow diverters are deployed in the parent vessel adjacent to the aneurysm, as shown in Figure 2.3.1. The goal of flow diversion is to reduce the blood flow into the aneurysm sac by redirecting flow away from the aneurysm and into the parent vessel. This process induces flow stasis in the aneurysm and promotes clot formation in the sac, which triggers the process of aneurysm healing [43]. Visualisations of blood flow streamlines before and after treatment and clot formation can be seen in Figure 2.6.1. Similarly to stents, flow diverters can be used in conjunction with coils for mitigating coil migration [43]. However, flow diverters can also be used by themselves and have demonstrated high occlusion rates irrespective of aneurysm size [44]. Studies have suggested higher complete occlusion with flow diversion (76%) compared with stent-assisted coiling (69%) [45–47].

Furthermore, cost-effectiveness analyses have favoured flow diversion over alternative endovascular treatments, particularly for large or giant aneurysms [45, 48].

The SILK and Pipeline Embolisation Device (PED) were the first two flow diverters to obtain CE mark approval in 2008 [43, 45]. The PED was also the first flow diverter to be approved by the U.S. Food and Drug Administration and is subsequently the most widely used device on the market at present [49]. However, the PED was only approved for use in a sub-group of aneurysms, namely those of large or giant size in the internal carotid artery from the petrous to the superior hypophyseal segments. As popularity and experience with the PED grew, its use has extended to various off-label indications, such as previously treated aneurysms, small aneurysms, posterior circulation aneurysms, fusiform aneurysms and distal circulation aneurysms [50]. In addition to the PED, intrasaccular devices such as the Woven Endobridge (WEB), Contour and Neqstent have recently emerged as alternative treatment approaches for wide-necked aneurysms. In spite of progress in treatments, flow diverter trials do not always produce good results. Two randomised trials into flow diverter performance (PARAT [51]) and FIAT [52]) reported below-expectation efficacies and high complication rates. The growing use of PEDs and other flow diverters in unapproved scenarios, as well as lingering questions in the approved use-case scenarios, motivates the demand for additional trials to better understand these issues. Performing the trials that would be required to answer these questions in-vivo (i.e. clinical trials) would be costly and potentially unethical in cases where the preliminary findings suggest that treatment performance will be inadequate. This motivates a paradigm shift towards in-silico trials based upon simulation and modelling, as this allows virtual performance assessment of a device in virtual patients across an envelope of morphologies, physiologies and device designs without risk to patients.

2.4 Vascular Flow Modelling

The impact of computational modelling is ubiquitous in the world around us. Modelling physical events and engineering systems governed by the laws of mechanics allows us to design and manufacture efficient transportation, predict weather patterns, improve healthcare and gain a deeper understanding of complex phenomena. Traditional computational models involve the simulation of high-dimensional numerical problems, which can incur prohibitively large run-times. This limits the use of traditional models in real-time and many-query applications. Real-time applications require that model evaluations occur almost instantaneously at the same or a faster rate than the real physical system being modelled. Many-query applications require large numbers of repeated model evaluations with varying parameters. Real-time and many-query problems often arise in the modelling of vascular blood flow. Real-time vascular flow simulations can be useful for providing personalised patient care quickly or for providing instant feedback to a surgeon undertaking a treatment procedure [53, 54]. Many-query vascular flow simulations can be used to iteratively design new vascular implements, establish safety and performance measures for treatment devices, and simulate interventions on a population scale through so-called in-silico trials [20, 55–57].

Vascular flow modelling poses various challenges due to the inherent complexities of the problem, which are highlighted in Figures 2.4.1 and 2.6.1 [58, 59]. Blood flow dynamics and tissue perfusion are governed by the Navier-Stokes equations, which are a nonlinear set of time-dependent partial differential equations [60]. Coupling the haemodynamics to solid mechanics or biochemical reaction models may also be required in certain applications. Fluid-structure interaction (FSI) is required when vessel distensibility is important or when there is a complex interaction between blood flow and valves or implanted devices [61–64]. Biochemical reactions are crucial in modelling thrombosis and endothelialisation depends on interactions between blood and blood-contacting surfaces of devices [23, 65]. The constitutive nature of blood adds additional complexity – it is a suspension containing various biochemically active particles and molecules, meaning that multi-phase multi-component flow-biochemistry models may be required when modelling flow-thrombosis in small vessels [23, 66].

As well as being multi-physical in nature, the length and time scales in vascular flow problems can differ greatly. Vascular flow is inherently pulsatile, which leads to features such as flow separation, vortex transport, mixing regions and impingement varying topologically throughout the cardiac cycle [67]. Variation in length scale and morphology can also influence these flow features. This leads to varying flow regimes in different regions of the vasculature and at different times of the cardiac cycle. Vascular flow modelling encompasses short-term processes such as systemic haemodynamics, autoregulation and recanalisation in addition to long-term processes such as remodelling and thrombosis [68–71]. Physiological changes due to factors such as age and lifestyle also have an impact on various flow problems. Vastly different length scales are also

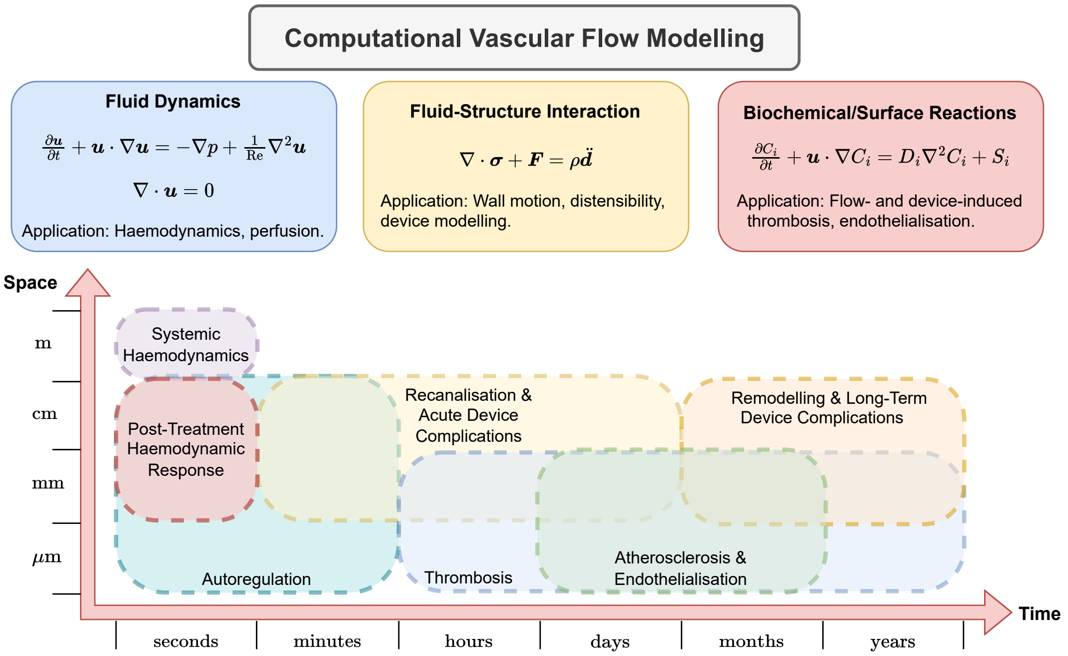


Figure 2.4.1: Vascular flow modelling is a multi-physics, multi-scale problem where nonlinearity and geometric complexity frequently arise.

present, with thrombosis and endothelialisation happening on a molecular level at the micro-scale, whereas systemic blood flow occurs in arteries with diameters up to a few centimetres.

Nonlinear effects further complicate vascular flow modelling. This can result from the convective nonlinearity in the Navier-Stokes equation, the geometric complexity of blood vessels, or the interactions across different length and time scales between blood flow and other physical and physiological phenomena. Blood is also shear-thinning, which can introduce additional nonlinearity [72]. Nonlinear flow features are often found in the presence of vascular pathologies such as stenosis, atherosclerosis, aneurysms or valve defects [73–76]. Flow-device interactions and the shear-thinning nature of blood can present additional sources of nonlinearity [77–80].

The most prominent complexities in vascular flow modelling can be summarised as: (i) nonlinearity, (ii) geometrical complexity, (iii) multi-physics, (iv) multi-scale in time, (v) multi-scale in space. In practice, assumptions can be made to simplify these complexities or eliminate them entirely for most problems, allowing for successful computational modelling. Despite this, using complex vascular flow models in real-time or many-query applications can still require additional acceleration through techniques such as reduced order modelling or machine learning.

2.5 Simulating Intracranial Aneurysm Haemodynamics

Haemodynamics is one of the most widely accepted factors contributing to aneurysm pathophysiology, playing a fundamental role in the mechanisms of initiation, growth and rupture [28]. Excessive haemodynamic loading of the vessel wall is thought to cause aneurysm rupture [35]. Studies using image-based computational fluid dynamics (CFD) modelling and statistical analyses have identified correlations between the haemodynamic properties of intracranial aneurysm flows and the likelihood of aneurysm growth and rupture [81, 82]. For this reason, CFD modelling is increasingly being used as a research tool to help understand the development, prognosis and treatment of intracranial aneurysms [83]. It is possible that aneurysm haemodynamics can provide metrics to improve rupture risk assessment in both untreated and treated aneurysms [28]. Some studies have attempted to replicate the exact anatomy of specific patients and connect haemodynamic factors to clinical events, which allows statistical analysis in a patient population [84, 85].

Xiang et al. [81] identified significant morphological and haemodynamic parameters that discriminate intracranial aneurysm rupture status. Both the morphological parameters (size ratio, undulation index, ellipticity index and non-sphericity index) and haemodynamic parameters (average wall shear stress (WSS), maximum intra-aneurysmal WSS, low WSS area, average oscillatory index, number of vortices and relative resident time) were found to be statistically significant. However, the only independently significant morphological variable was found to be size ratio, which is the ratio between aneurysm height and mean parent vessel diameter [86]. Both WSS and oscillatory shear index were found to be independently significant from the list of haemodynamic variables, suggesting that haemodynamics is as important as morphology in discriminating aneurysm rupture status.

Further to performing haemodynamic analyses of aneurysms, it is possible to implant treatment devices in the virtual patients and assess the performance and safety of the device. Various CFD studies have investigated a number of aneurysm treatment devices, such as coils [87, 88], stents [89], endovascular flow diverters [20, 23] and intrasaccular flow diverters [90].

2.6 The Complexity of Intracranial Aneurysm Haemodynamics

Section 2.4 outlined the various complexities present in vascular flow in general. Intracranial aneurysm haemodynamics demonstrate similar complexities.

The inclusion of a device into the simulation adds a large number of degrees of freedom into the problem due to the highly discretised mesh required to model the micro-scale device's components (e.g. the PED strut width is 30 μm). The device also impacts the flow dynamics, with the struts acting to break up the flow around the neck of the aneurysm. This leads to complex small-scale flow features and recirculation in the aneurysm sac, which are typically nonlinear flow behaviours that require a modelling approach that can resolve both the large- and small-scale flow features, i.e. one that is multi-scale in space. The application of suitable boundary conditions also requires that the multi-scale nature of the problem is taken into account. Inflow conditions can depend on the demography and physiology of the patient at hand, and the outflow conditions must similarly take care to produce a realistic flow split at vessel bifurcations that do not introduce unwanted uncertainty into the results [91].

Another complexity arises when considering the elasticity of the vessel and aneurysm walls. Fluid-structure interaction (FSI) can occur between the blood and vessel walls and plays an important role in the growth and rupture of aneurysms [92]. FSI can lead to a range of fluid mechanical effects, such as nonlinear pressure-drop/flow-rate relationships, wave propagation and generation of instabilities [93]. In addition, the device itself will interact with the vessel walls and the blood flow during deployment, so inclusion of these interactions may be important in order to accurately predict the deployed device configurations [94].

Once the device has been implanted, biochemical reactions are important to produce a more complete understanding of the treatment success and potential risks. Provided the device successfully reduces flow into the aneurysm sac, stasis-induced thrombosis is likely to occur. Sarrami-Foroushani et al. [23] developed a computational model for stasis-driven thrombosis based upon a series of biochemical equations modelled as a system of eight advection-diffusion-reaction equations for eight biochemical species. The biochemical model includes two-way coupling with the CFD model in order to capture the complex interplay between clot and flow. It is also possible that an alternative

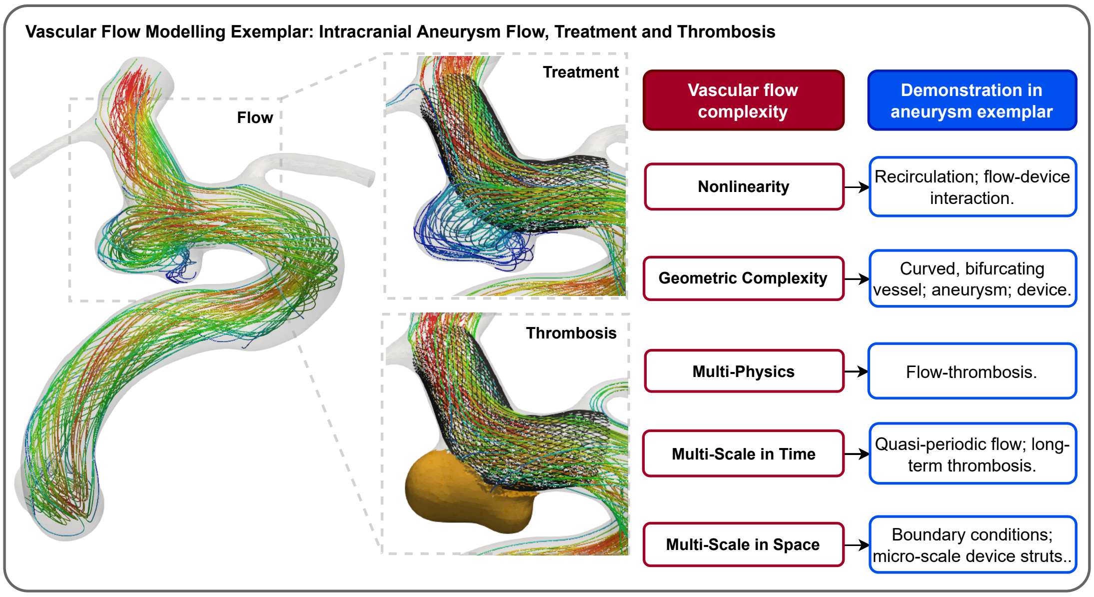


Figure 2.6.1: Aneurysm flow modelling as an exemplar that demonstrates the complexities that are common across various vascular flow modelling problems.

thrombosis pathway would be activated by the device itself and this would require a different set of equations to model. Thrombosis is typically a much slower process than the timescales over which flow reduction occurs, so simulating both the short-term haemodynamic effects of device implantation as well as the long-term thrombus formation presents a challenge that is multi-scale in time.

Figure 2.6.1 summarises the potential complexities that may need to be taken into consideration when modelling intracranial aneurysm flow, treatment and thrombosis. This application therefore represents a suitable exemplar to use throughout this thesis.

2.7 Aneurysm Haemodynamics Simulation Costs

While simulations can be used to understand aneurysm haemodynamics and treatments, they often take prohibitively long and cannot be used in a real-time or many-query context due to the excessive computational demand. Central Processing Unit (CPU) time of 40 hours per case was reported when running aneurysm flow with a mesh of roughly 1.7 million nodes on a standard workstation with four cores [95]. Sarrami-Foroushani et al. [23] found that flow simulations in untreated aneurysms with approximately 2 million mesh elements can take up to 10 hours on 32 processors to run for three cardiac cycles with 200 timesteps per cycle. Simulations with a flow diverter implanted have of order 10 million mesh elements and a correspondingly longer run time of roughly one day for the same simulation and CPU specifications. Simulations of flow and thrombosis in treated aneurysms can have run times of weeks or even months on 32 processors. Including fluid-structure interaction modelling would similarly increase simulation run times.

2.8 Conclusion

This chapter has introduced intracranial aneurysms and their treatment, vascular flow modelling, aneurysm haemodynamics and its complexities, and highlighted the cost of running such haemodynamics simulations. The development of accelerated simulation methodologies is essential for computational modelling to be viable in real-time and many-query contexts such as in-silico trials, patient-specific treatment planning and real-time surgery feedback. There are a vast array of accelerated simulation methodologies, some of which have been applied to vascular flow modelling and some of which have not. It is unclear at present which approaches are best suited the the various complexities of vascular flow modelling and what level of accuracy and acceleration they are capable of providing. To overcome this, Chapter 3 provides a thorough literature review of accelerated simulation methodologies for the target application of computational vascular flow modelling.

CHAPTER 3

Accelerated Simulation Methodologies for Vascular Flow Modelling

---

Abstract

Vascular flow modelling can improve our understanding of vascular pathologies and aid in developing safe and effective medical devices. Vascular flow models typically involve solving the nonlinear Navier-Stokes equations in complex anatomies and using physiological boundary conditions, often presenting a multi-physics and multi-scale computational problem to be solved. This leads to highly complex and expensive models that require excessive computational time.

This review explores accelerated simulation methodologies, specifically focusing on computational vascular flow modelling. Reduced order modelling (ROM) techniques like 0D/1D and modal decomposition-based ROMs and machine learning (ML) methods including ML-augmented ROMs, ML-based ROMs, and physics-informed ML models are reviewed. The applicability of each method to vascular flow acceleration and the effectiveness of the method in addressing domain-specific challenges are discussed. When available, statistics on accuracy and speed-up factors for various applications related to vascular flow simulation acceleration are provided.

The findings indicate that each type of model has strengths and limitations depending on the context. To accelerate real-world vascular flow problems, future research on developing multi-scale acceleration methods capable of handling the significant geometric variability inherent to such problems is proposed.

3.1 Introduction

The Motivation for Accelerating Vascular Flow Simulations

Despite the widespread use of computational models across many scientific disciplines, their use in real-time and many-query contexts is limited by their high computational cost. These scenarios frequently arise in vascular blood flow modelling. Real-time vascular flow simulations could provide guidance to clinicians prior to performing a treatment procedure or provide near-instant feedback during the procedure [53, 54]. Many-query vascular flow simulations can be used to iteratively design new vascular implements, establish safety and performance measures for treatment devices, and simulate interventions on a population scale through so-called *in-silico* trials [20].

Reduced Order Models and Machine Learning for Acceleration

Simulation acceleration refers to reducing the run-time of computational models and is typically achieved through modelling assumptions and simplifications. Reduced order models (ROMs) are low-order representations of high-order models that preserve essential model input-output behaviour at the cost of some model accuracy and are a common approach for accelerating expensive computational models [96, 97]. ROMs can be categorised into two families, *a priori ROMs* and *a posteriori ROMs*. The former seek to reduce the order of the system prior to solving the high-dimensional model, using techniques such as Spatial Dimension Reduction (SDR) or Proper Generalised Decomposition (PGD). *A posteriori ROMs* are data-driven techniques that depend on first solving the high-dimensional model or acquiring experimental data to generate snapshot solution fields. Snapshot data is decomposed into a reduced representation using, for example, Proper Orthogonal Decomposition (POD) [98–101], Dynamic Mode Decomposition (DMD) [102, 103] or variants thereof. The reduced representation can then be advanced in time directly or combined with projection or interpolation techniques to construct a ROM. There are a multitude of ROM techniques, some of which have been applied to vascular flow problems.

Recent advances in machine learning have improved some ROM methodologies and provided alternative techniques to accelerate simulations. Machine learning acceleration methods operate under a similar paradigm to many ROM techniques, with an expensive offline training phase that primes the model for fast online inference in new geometries, parameter values, or time points. There are various ways to use machine learning in simulation acceleration. Machine learning ROMs typically use machine learning to augment/replace a component of a ROM or they use machine learning entirely in place of existing ROM components [104, 105]. Physics-informed machine learning strategies are another possibility. In this approach, flow measurements are supplemented by additional constraints based on the underlying governing equations and boundary conditions [106]. Physics-agnostic techniques ignore the underlying physics of the problem, but instead use large amounts of data to identify mappings from images or geometries to flow quantities of interest [107]. Other techniques include tailor-made networks designed to handle point-cloud data [108, 109] and operator learning strategies [110, 111]. Given the relatively recent emergence of machine learning simulation techniques, they have not been widely applied to acceleration of vascular flow simulations yet.

Overview

This review aims to provide an overview of various methods for accelerating simulations and to collate, categorise and critique each method with respect to the target application of vascular flow modelling. Vascular flow modelling is decomposed into a series of complexities (nonlinearity, geometric complexity, multi-physics and multi-scale in time and space) and various acceleration methods are assessed with respect to these complexities. For ROM approaches, the review provides guidance on what type of vascular problems the method may be suitable for, what problems they have already been applied to, and how successful these studies were in terms of the accuracy and acceleration offered by the approach compared to traditional numerical methods. For machine learning approaches, the review introduces some common methods, discusses their benefits and limitations, and advises what vascular problems they may be suitable for. Throughout this review, acceleration factors are measured by comparing run times for a single evaluation of the accelerated and full-order models, unless otherwise stated. For complementary reviews on parametric model reduction, model order reduction in fluid dynamics, data-driven cardiovascular flow modelling, machine learning for cardiovascular biomechanics, real-time simulation of computational surgery, and the challenges of vascular fluid dynamics, see [59, 96, 99, 112–114]. Finally, note that although this review focuses on vascular flow acceleration, the complexities of this application (nonlinearity, geometric complexity, multi-physics and multi-scale) are encountered across many other computational modelling domains. Therefore, this review will be relevant and useful to computational vascular flow modelling researchers and the broader computational modelling community.

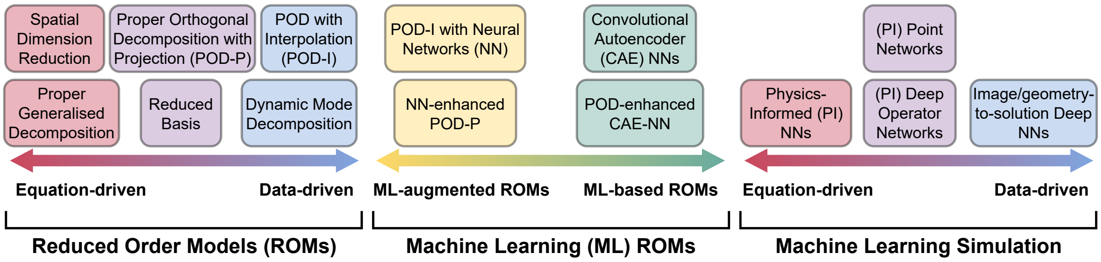


Figure 3.1.1: Taxonomy of various simulation acceleration methods reviewed in this chapter.

3.2 Reduced Order Modelling of Vascular Flow

Reduced order models (ROMs) aim to reduce the dimensionality of a numerical problem either by applying prior knowledge of the problem itself or by inferring knowledge based on previously gathered data from the system of interest. ROM methods can be described as *a priori* or *a posteriori*, depending on whether the reduction of the system exploits prior knowledge about the full-order model (FOM) or information (data) collected after solving it, respectively. *A priori* methods are useful when there exist symmetries or other known information about the underlying system, or when the system is too complex to solve with traditional techniques. *A posteriori* methods are useful when readily available data from the FOM can be used to guide the construction of the ROM. Whether the approach is intrusive or non-intrusive is another way to categorise ROM methods. Intrusive methods require the explicit use of the underlying high-order numerical implementation of the FOM, whereas non-intrusive methods operate entirely separate to the FOM. Intrusive methods can be more numerically robust due to their incorporation of the underlying governing equations, but non-intrusive techniques can be easier to implement and use in conjunction with commercial solvers, which are common when studying fluid dynamics problems. Many categories of ROM have been applied to vascular flow, with various benefits and limitations to each approach. This section will describe some of the most common ROM techniques and their suitability to model various vascular flow complexities.

3.2.1 Spatial Dimension Reduction

The 3D unsteady incompressible Navier-Stokes equations in non-dimensional form are: find $(\boldsymbol{u}, p) \in H^1(\Omega; \mathbb{R}^d) \times L^2(\Omega; \mathbb{R})$ s.t.

$$\frac{\partial \boldsymbol{u}}{\partial t} + \boldsymbol{u} \cdot \nabla \boldsymbol{u} = -\nabla p + \frac{1}{\mathrm{Re}} \nabla^2 \boldsymbol{u}, \qquad \nabla \cdot \boldsymbol{u} = 0, \tag{3.2.1}$$

where $\boldsymbol{u}$ is the velocity, p is the pressure, and Re is the Reynolds number dependent upon the fluid density ρ and dynamic viscosity μ. The spatial dimension is $d = 3$ except for some cases of plane symmetric or axisymmetric flow, when $d = 2$, and the domain $\Omega \subset \mathbb{R}^d$ has a suitably regular boundary to ensure the existence of solutions. Spatial Dimension Reduction (SDR) involves reducing these equations down to a 0D/1D/2D model that describes bulk quantities instead of the full spatio-temporal flow fields. A comprehensive review of 0D and 1D techniques has been provided by Shi et al. [115]. This review provides an overview of the approach, quantifies the acceleration and accuracy offered, and discusses how applicable this method is to vascular flow simulation acceleration.

0D Models Lumped parameter models (referred from hereon in as 0D models) exploit the analogy between hydraulic networks and electrical circuits. Blood pressure and flow rate are represented by voltage and current, and the frictional, inertial, and elastic effects of blood flow are described by electrical resistance, inductance, and capacitance, respectively [115]. Established methods for modelling electrical circuits (Kirchhoff's current law, Ohm's law for voltage-current) with ordinary differential equations (ODEs) can then be used to describe vascular flow problems.

The first 0D models were based on the Windkessel model, which consists of a capacitor that describes the storage properties of large arteries and a resistor that describes the dissipative nature of small peripheral vessels [115]. This simple approach cannot model specific pressure and flow rate changes in particular vascular segments and cannot fully describe the effects of arterial impedance, venous pressure fluctuations or pulse wave transmission. Various extensions to this model have been used to capture these more complex physiological phenomena by adding additional resistors, inductances and capacitors. For example, in a system with capacitance/compliance C, voltage/pressure

P, charge/flow rate Q, inductance/inertia L, and resistance R, the two ODEs describing the system are [116]:

$$C\frac{\mathrm{d}P_1}{\mathrm{d}t} + Q_2 - Q_1 = 0, \quad L\frac{\mathrm{d}Q_2}{\mathrm{d}t} + P_2 - P_1 = -RQ_2. \tag{3.2.2}$$

Multi-compartment models can also be used to describe flow and pressure characteristics within specific vascular segments.

1D Models In 1D models, the form of the velocity profile across the vessel radius is constrained, which simplifies the 3D governing equations. One-dimensional blood flow is governed by the axisymmetric forms of the incompressible continuity and Navier-Stokes equations, which can be written as:

$$\frac{\partial A}{\partial t} + \frac{\partial (AU)}{\partial x} = 0, \qquad \frac{\partial U}{\partial t} + U\frac{\partial U}{\partial x} + \frac{1}{\rho}\frac{\partial p}{\partial x} = \frac{f}{\rho A}, \tag{3.2.3}$$

where x is a local coordinate describing the vessel segment, A is the cross-sectional area, U and p are the cross-sectionally-averaged velocity and pressure, ρ is the blood density and f is a viscosity-dependent term describing the frictional force per unit length [115, 117]. These equations can be further coupled to a pressure-radius relationship that describes the elasticity of the vessel wall. The reduced equations can be solved using various numerical techniques, such as the method of characteristics [118, 119] or finite differences [120].

A primary benefit of 1D models over 0D models is that they can capture pressure and velocity pulse wave propagation [121]. Waves carry information about the medium in which they travel, so capturing the pressure and velocity waves in blood vessels can provide insight regarding the function of the cardiovascular system and provide information about various vascular pathologies, such as atherosclerosis and hypertension [122].

2D Models For 2D vascular models, the 3D vessel loses its torsion and curvature, becoming a straightened tube governed by the 2D Navier-Stokes equations. Two-dimensional models include the radial variation of the velocity and pressure fields in an axisymmetric tube, whereas 1D models only consider the cross-sectionally-averaged quantities. These models are used less frequently now due to improved computer processing power and widely available commercial solvers that make solving the 3D problem more tractable [123]. However, in certain applications, such as the calculation of fractional flow reserve (FFR), 2D models are shown to be significantly faster than 3D models while retaining a clinically viable level of accuracy [124].

Summary Table 3.2.1 summarises several vascular flow ROM studies using SDR methods. Included in the table are the specific application, the reported accuracy compared to the FOM as a baseline, and the acceleration factor compared to the FOM. The accuracy reported for most ROMs was $> 90\%$ and the acceleration factors ranged from 10^2–10^5. However, the ROMs are limited to investigating simple flow parameters, such as FFR or flow rates. Gashi et al. [124] demonstrated that adding complexity (steady-state to unsteady) reduces the acceleration offered by three orders of magnitude. Mirramezani and Shadden [125] presented a comprehensive study applying distributed 1D lumped parameter models to aortic, aorto-femoral, coronary, cerebrovascular, pulmonary and paediatric blood flow problems. Analytical expressions were used to allow the model to capture energy losses along vascular segments due to viscous dissipation, unsteadiness, flow separation, vessel curvature and vessel bifurcations.

Conclusion Zero-dimensional SDR models are suitable for global pressure/flow rate analysis of large regions of the cardiovascular system [115]. One-dimensional models assume axisymmetric flow solutions to capture pressure and velocity pulse wave propagation [121]. Two-dimensional models can evaluate local flow fields with radial velocity variation in axisymmetric domains [131]. A prominent use of SDR models is providing boundary conditions to 3D models that incorporate information from significantly larger portions of the vasculature than it would be feasible to model in 3D [132–140]. In this way, SDR models can facilitate multi-scale spatial models that provide well-resolved 3D flow information in local regions of interest while still including the effect of distal or proximal regions. Zero-dimensional SDR models are unable to describe the nonlinearities that can arise in cardiovascular mechanics due to the convective acceleration term in the Navier-Stokes equations and/or the complex velocity-pressure relationship in distensible vessels [115]. One-dimensional SDR models can approximate the effect of vessel wall elasticity on blood flow by adding a constitutive law that relates blood pressure to vessel cross-sectional area [121]. SDR models are generally only suitable for bulk velocity/pressure analysis in relatively simple geometries (i.e. axisymmetry is a valid assumption). They are typically unsuitable for complex multi-physics or multi-scale temporal problems but well-suited for spatial multi-scale problems.

Table 3.2.1: Various ROM papers using SDR for vascular flow problems. Acceleration is measured by comparing the time taken for one ROM evaluation with one FOM evaluation. This is the case for all tables presenting acceleration statistics, unless otherwise stated.

Reference	**Method**	**Application**	**Accuracy**	**Acceleration Factor**
Grinberg et al. [117]	1D	Pulsatile intracranial blood flow	-	147,000*
Reymond et al. [126]	1D	Pressure drop and WSS calculation in systemic arterial tree	-	-
Blanco et al. [127]	1D	FFR calculation in coronary arteries	98%	WCT: 302 NT: 2870
Xiao et al. [128]	1D	Baseline CCA	>99%	-
		Baseline aorta	>98%	-
		Aortic bifurcation	>98%	-
Papadakis and Raspaud [129]	1D (extended for stenosis)	Wave propagation in stenotic vessels	>99%	-
Jonášová et al. [130]	1D	Outlet flow rate in hepatic vein network	88% AW: 99%	-
Mirramezani and Shadden [125]	1D	Flow rate and pressure calculations in various vascular domains	>93%	>1000
Gashi et al. [124]	2D steady	FFR calculation in coronary arteries	95%	162,000
	2D unsteady		98%	195

* Calculated by assuming a linear relationship between number of CPUs and simulation execution time.

AW, area-weighted; CCA, common carotid artery; FFR, fractional flow reserve; NT, normalised time (WCT × number of computation tasks); ROM, reduced order model; SDR, spatial dimension reduction; WCT, wall clock time.

3.2.2 Proper Orthogonal Decomposition

Spatial Dimension Reduction (SDR) methods depend upon being able to apply geometrical simplifications (i.e. axisymmetry) or analogies with electrical circuit analysis to the vascular flow problem at hand to simplify the 3D Navier-Stokes equation into something easier and faster to solve. While SDR methods can be useful in capturing bulk quantities across large spatial scales, the applicability of these methods to other vascular flow complexities is limited. An alternative approach is to solve the expensive 3D Navier-Stokes equations and leverage the wealth of information contained in the data generated from these simulations to develop a ROM for the specific problem solved in the first instance. This is often referred to as a data-driven (or *a posteriori*) approach, as prior to ROM construction the FOM must be solved for some instances.

The method used to extract low-dimensional structures from high-dimensional data is key to any data-driven ROM. The most commonly used approach for this in fluid dynamics is the Proper Orthogonal Decomposition (POD). POD was first introduced in fluid dynamics to analyse the structure of experimental turbulent flow [141] and was later adopted for the purpose of efficient simulation and control of fluid flows [142]. POD extracts leading-order information from data in the form of orthogonal modes ordered by their energetic contribution to the data. In fluid flows, these modes typically capture spatial information contained within the data.

Before performing the POD, a snapshot matrix $\boldsymbol{U}$ is constructed by stacking columns of spatial data from different timesteps or input parameter configurations in a large matrix. A mean state derived by averaging over the timesteps or parameter configurations will often be subtracted from the snapshot matrix prior to performing the decomposition. Typically, the snapshot matrix will have many more rows than columns. POD is then performed by taking the Singular Value Decomposition (SVD) of $\boldsymbol{U}$:

$$\boldsymbol{U} = \boldsymbol{\Phi}\boldsymbol{\Sigma}\boldsymbol{V}^*, \tag{3.2.4}$$

where $\boldsymbol{\Phi}$ is a matrix of the left singular vectors, or POD modes, $\boldsymbol{\Sigma}$ is a diagonal matrix containing the singular values, and $\boldsymbol{V}^*$ is a matrix of right singular vectors. The success of POD in model order reduction stems from the observation that, in most complex physical systems, the meaningful behaviour of a system is captured by a low-dimensional subspace spanned by the first few POD modes. The singular values quantify the relative importance of each POD mode based upon its energetic contribution to the snapshot matrix. This knowledge makes it possible to truncate the system to a certain energy level by discarding the low-energy POD modes and retaining the high-energy modes.

POD-based ROMs have seen widespread application in areas such as classical fluid dynamics problems [143–145], aerodynamics [146], fluid-structure interaction (FSI) [147, 148] and blood flow problems [93, 100, 148–152]. However, POD alone is not sufficient to build a ROM. POD provides a low-dimensional representation of the snapshots of the system, but the low-order representation must be combined with projection or interpolation techniques to build a ROM that can predict solution fields at new timesteps or input parameter configurations.

POD-Projection

Projection-based methods use the underlying governing equations of a system and POD modes to construct a ROM. The governing equations are projected onto the POD basis to derive a set of reduced equations embedded in this low-dimensional space. A common approach is to use the Galerkin projection (GP) [153, 154]. POD-GP ROMs are among the most common ROMs that have been applied to vascular flow problems [100, 151, 152].

A POD-GP ROM can be derived by decomposing the velocity field $\boldsymbol{u}(\boldsymbol{x}, t)$:

$$\boldsymbol{u}(\boldsymbol{x}, t) \approx \sum_{j=1}^{N} a_j(t) \boldsymbol{\Phi}_j(\boldsymbol{x}), \tag{3.2.5}$$

where $\boldsymbol{\Phi}_j$ denote the POD modes and a_j are the temporal coefficients. The Galerkin projection of the Navier-Stokes equations is written as

$$\langle \boldsymbol{\Phi}_i, \frac{\partial \boldsymbol{u}}{\partial t} + \boldsymbol{u} \cdot \nabla \boldsymbol{u} \rangle = -\langle \boldsymbol{\Phi}_i, \nabla p \rangle + \langle \boldsymbol{\Phi}_i, \frac{1}{\mathrm{Re}} \nabla^2 \boldsymbol{u} \rangle, \tag{3.2.6}$$

where $\langle \cdot, \cdot \rangle$ represents the inner product. Following some algebraic manipulation using the decomposition from equation (3.2.5), the POD-GP ROM can be written as [155]:

$$\frac{\mathrm{d}a_i(t)}{\mathrm{d}t} = A_i + \sum_{j=1}^{N} B_{ij} a_j(t) + \sum_{j=1}^{N} \sum_{k=1}^{N} C_{ijk} a_j(t) a_k(t), \quad i = 1, \ldots, N. \tag{3.2.7}$$

A_i, B_{ij} and C_{ijk} are tensors determined by the specific form of the governing system.

The functional forms of the coefficient tensors are:

$$\begin{aligned} A_i &= -\frac{1}{\mathrm{Re}}\langle \nabla \boldsymbol{\Phi}_i, \nabla \bar{\boldsymbol{u}} \rangle - \langle \boldsymbol{\Phi}_i, (\bar{\boldsymbol{u}} \cdot \nabla)\bar{\boldsymbol{u}} \rangle \\ B_{ij} &= -\langle \boldsymbol{\Phi}_i, (\bar{\boldsymbol{u}} \cdot \nabla)\boldsymbol{\Phi}_j \rangle = \langle \boldsymbol{\Phi}_i, (\boldsymbol{\Phi}_j \cdot \nabla)\bar{\boldsymbol{u}} - \frac{1}{\mathrm{Re}}\langle \nabla \boldsymbol{\Phi}_i, \nabla \boldsymbol{\Phi}_j \rangle \\ C_{ijk} &= -\langle \boldsymbol{\Phi}_i, (\boldsymbol{\Phi}_j \cdot \nabla)\boldsymbol{\Phi}_k \rangle, \end{aligned} \tag{3.2.8}$$

where $\bar{\boldsymbol{u}} = \int_0^T \boldsymbol{u}(\boldsymbol{x}, t)\mathrm{d}t$ is the time-averaged flow [99]. The double sum in equation (3.2.7) arises due to the nonlinearity of the Navier-Stokes equations and is responsible for the slower ROM speeds and greater storage demands required in the case of nonlinear systems.

Nonlinearity When applied to problems governed by nonlinear equations, POD-GP does not fully decouple the ROM equations from the FOM, as the algebraic form of the ROM equations retains dependence on the FOM [99]. This means that the algebraic operators for the ROM need to be recomputed at every iteration of the system, which limits the acceleration that this approach can offer for the target application of vascular flow. It is possible to overcome this issue by using hyper-reduction techniques, such as the Discrete Empirical Interpolation Method (DEIM), which approximates the algebraic operators instead of calculating them exactly [156]. Buoso et al. [100] employed this technique in a POD-GP-DEIM ROM to evaluate coronary blood flow, and found an acceleration by a factor of 25 for this method compared to the FOM.

Geometric complexity Complex geometric variability can be modelled by POD-Projection methods, as the POD modes can be made to contain spatial information about the geometry used to generate the data by mapping them back to a fixed reference geometry. However, applying any kind of ROM to a geometry not included in the training data is typically very challenging. In particular, when looking at vascular flow, the variability in morphology from one person to the next can be extreme, with entire vascular segments sometimes missing in certain regions [157]. In some cases, for example when modelling relatively simple features such as stenosis in reasonably straight vessels, it is possible to parameterise the geometric variation and include these parameters as input to the ROM, as in [100]. However, for pathologies such as intracranial aneurysms, where blood flow is usually highly dependent on the morphology, the number of parameters needed and the amount of high-fidelity data required can be prohibitive. Buoso et al. [100] demonstrated the use of DEIM to accelerate mesh generation by a factor of ten, which could help improve the overall efficiency of a simulation pipeline studying blood flow in multiple geometries.

Multi-physics Provided the governing equations are known and data can be generated for the system, POD-Projection techniques are suitable for multi-physics problems. A common multi-physics application of POD-Projection is to FSI problems [158–160]. Ballarin and Rozza [161] applied a POD-GP ROM to three idealised 2D FSI problems, including a parameterised valve configuration. The ROM showed good qualitative agreement across all cases and an acceleration factor of the order of ten.

Multi-scale (time) While POD-Projection ROMs are able to reduce simulation times significantly, the long-term stability of the ROM for unsteady flow problems is not guaranteed [99, 162]. This instability can be related to the truncation of the POD basis, the violation of boundary conditions, or an inherent lack of numerical stability [155]. Various stabilisation techniques can overcome these issues, such as balanced truncation and balanced POD [99], pressure stabilisation [144] or adding corrective terms to the ROM equations to increase dissipation [163]. Adding these stabilisation techniques to a ROM may increase its long-time accuracy, but will likely come at the cost of increased computational demands [159]. Lassila et al. [99] noted that periodically driven inflow problems have been shown to demonstrate accurate long-term predictions. Given the quasi-periodic nature of vascular flow, this may imply that ROM stability is satisfactory in this context. However, care must be taken to train the ROM with data that is representative of the entire cardiac cycle. Flow features will exhibit strong time dependence due to the pulsatile nature of vascular flow [67]. As a result, training a ROM using data from only one part of the cardiac cycle (e.g. flow acceleration) is unlikely to produce a ROM capable of accurately predicting the flow at another time (e.g. diastole).

Multi-scale (space) The spatial information is contained within the POD modes when constructing a POD-Projection ROM. The number of spatial degrees of freedom is the same as the number of rows in each POD mode, which means that data and computing requirements for POD ROMs will increase as the mesh size grows. Furthermore, as POD requires input data from a FOM, using a refined mesh that captures fine flow details could lead to prohibitive run times when solving the FOM. This means that POD-Projection ROMs are often unsuitable for problems where large regions of the vasculature need to be modelled.

A possible strategy to mitigate this issue is to couple a POD-Projection ROM with boundary conditions that are derived from a SDR ROM. Using this technique allows for the high spatial resolution of the POD-Projection approach in the region of interest while still accounting for the effects of the proximal and/or distal vasculature using the SDR model. This technique has been used in various haemodynamics studies to couple high-fidelity 3D models to SDR models, but POD-Projection ROMs have not been used for the 3D model [137, 150, 164].

Additional comments While using the underlying governing equations is thought to improve the robustness of projection-based ROMs, it also reduces the ease of implementation which could be regarded as a weakness. Constructing a projection-based ROM requires explicit use of the underlying numerical implementation of the FOM, which may not be available or straightforward to use. In particular, when solving fluid dynamics problems, researchers often turn to commercial software for which source code is not readily available. This can hinder incorporating projection-based ROMs into simulation pipelines that are not built upon open-source software. Equation-free or non-intrusive methods offer an alternative strategy that mitigates these issues.

POD-Interpolation

An alternative to projection-based ROMs is to use interpolation-based methods. Given a snapshot matrix $\boldsymbol{U}$, with SVD given by $\boldsymbol{U} = \boldsymbol{\Phi\Sigma V}^*$, it is possible to reconstruct each column of $\boldsymbol{U}$ using:

$$\boldsymbol{u}^n(\boldsymbol{x}, t; \boldsymbol{\mu}) = \sum_{j=1}^{N} a_j^n(t; \boldsymbol{\mu}) \boldsymbol{\Phi}_j(\boldsymbol{x}), \tag{3.2.9}$$

where $\boldsymbol{\mu}$ are the parameter configurations contained in the snapshots, $a_j^n(t; \boldsymbol{\mu})$ are a set of time and parameter dependent coefficients, N is the number of truncated POD modes retained for the ROM and $\boldsymbol{\Phi}_j$ are the POD modes. $\boldsymbol{a}^n$ are a set of temporal coefficients that can be considered as a path through the coordinate system given by $\boldsymbol{\Phi}$ [146]. The goal of POD-Interpolation is to predict the trajectory of the system under a new set of parameter values by using interpolation between the trajectories of previously computed parameter values. To perform the interpolation step, authors have turned to various techniques, such as linear interpolation [165], radial basis functions (RBFs) [146, 166, 167], Taylor series methods or Smolyak grids [168]. Once calculated, the new set of coefficients can be multiplied by the retained POD modes to efficiently calculate the solution field of interest for a new parameter configuration or time point.

Nonlinearity POD-Interpolation is a non-intrusive method, meaning that no modification of the underlying FOM numerical code is required. This means that the ROM is agnostic to the system it is being applied to and, therefore, POD-Interpolation does not suffer the same drawbacks as POD-Projection when applied to nonlinear systems. This does not guarantee that results with a POD-Interpolation approach will be accurate for a nonlinear system, but the speed of the model is not drastically reduced in this scenario as can be the case when using POD-Projection on nonlinear problems.

Geometric complexity Similarly to POD-Projection methods, POD-Interpolation is suitable for complex-shaped individual geometries due to the POD modes containing rich spatial information. However, the success of this approach is also limited when applied to geometries that were not included in the training data. Girfoglio et al. [139] applied POD-Interpolation methods to patient-specific aortic blood flow in the presence of a left ventricular assist device, but only constructed their ROM for a single patient geometry. Geometric parameterisation approaches have been applied to POD-Interpolation methods, but not in the context of vascular flow problems [146].

POD-Interpolation approaches can be applied to sub-domains of the FOM domain used to generate the snapshots. For example, if high-fidelity data was generated for a vessel with an aneurysm, it is possible to build a POD-Interpolation ROM for only the aneurysm rather than the full geometry. This can further accelerate the ROM, as the number of data and interpolation operations required is reduced. This feature of POD-Interpolation ROMs gives them an advantage when modelling flow in complex geometries where dense volumetric meshes are required (e.g. when modelling a flow-diverting stent), as the amount of data is vastly reduced without affecting the model performance.

Multi-physics POD-Interpolation techniques have been applied infrequently to multi-physics problems. Xiao et al. [147] used a non-intrusive POD-RBF ROM for one-way and two-way coupled FSI problems and found acceleration factors of order 10^5–10^6 whilst showing qualitative ROM-FOM agreement. Hajisharifi et al. [167] applied a POD-RBF ROM to a fluidised bed problem. Compared to the FOM, the POD-RBF ROM provided an acceleration factor of order 10^5 and an accuracy of approximately 99% when reconstructing the time evolution of the Eulerian and Lagrangian fields. They tested local and global POD approaches and found the local calculation of POD bases produced a more accurate and efficient ROM.

Multi-scale (time) Similarly to POD-Projection techniques, POD-Interpolation methods do not have any guarantee of long-term solution stability.

Multi-scale (space) In principle, POD-Interpolation ROMs can be coupled with 0D/1D models for boundary conditions by including the coupling parameters describing the inflow/outflow conditions in the ROM construction. When evaluating the POD-Interpolation ROM, one can obtain the boundary condition parameter input from the output of the 0D/1D boundary condition model and use this to evaluate the 3D flow field using the ROM. In this way, POD-Interpolation approaches can be suitable for modelling highly resolved regions of interest in 3D while conscribing to the effects of the peripheral vasculature. This POD-Interpolation-SDR approach is yet to be applied to vascular flow, but coupling 0D/1D models with 3D computational fluid dynamics (CFD) is common [137, 150, 164].

Additional comments Walton et al. [146] noted that POD-Interpolation, when all POD modes are retained, is equivalent to performing element-wise interpolation across all spatio-temporal coordinates. Therefore, the maximum accuracy for a POD-Interpolation ROM will be bounded by the element-wise interpolation error. For this reason, the acceleration offered by POD-Interpolation ROMs should not only be calculated relative to the high-fidelity CFD model, but also relative to the cost of performing element-wise interpolation of the solution field. Despite this limitation, relative to element-wise interpolation, POD-Interpolation is still capable of vastly reducing the number of interpolation operations required to calculate a new solution and the amount of data that needs to be stored offline.

Summary POD-Projection and POD-Interpolation techniques have been applied to a wide range of vascular flow problems, including blood flow in tetralogy of Fallot patients [149, 150], coronary blood flow [100, 151, 169], aneurysm blood flow [170], aortic blood flow [139, 171] and FSI problems [161]. Tables 3.2.2 and 3.2.3 demonstrate that POD-Interpolation ROM techniques typically accelerate by factors ranging from 10^2–10^6, while acceleration factors for POD-Projection ROMs range from 10^1–10^3. Wang et al. [165] compared POD-GP and POD-Interpolation approaches for steady-state heat conduction problems with different numbers of parameters. They found that the POD-GP approach was more reliable, with better performance as the number of parameters grew. POD-Interpolation may require more snapshots than POD-GP to achieve similar accuracy, so despite the faster evaluation times of POD-Interpolation, the overall offline cost to build a ROM of equal accuracy to the POD-GP ROM may be greater. Xiao et al. [166, 168] performed two studies comparing POD-GP with various POD-Interpolation techniques (Taylors, Smolyak, RBF interpolation). In both studies, the interpolation-based ROMs were found to be approximately one order of magnitude faster while maintaining good accuracy relative to the high-fidelity model.

Table 3.2.2: Various POD ROM papers for vascular flow and other applications.

Reference	**Method**	**Application**	**Accuracy**	**Acceleration Factor**
General applications				
Xiao et al. [147]	POD-Interpolation (RBF)	One-way FSI: Flow past a cylinder	-	727,000
		Two-way FSI: Free-falling square	-	73,200
		FSI: Bending beam	-	257,000
Hajisharifi et al. [167]	Local POD-Interpolation (RBF)	Fluidised bed time evolution	Eulerian field 98.9%, Lagrangian field 98.4%.	200,000
		Parametric fluidised bed	88.8% %	
Vascular flow applications				
McLeod et al. [149]	Atlas-based POD	ToF PA flow: case 1	$\sim 70\%$[*]	-
		ToF PA flow: case 2	$\sim 80\%$[*]	
		ToF PA flow: case 3	$\sim 50\%$[*]	
		ToF PA flow: case 4	$\sim 80\%$[*]	
Guibert et al. [150]	Atlas-based POD	ToF PA flow: patient 7	Δp 95.7%, outlet flow 96.0%	~ 1.33
		ToF PA flow: patient 13	Δp 93.9%, outlet flow 97.7%	
Buoso et al. [100]	POD-GP-DEIM	FFR calculation in coronary stenosis: case 1	FFR 98%, min. p accuracy 70%.	25
		FFR calculation in coronary stenosis: case 2	FFR 92%, min. p accuracy 90%.	
Ballarin and Rozza [161]	POD-GP	Fluid problem on moving domain	-	30
		Stationary FSI of parameterised idealised valve	-	16
		Unsteady FSI of parameterised channel	-	10
Ballarin et al. [151]	POD-GP	Coronary blood flow with varying physical and geometric parameters	> 99%	100
Ballarin et al. [169]	POD-GP	Coronary blood flow with varying physical and geometric parameters	> 99%	1530[†], 100[†]
Han et al. [170]	POD-GP	Aneurysm blood flow with varying PI	> 95%	2,410
Zainib et al. [172]	POD-GP	Coronary artery bypass grafts	> 99%	9[‡]
Girfoglio et al. [171]	POD-Interpolation (RBF)	Aortic flow with LVAD	p 99.5%, WSS 92.3%, u_x 91.5%, u_y 87.8%, u_z 88.6%	240
Girfoglio et al. [139]	POD-Interpolation (RBF)	Aortic flow with LVAD: case 1 (PF 3.45 l/min)	p 99.8%, WSS 95.9%, u_x 95.0%, u_y 92.2%, u_z 94.2%	7,200,000
		Aortic flow with LVAD: case 1 (PF 4.35 l/min)	p 99.5%, WSS 92.8%, u_x 90.3%, u_y 86.5%, u_z 90.7%	

[*] Maximum error estimated from graph in paper and used to calculate minimum accuracy (which occurs close to systole).

[†] Authors report computational savings of 99% (therefore acceleration factor of 100). 1530 acceleration factor is calculated from simulations times presented for ten patients in Table 2 of [169].

[‡] Mean acceleration calculated across three test cases in Table 1 of [172].

DEIM, discrete empirical interpolation method; FFR, fractional flow reserve; FSI, fluid-structure interaction; GP, Galerkin projection; LVAD, left ventricular assist device; p, pressure; Δp, pressure drop; PA, pulmonary artery; PI, pulsatility index; POD, proper orthogonal decomposition; RBF, radial basis functions; ROM, reduced order model; ToF, Tetralogy of Fallot; u_x, x-component of velocity; WSS, wall shear stress.

Table 3.2.3: ROM papers comparing POD-Projection and POD-Interpolation approaches for various applications.

Reference	**Method**	**Application**	**Accuracy**	**Acceleration Factor**
Xiao et al. [168]	POD-GP	Flow past a cylinder	-	10
	POD-I (Taylors)		-	260
	POD-I (Smolyak)		-	390
Xiao et al. [166]	POD-GP	Lock exchange	-	12
		Flow past a cylinder	-	10
	POD-I RBF	Lock exchange	-	496
		Flow past a cylinder	-	779
Wang et al. [165]	POD-GP	Four-variable heat conduction	99.81%	-
		Six-variable heat conduction	98.17%	-
	POD-I	Four-variable heat conduction	$>$99.99%	-
		Six-variable heat conduction	$\sim$ 50%	-

GP, Galerkin projection; POD, proper orthogonal decomposition; POD-I, POD-Interpolation; RBF, radial basis functions; ROM, reduced order model; .

Conclusion Both POD-Projection and POD-Interpolation approaches have been applied to nonlinear, geometrically complex, multi-physics vascular flow problems. It is possible to couple both of these approaches to 0D/1D models to capture multi-scale phenomena across large spatial scales in the vasculature. Geometric parameterisations can be incorporated into POD-based ROMs in an attempt to build models suitable for unseen geometries, but these models are limited in their generality and in the complexity of geometry they can model with a reasonable number of parameters. Attempts to build POD-based ROMs that are entirely general to geometry have seen either large errors [149] or minimal acceleration [150]. POD-based ROMs are often unsuitable for problems with large time scales, as the long-term stability of the POD modes is not guaranteed.

3.2.3 Dynamic Mode Decomposition

Dynamic Mode Decomposition (DMD) was originally developed by Schmid [173] for analysing spatio-temporal data from simulations and experiments. Modes are extracted from the data and can then be used to describe the physical mechanisms present in the data or for dimensionality reduction. For ROM construction, DMD can provide an alternative technique to POD for extracting leading-order modes from data. DMD trades the optimal reconstruction property of POD for physical interpretability, as the eigenvalue associated to each mode provides quantitative information on the oscillation frequency or growth/decay rate of the given mode [174].

Both DMD and POD utilise the SVD, but the difference arises in the construction of the snapshot matrix prior to performing SVD. In POD, the snapshot matrix is given by $\boldsymbol{U} = [\boldsymbol{u}_1 \dots \boldsymbol{u}_N]$. For DMD, the snapshot matrix is first divided into two submatrices, $\boldsymbol{U}_1 = [\boldsymbol{u}_1 \dots \boldsymbol{u}_{N-1}]$ and $\boldsymbol{U}_2 = [\boldsymbol{u}_2 \dots \boldsymbol{u}_N]$. The goal of DMD is to compute an approximation to the matrix $\boldsymbol{A}$, where $\boldsymbol{U}_2 \approx \boldsymbol{A}\boldsymbol{U}_1$ [175]. To do this, SVD is applied to $\boldsymbol{U}_1$ and the resulting decomposition is used to calculate the pseudoinverse of $\boldsymbol{U}_1$, which is then used to calculate $\boldsymbol{A}$. Thus, DMD finds a best-fit linear model that approximates the underlying time dynamics present in the data. In DMD, N will typically be a set of timesteps for the evolution of the system for one set of parameter values. Using the DMD model, an initial state can be propagated forward in time at a low cost. DMD ROMs are non-intrusive by being equation-free and entirely data-driven.

Since its inception, numerous extensions to DMD have been proposed to help tackle complexities such as nonlinearity, varying characteristic time scales in a given application, or handling externally driven data sequences. These extensions are thoroughly presented in Schmid [103]. Despite its growing use as a tool for analysing complex spatio-temporal data, DMD has seen limited application to vascular flow. Herein, the applicability of DMD and its extensions to modelling vascular flow are discussed.

Nonlinearity DMD aims to find an optimal linear model based on data. The underlying system in blood flow problems is nonlinear but the strength of this nonlinearity will vary depending upon the application. Habibi et al. [174] found that more DMD modes are required in an aneurysm model than in a stenosis model to achieve a particular reconstruction accuracy, highlighting the problem-specific nature of the complexity of vascular flow. In cases where nonlinearity is strong, a large number of measurements of the field of interest may be required to ensure the nonlinearity is captured in the reduced model. Extended DMD (EDMD) is an approach designed to help with this issue by using nonlinear functions of the measurements as input to the DMD algorithm [103, 176].

Geometric complexity Similarly to POD modes, DMD modes contain spatial information, so this approach is well-suited to constructing ROMs for individual complex geometries. Habibi et al. [174, 177, 178] have demonstrated the use of DMD to identify blood flow structures in cerebral aneurysms and stenosis models. However, as with POD, using DMD to evaluate flow fields in an unseen geometry is very challenging. DMD is a less well-established technique than POD, so few (if any) attempts have been made to tackle this problem.

Multi-physics DMD is suitable for multi-physics problems as the decomposition can be applied separately to each field. DMD can also be used to identify spectral coherence between each field in multi-physics applications, which can help to improve understanding of the problem. So far, the main use of DMD in multi-physics problems is to study FSI. Rodríguez-López et al. [179] used DMD to capture spatio-temporal evolution of flow over a flexible membrane wing using experimental data. They found that basic DMD could not reconstruct the fields accurately. Instead, they used high-order DMD (DMDho), developed by Le Clainche and Vega [180]. Where basic DMD only uses the previous snapshot, DMDho estimates each snapshot as a linear combination of a number of previous snapshots, thus improving performance in regimes where the FSI was stronger. This suggests that as the complexity of the system increases, accurate propagation of the time dynamics may require more than just the previous snapshot. This is worth considering when adding complexity (e.g. vessel elasticity, thrombosis models, device interactions) to vascular flow DMD models.

Multi-scale (time) DMD ROMs are perhaps most beneficial for problems of complex temporal nature. A DMD ROM is inherently designed to uncover time dynamics in a system and then propagate the reduced system forwards in time. Vascular flow is often modelled as periodic, with results from a single cardiac cycle taken to be representative of the flow for all time. This assumption can break down when autoregulation occurs or when complex long-term physiological phenomena, such as blood clotting, occur. The period of a cardiac cycle is roughly one second, whereas processes such as blood clotting can occur over a period of months. Multi-resolution DMD (DMDmr) provides a way to robustly separate complex systems into a hierarchy of multi-resolution time components [181]. DMDmr uses iteratively shorter snapshot sampling windows and recursive extraction of DMD modes from slow to fast time scales, which improves the predictions for short-time future states. This technique has been further generalised by Dylewsky et al. [182]. Provided with the appropriate data, DMDmr may be able to produce ROMs that can capture both long- and short-term effects of blood flow. Identifying a ROM for long-term effects (clotting, plaque build-up, etc.) may be particularly useful in reducing the cost of vascular models, as current approaches are too expensive to simulate these processes for the time scales over which they occur [20]. Another approach to handle complex temporal patterns is multi-stage DMD (mDMD) [174]. mDMD divides a temporal system into stages and applies DMD to each stage in turn. This allows more DMD modes to be used during periods with a more complex flow, while reducing the number of modes required when the flow is simpler, as demonstrated by Habibi et al. [174]. This approach can improve the efficiency of the ROM and reduce data storage requirements, but does not extend the original DMD method to more complex problems.

Multi-scale (space) DMD modes are local to wherever the high-fidelity data was generated, so using this approach for large regions of the vasculature is not possible without generating enormous amounts of high-fidelity data. However, DMD with control (DMDc) allows for input controllers to be integrated into the DMD algorithm. Habibi et al. [174] used inlet velocity as a controller for cardiovascular flow. It may be possible to extend this approach to account for other flow parameters or boundary conditions, thus allowing the inexpensive DMD ROM to be coupled to 0D/1D SDR models that account for the large-scale flow changes in the vasculature.

Summary Despite DMD being used as a ROM technique, very few papers directly compare the efficiency of the DMD ROM with the FOM used to generate the training data. Table 3.2.4 highlights a few studies that did evaluate the DMD ROM efficiency, showing speed-ups ranging from $\sim 10^0$–10^2. This acceleration seems small, but given the non-iterative equation-free nature of DMD ROMs, it is likely that they can provide more acceleration than this in some scenarios. Furthermore, Lu and Tartakovsky [183] included offline calculation times when determining the ROM speed-up, so higher acceleration values would be found if they only compared the online evaluation time with the FOM.

Few papers use DMD for vascular flow problems in the literature. Habibi et al. [174] used multi-stage DMD with control (mDMDc) to reveal hidden low-dimensionality in patient-specific blood flow in coronary stenosis and cerebral aneurysms. They found that mDMDc requires fewer modes than DMD to reconstruct the velocity fields to a given accuracy, but these modes were not used to construct a ROM. Habibi et al. [178] used DMD for data assimilation in Womersley flow, 2D idealised aneurysm flow and 3D real aneurysm flow, but in this instance the DMD analysis was not used to construct a ROM. Di Labbio and Kadem [184] performed POD and DMD analysis of left ventricular flow and found that while DMD requires more modes to achieve a particular energy level, it also preserves global particle advection using fewer modes. Another important point to consider when using DMD for vascular flow is that due to the periodic nature of the flow, unstable modes will either decay or grow over time, thus potentially under- and over-influencing the dynamics as time goes on [184].

Table 3.2.4: ROM papers using DMD for various applications.

Reference	**Method**	**Application**	**Accuracy**	**Acceleration Factor**
		General applications		
Bourantas et al. [185]	DMD	Tumour ablation treatment simulation	> 99.8%	∼ 13-37
Lu and Tartakovsky [183]	Lagrangian DMD	1D advection	-	0.21*
		1D advection-diffusion	-	581*
		1D inviscid Burgers equation	-	0.81*
		1D viscous Burgers equation	-	993*
	POD-GP	1D advection	-	0.15*
		1D advection-diffusion	-	84.2*
		1D inviscid Burgers equation	-	0.09*
		1D viscous Burgers equation	-	69.4*
Beltrán et al. [186]	DMDho-augmented FOM	1D Ginzburg-Landau equation	-	6-254†

* Authors include include offline calculation times (e.g. SVD) in DMD computational time, hence the ROM sometimes being slower than the FOM [183].

† Authors define speed-up as ratio of total simulation time to the sum of the time-lengths of the snapshots computational intervals, which is a particular definition suitable for their method [186].

DMD, dynamic mode decomposition; DMDho, high-order DMD; GP, Galerkin projection; FOM, full-order model; ROM, reduced order model; POD, proper orthogonal decomposition.

Conclusion DMD can be used to construct reduced order linear dynamical systems from data that approximate underlying nonlinear dynamics. DMD ROMs can be inexpensively propagated forwards in time or used to extract coherent structures from data.

DMD offers the benefit of having an associated frequency attached to each mode, thus providing interpretability (i.e. growth/decay/oscillation for each mode). DMD modes contain spatial information so this approach can be used to model individual complex geometries. DMD models are typically built with time as the only input parameter, so parametric DMD ROMs are rare; however, very recent work has begun to investigate this by adding interpolation into the DMD approach [187]. DMDc offers the potential to include input controllers into a DMD model, so this approach can be used to include the effects of, for example, varying inlet flow rate [174]. The input controllers could also potentially be boundary conditions derived from 0D/1D blood flow models, thus allowing DMD ROMs to account for larger portions of the vasculature. DMD can be applied to multi-physics problems, however a high-order DMD approach may be required to correctly reconstruct the fields of interest [180]. DMD ROMs are not commonly applied to vascular flow problems to date. A promising application of DMD in vascular flow is to problems where evaluating the long-term effects is not possible with conventional models. For these problems, DMD could perhaps be used to construct an efficient ROM for the time dynamics of long-term blood flow phenomena.

3.2.4 Other Techniques

There are various other ROM techniques that have not been as widely used as those discussed previously. Two of those techniques are discussed herein, the Reduced Basis (RB) method, which has seen some application to vascular flow problems, and the Proper Generalised Decomposition (PGD), which has not been applied to vascular flow modelling.

Reduced Basis

The RB method is usually applied to the fast solution of parameter-dependent problems [99, 188, 189]. Similarly to POD-based ROMs, the RB method utilises a set of snapshots of the FOM. Whereas POD uses the SVD to extract an optimal basis from the snapshots, the RB method is more general and can use various alternative approaches (e.g. Gram-Schmidt orthonormalisation [190]) to construct a basis spanning a sub-space of typically much lower dimension than that of the full-order solution manifold. RB methods often employ a greedy procedure for basis construction, whereby optimal snapshots are computed based upon an *a posteriori* error estimation [191]. A key advantage of the greedy approach is that the specific dynamics of the problem at hand guide the sample selection process [96]. Following basis construction, a Galerkin projection is often applied to build the ROM, similarly to POD-Projection ROMs.

The RB method has seen some application to vascular flow problems. Manzoni et al. [192] used this approach with radial basis functions for interpolating the geometric parameters to calculate flow fields in 2D parameterised carotid artery bifurcation geometries. For two test cases of global deformations of the carotid branches and stenosis near the carotid sinus, they achieved speed-ups of 96 and 88 times, respectively. Lassila et al. [193] applied the RB method to inverse problems in flow through stenosed arteries and in optimal shape design for femoropopliteal bypass grafts, reporting estimated speed-ups of 30–175 times. While effective in predicting downstream shear rates in the stenosis problem and in identifying optimal design configurations, the models were only applied to 2D steady-state problems. Colciago and Deparis [194] combined POD and the RB method, specifically the greedy algorithm, to build a ROM for a haemodynamics problem, noting CPU time gains of order 10^3. The application was to a femoropopliteal bypass problem, which was modelled using a 3D reduced FSI formulation, highlighting the suitability of the RB approach to multi-physics applications. The authors note that the greedy enrichment scheme can favour reducing the error in certain variables, especially when the quantities in the problem are of different orders of magnitude, so care should be taken in building an appropriate error estimator for multi-physics applications. Aside from vascular flow applications, the RB method has been applied to various other nonlinear Navier-Stokes problems [195, 196], including FSI problems [197]. Coupling the parametric RB method to boundary conditions derived from 0D vascular models provides a way to capture some multi-scale spatial effects.

Proper Generalised Decomposition

PGD generalises POD using separated representations while avoiding the need for any *a priori* knowledge about the solution [198]. Not utilising snapshot generation allows PGD to be applied to previously unsolved problems, which POD, DMD and RB ROMs are mostly incapable of. For a problem defined in space of dimension D, PGD provides an approximate solution u^N in the separated form

$$u^N(x_1, \ldots, x_D) = \sum_{i=1}^{N} F_i^1(x_1) \times \cdots \times F_i^D(x_D). \tag{3.2.10}$$

The PGD approximation is a sum of N functional products involving D functions $F_i^j(x_j)$ [199]. PGD solutions are constructed by successive enrichment, where a functional product F_n is determined using the functions from the previous $n-1$ steps. It should be noted that each enrichment step involves solving a nonlinear problem by means of a suitable iterative process. In PGD, both the number of terms N and the functions F are unknown *a priori*, making PGD an *a priori* ROM method. In a typical separation of variables, the coordinates x_i could be space and time coordinates, but in PGD additional coordinates can be included for problem-specific inputs such as boundary conditions or material parameters. Furthermore, if M nodes are used to discretise each of the coordinate spaces, the total number of PGD unknowns is $N \times M \times D$ instead of the M^D degrees of freedom found in standard mesh-based discretisations [199]. When the solution field is sufficiently regular, the number of terms N will be relatively small, highlighting how PGD overcomes the curse of dimensionality [200].

PGD was initially developed for solving time-dependent nonlinear problems in structural mechanics [201]. It has since been applied to rheology [202] and the incompressible Navier-Stokes equations [200]. Chinesta et al. [202] noted a speed-up on the order of 10^2 when using PGD for a transient rheology problem. Dumon et al. [200] found a speed-up of $\sim 10^2$ times for a 2D stationary diffusion problem, whereas a speed-up of 5–10 times was found for various Navier-Stokes problems, the most complex of which was a 2D lid-driven cavity flow. PGD has also been applied to multi-scale in time applications, where it is possible to separate the time dimension (1D in nature) into a multi-dimensional time space; however in this study the authors are not able to draw conclusions on the efficiency of the ROM [203]. PGD has also seen application to multi-scale in space and multi-physics problems, where the authors highlight that the savings due to PGD increase with problem complexity [204, 205]. Despite its potential usefulness in complex problems with known/unknown equations, PGD has not seen as widespread use as other reduced order techniques.

3.3 Accelerating Simulations with Machine Learning

Machine learning is a branch of artificial intelligence that excels at extracting underlying patterns in data. The basic building block of many machine learning algorithms is the neural network, shown in Figure 3.3.1. Neural networks consist of a collection of processing units, called neurons, and a set of directed weighted synaptic connections between the neurons. The connections between neurons symbolise the passing of information between neurons, with a fully-connected neural network (FCNN) meaning that all neurons in a given layer receive information from all neurons in the previous layer and pass information to all neurons in the subsequent layer. Each neuron processes the information it receives via some calculations and produces an output. The final layer is referred to as the output layer, where the final output of the network is produced. The FCNN in Figure 3.3.1 has two inputs, two hidden layers with four neurons per layer and one output. The objective of the network is to approximate a mapping between the input and output variables, given data to learn from. In vascular flow modelling, the inputs may be variables like space, time or Reynolds number and the outputs may be velocity, pressure or other variables of interest.

Each neuron is characterised by three functions: the propagation function, the activation function and the output function. The propagation function converts the vectorial input from the previous layer's outputs into a scalar input. The activation function quantifies the extent to which a particular neuron is active by applying a chosen function to the net input, such as the hyperbolic tangent or rectified linear unit functions [206]. Including activation functions for several sequential layers allows the deep network to approximate nonlinear mappings from inputs to outputs. The output function calculates the scalar output of a neuron based upon its activation state. Each neuron has a trainable weight associated to it, and each layer often has a trainable bias. These weights and biases are the network parameters that are optimised through training.

For a supervised learning problem, training data consists of a set of inputs with known outputs. During the training procedure, input data is passed through the network to give an output that is compared to the ground truth values for the output. A loss function is used to quantify the discrepancy between the network output and the ground truth output. The parameters associated to the network are optimised, typically through back-propagation and gradient descent algorithms, in order to minimise

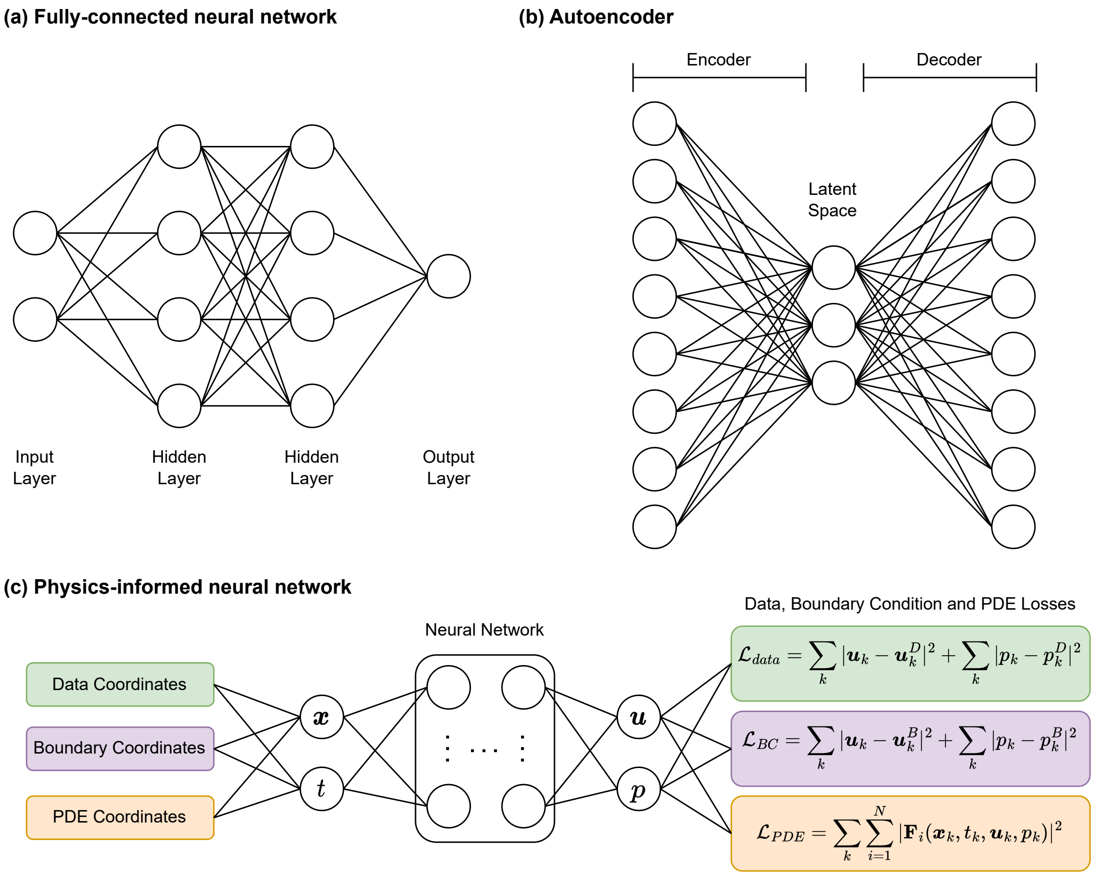


Figure 3.3.1: Selected neural network designs that can be used for simulation acceleration. (a) A fully-connected neural network with two inputs, two hidden layers with four neurons per layer and one output. (b) A fully-connected autoencoder, consisting of an encoder, a latent space and a decoder. (c) A physics-informed neural network, where physical constraints based on partial differential equations (PDEs) and boundary conditions (BCs) are included in the loss function of the network. $\boldsymbol{x}$ is position, t is time, $\boldsymbol{u}$ is velocity, p is pressure, superscript D or B means data or boundary point, $\mathbf{F}_i$ are residual equations.

the loss [207]. Once the network has been trained to accurately match predictions for the training data set, it can be used for input data where ground truth output values are unknown. Typically, the accuracy of the network will be assessed by evaluating its output on a data set that was not used in training, or through procedures such as cross-validation. A trained neural network can be considered to approximate a function that maps the input data to the output data. Hornik et al. [208] have demonstrated the approximation power of sufficiently large and deep networks. Variations on basic neural networks include autoencoders, convolutional neural networks (CNNs), recurrent neural networks (RNNs), and physics-informed neural networks (PINNs), amongst others [209].

Machine learning and deep learning have both employed neural networks to great effect in various classification and regression tasks in fields such as computer vision and natural language processing [210, 211]. Common across all learning-based strategies is the utilisation of data and the framework of an expensive up-front training stage preceding a cheap inference stage when evaluating the model for new data. In this way, machine learning approaches bear resemblance to ROM methods. A benefit of machine learning compared to ROMs is that the operations used in machine learning are highly parallelisable, which allows them to be trained and tested using highly parallel computing hardware, such as Graphics Processing Units (GPUs). This can reduce the time taken for training and inference, which is driving the growing interest in using machine learning-based simulation methods for acceleration.

Machine learning can be used in conjunction with ROMs, where the dimensionality reduction inherent to the ROM provides acceleration and machine learning is used to improve or replace some aspect of the ROM. For example, when constructing an interpolative ROM, such as in the POD-Interpolation method, using a neural network for interpolation can produce a ROM capable of outperforming POD-GP ROMs both in terms of acceleration and accuracy for certain applications [212–214]. Alternatively, machine learning can be used in place of conventional simulation methods to directly infer solutions fields or other quantities of interest from inputs such as medical images and point clouds of spatio-temporal coordinates [106, 107]. In this instance, the machine learning model itself provides acceleration relative to the full-order model, either through reduction of the dimensionality of the problem or through exploitation of parallel computing hardware.

Table 3.3.1: Machine learning ROM studies for various applications.

Reference	**Method**	**Application**	**Comments on accuracy and/or acceleration**
		ML-augmented ROMs	
Hesthaven and Ubbiali [212]	POD-NN	Parameterised steady-state PDEs (Poisson equation, LDC).	POD-NN achieves similar accuracy to POD-GP while reducing CPU time by 2-3 orders of magnitude.
Wang et al. [215]	POD-NN	Parameterised unsteady PDE (quasi-1D CVRC flow).	Accuracy of $\sim 99\%$ and acceleration factor of 10^7.
San et al. [213]	POD-NN (SN and RN approaches)	Viscous Burgers equation (time-dependent nonlinear wave propagation).	POD-NN approach outperforms POD-GP in interpolation and extrapolation and is 10^2 times faster.
Balzotti et al. [216]	POD-NN	Steady-state flow in a coronary artery bypass graft.	POD-NN achieves similar accuracy to POD-GP and speed-up of 10^6 and 10^4 relative to FOM and POD-GP, respectively.
Wang et al. [217]	LSTM-enhanced POD-GP	3D Stokes flow, 1D Kuramoto-Sivashinsky equation, 2D Rayleigh-Bernard convection.	ROM improves stability and accuracy of POD-GP for nonlinear problems and allows time predictions beyond training data.
Gao et al. [218]	FCNN-enhanced POD-GP	Nonlinear PDEs (1D viscous Burgers equation and 2D flame model).	ROM accuracy is $\sim 95\%$. ROM is more stable and accurate than POD-GP with DEIM (in the small basis limit).
		ML-based ROMs	
Maulik et al. [104]	CAE-LSTM	Viscous Burgers equation and shallow water equations.	CAE-LSTM has similar accuracy to POD-GP and is ~ 14 times faster.
Pant et al. [105]	3D CAE	2D flow (past a circular/square cylinder, over a plate, in a channel) and SST data.	Reconstruction accuracy is good and model can predict future timesteps accurately. Acceleration factor of 10^2.
Fresca et al. [219]	POD-enhanced CAE NN	Left ventricular cardiac electrophysiology.	POD-enhancement reduces training time from 15 hrs to 24 mins.
Fresca and Manzoni [220]	POD-enhanced CAE NN	Flow around cylinder, FSI of beam and laminar flow, cerebral aneurysm flow.	High levels of accuracy are displayed and acceleration factors are of order 10^5 for all applications.
Fresca and Manzoni [214]	POD-enhanced CAE NN	Flow past a cylinder.	POD-enhanced ROM has similar accuracy to non-enhanced DL-ROM. 10^5 acceleration factor relative to FOM.

CAE, convolutional autoencoder; CPU, central processing unit; CVRC, continuously variable resonance combustor; DEIM, Discrete Empirical Interpolation Method; DL, deep learning; FCNN, fully-connected NN; FOM, full-order model; GP, Galerkin projection; LDC, lid-driven cavity; LSTM, long short-term memory; ML, machine learning; NN, neural network; PDE, partial differential equation; POD, proper orthogonal decomposition; RN, residual network; ROM, reduced order model; SN, sequential network; SST, sea surface temperature.

3.3.1 Machine Learning Reduced Order Models

Machine Learning-Augmented ROMs Various attempts have been made to augment ROMs with machine learning. Neural networks (NNs) are adept at interpolation, so using them in POD-Interpolation ROMs is a natural choice. Hesthaven and Ubbiali [212] were among the first to apply a POD-NN ROM to parameterised steady-state PDEs (the Poisson equation and lid-driven cavity problems). In this model, the network approximates a mapping from the input parameter vector (including, e.g. material/geometry parameters) to the ROM coefficients. The POD-NN approach offers similar accuracy to POD-GP, while reducing computation time by two to three orders of magnitude. Wang et al. [215] extended the work by Hesthaven and Ubbiali [212] to time-dependent PDEs and applied it to a quasi-1D PDE problem. In this case, the time coordinate is included as an additional input to the neural network, allowing evaluation of the ROM at different timesteps. For the simple test problem, the authors found ROM accuracy of $\sim 99\%$ and an acceleration factor of order 10^7 relative to the FOM. San et al. [213] applied the POD-NN approach to the viscous Burgers equation to model time-dependent nonlinear wave propagation. San et al. [213] used a different network design to Hesthaven and Ubbiali [212] and Wang et al. [215], with San et al. [213] building a network that maps from the ROM coefficients at time t_n and any controllable input parameters (e.g. Reynolds number) to an output that characterises the ROM coefficients at time t_{n+1}. Within this framework, they present two variations: (i) a sequential network, where the outputs are the ROM coefficients, and (ii) a residual network, where the outputs are the residual between the ROM coefficients of t_{n+1} and t_n. Of these two approaches, the residual network is found to be superior and both approaches outperform POD-GP for the Burgers equation application. Balzotti et al. [216] applied the POD-NN approach to optimal control of steady-state flow in a patient-specific coronary artery bypass graft. The Reynolds number parameterised the inflow and was the single input parameter for which the ROM was constructed. The objective of the optimal control algorithm was to identify the normal stress that has to be imposed at the outlet to ensure a satisfactory agreement between the computed and clinically measured velocity fields. Online evaluation of the ROM took approximately 10^{-4} seconds, which is a speed-up of order 10^6 compared to the FOM. The POD-NN model was comparably accurate to a POD-GP model applied to the same problem, but the POD-NN ROM was four orders of magnitude faster [172].

It is also possible to augment POD-GP ROMs with machine learning. Two challenges in POD-GP ROMs are: (i) the potential lack of long-term stability and accuracy, and (ii) the lack of complete decoupling for nonlinear governing equation projection onto the reduced basis and the subsequent high cost of evaluating these nonlinear reduced operators. To address the first challenge, Wang et al. [217] used a long short-term memory (LSTM) network, a type of recurrent neural network designed to operate on sequential data. The POD coefficients are fed into the LSTM units and the physical/geometric parameters are fed into the initial hidden state of the LSTM. When applied to various problems (3D Stokes flow, 1D Kuramoto-Sivashinsky equation, and 2D Rayleigh-Bernard convection), the LSTM-POD-GP ROM is found to improve stability and accuracy compared to POD-GP for nonlinear problems. Furthermore, the LSTM ROM facilitates accurate predictions beyond the time interval of the training data. To address the second challenge, Gao et al. [218] proposed a non-intrusive approach to hyper-reduction that approximates the ROM velocity function using a FCNN. The FCNN-enhanced POD-GP ROM was applied to two nonlinear PDEs (1D viscous Burgers equation and 2D flame model) and found to be accurate to approximately 95%. The ROM was also shown to be more stable and accurate for the test problems than POD-GP with alternative hyper-reduction methods (DEIM), in the limit of a small basis. Another approach to improve accuracy is to use machine learning to adapt the ROM to a given input. Daniel et al. [221] used a deep classification network to recommend a suitable local POD-GP ROM from a dictionary of possible ROMs. This approach could be used in conjunction with small local ROMs, which have been shown to outperform a single global ROM in terms of accuracy and acceleration [167, 222].

Machine Learning-Based ROMs Dimensionality reduction is a crucial step in ROM construction and is commonly performed using techniques such as POD or DMD. Autoencoders (Figure 3.3.1) are neural networks used to compress and decompress high-dimensional data and are thus being increasingly used in the dimensionality reduction step in reduced models. Autoencoders can provide nonlinear data embedding, whereas POD and DMD offer only a linear reduced basis [104, 105]. This could allow autoencoders to compress complex nonlinear data more accurately than POD or DMD. Another approach that can offer nonlinear dimensionality reduction is manifold learning. Csala et al. [223] compared four manifold learning (locally linear embedding, kernel principal component analysis, Laplacian eigenmaps, isometric mapping) and two

ML-based (autoencoder, mode decomposing autoencoder) nonlinear dimensionality reduction methods to principal component analysis (PCA). They found that all six of the nonlinear dimensionality reduction methods achieved lower reconstruction errors than PCA for spatial reduction, but that only the autoencoder-based reduction was definitively superior for temporal reduction. Maulik et al. [104] used a ROM based on a convolutional autoencoder (CAE) and an LSTM to model the viscous Burgers equation and the inviscid shallow-water equations. In these advection-dominated systems, the deep learning (DL)-based ROM outperforms the POD-GP method. The CAE-LSTM approach is 14 times faster than the POD-GP method, producing errors of the same magnitude. Pant et al. [105] used a 3D CAE to compress simulation data and advance the solution in time without solving the Navier-Stokes equations in an iterative fashion. Using a 3D CAE allows for features to be extracted in both spatial and temporal axes, which mitigates the need for an additional network (e.g. an LSTM) for time propagation. Using this approach, the authors reduce computational run times by two orders of magnitude compared to traditional CFD solvers.

Fresca et al. [219] constructed a POD-DL-ROM that uses POD to reduce the dimensionality of the training data, improve training efficiency and reduce complexity. Compared to previous work by the same authors, enhancement with POD reduces the DL-ROM training time from 15 hours to 24 minutes. The DL-ROM itself uses CAEs and feedforward neural networks trained on the POD-reduced solution vectors. Fresca and Manzoni [214] used the same approach for a series of additional applications including an unsteady advection-diffusion-reaction system, a coupled PDE-ODE Monodomain/Aliev-Panfilov system, a nonlinear elastodynamics problem and the unsteady Navier-Stokes equations. For the most pertinent example, the Navier-Stokes problem, the acceleration factor was of the order 10^5 compared to the FOM while achieving a comparable accuracy to the more expensive non-enhanced DL-ROM. Fresca and Manzoni [220] used the same POD-DL-ROM for flow around a cylinder, FSI between an elastic beam and laminar flow, and blood flow in a cerebral aneurysm. High levels of accuracy are qualitatively displayed for each application. Acceleration factors for all applications are of the order 10^5. Essentially, the approach of Fresca and Manzoni [214], Fresca et al. [219], Fresca and Manzoni [220] reduces the size of the data passed through the network and the amount of training parameters required, thus improving the efficiency of training and testing while preserving the precision of the DL-ROM without POD enhancement.

Conclusion Machine learning (ML) has a lot to offer the ROM field, as demonstrated by the various studies in Table 3.3.1 that used ML in conjunction with ROMs. ML can be used to provide closure in projection-based ROMs, improve interpolation in POD-Interpolation ROMs, improve long-time ROM predictions, or offer alternative dimensionality reduction algorithms that are essential in almost all ROMs. ML-ROMs are able to address the weaknesses that hinder various reduced order methods, such as poor performance for nonlinear problems, lack of stability or lack of generality. As a result, ML-ROMs will typically be suitable for a wider array of vascular flow problems than the traditional ROM techniques from which they are derived. Balzotti et al. [216] demonstrated the superior acceleration capacity of a POD-NN ROM compared to a POD-GP ROM for a vascular flow problem due to the POD-NN approach being better-suited for the nonlinear nature of the problem. Similarly, Csala et al. [223] demonstrated the superior spatial reduction capability of nonlinear ML-based dimensionality reduction techniques when applied to aneurysm blood flow, which suggests that more accurate models may be possible using ML-based reduction techniques. Fresca and Manzoni [220] conversely used traditional dimensionality reduction techniques (POD) in conjunction with an ML-based ROM and achieved high levels of accuracy and acceleration for aneurysm blood flow. While not for vasclar flow applications, Wang et al. [217] and Gao et al. [218] augmented POD-GP ROMs with machine learning and achieved improved stability and accuracy. These findings demonstrate that ML-ROMs are a compelling option for vascular flow problems. In particular, ML-ROMs can offer methods suitable for vascular flow problems that are nonlinear, geometrically complex, multi-physics and multi-scale in time.

3.3.2 Physics-Informed Machine Learning Simulation

Machine learning can be used to construct fast surrogate models for vascular flow problems that directly predict haemodynamic quantities of interest, as in work by Itu et al. [107], Rutkowski et al. [224] and Liang et al. [225] (discussed further in section 3.3.3). A criticism of this approach is that the models do not guarantee the underlying physics in the problem will be respected. This can be somewhat resolved by incorporating known physics into the learning procedure [226]. The most widely used technique to achieve this are physics-informed neural networks (PINNs), which can combine data acquired from simulations or experiments with knowledge of the underlying governing equations and boundary conditions [106, 227]. In contrast to most machine learning simulation techniques, PINNs can be used in the absence of data. PINNs without training data may be less accurate than with data, but data-free PINNs offer a direct alternative to standard numerical techniques [228]. While PINNs were initially developed for solution and discovery of partial differential equations (PDEs) in forward and inverse scenarios, the development of data-free and parametric PINNs have since seen them applied to simulation acceleration. PINNs have been demonstrated to vastly reduce simulation times, particularly in the context of parametric design optimisation problems, hence the focus on this technique in the review [229, 230].

A typical PINN is shown in Figure 3.3.1. The PINN consists of a network with simulation parameters (e.g. space/time coordinates) as input and solution fields (e.g. velocity/pressure) as output. FCNNs are typically used for PINNs, but various other approaches have demonstrated superior results for certain applications [231]. For the chosen architecture, automatic differentiation is typically used to differentiate network outputs with respect to its inputs, thus acquiring derivatives such as u_x, p_x, u_t, etc., which can be combined to formulate governing equation residuals. For the incompressible Newtonian Navier-Stokes equations, the residual of the x-momentum equation will take the form:

$$F_1 = u_t + uu_x + vu_y + wu_z + p_x - \frac{1}{\mathrm{Re}}\left(u_{xx} + u_{yy} + u_{zz}\right), \tag{3.3.1}$$

where $\boldsymbol{u} = (u, v, w)$ is velocity, p is pressure and Re is the Reynolds number. Reduced Navier-Stokes equations (e.g. equation (3.2.3) for 1D blood flow) can also be used as residuals [232]. The residuals are included in the loss function for the network, which encourages the network to learn mappings that minimise the residuals and therefore satisfy the underlying governing equations. It is possible to enforce additional loss constraints that penalise the network for non-satisfaction of boundary conditions, such as the no-slip condition that is often applied on blood vessel walls. Alternatively, boundary conditions can be imposed as hard constraints through the network architecture [233]. Once trained, the PINN is able to infer solution fields that satisfy data, governing equations and boundary conditions.

PINNs are designed to improve the efficiency of non-informed networks through reducing the amount of data required and helping the network train efficiently by discarding non-physical mappings. A further benefit of PINNs is their potential to be used as an alternative to traditional numerical solvers. If data is unavailable, PINNs can be trained on PDE residual points and boundary conditions alone, mirroring the procedure for traditional numerical techniques. However, the input coordinates need only be a point cloud rather than the volumetric mesh required for typical numerical solvers. Furthermore, unlike traditional numerical solvers, when a problem has (removed: is ill-posed with) incomplete or noisy boundary conditions, PINNs are still a viable option [234]. A final benefit of PINNs is that they are well-suited to solving inverse problems as well as forward problems, whereas traditional numerical techniques are usually only suitable for forward problems.

Once trained, a PINN can quickly infer physics-respecting solution fields given spatio-temporal inputs, making them a promising acceleration technique. However, generalising a PINN for additional input parameters can decrease accuracy and increase training time, so the fast inference speeds must be balanced against training cost and accuracy. Despite their promise, PINNs are a relatively new technique for simulation and the application of PINNs towards acceleration and vascular flow is in its infancy. Three questions are addressed in order to determine the usefulness of PINNs for vascular flow acceleration: (i) How suitable are PINNs for simulation acceleration? (ii) How fast are PINNs relative to traditional numerical techniques? (iii) Are PINNs suited to the complexities of vascular flow acceleration?

Table 3.3.2: Various PINN papers that mention the acceleration capability of their method.

Reference	**Method**	**Application**	**Comments on accuracy and/or acceleration**
		General applications	
Hennigh et al. [230]	PINN	Heat sink design optimisation problem.	Total compute time is reduced by ∼45,000 times and ∼150,000 times compared to commercial and OpenFOAM solvers, respectively.
Arthurs and King [229]	PINN with active training	Parametric Navier-Stokes with parameters describing peak inflow velocity and tube shape.	PINN parameter sweep takes 7.6 s compared to 54 minutes for FEM. 400 times faster.
		Cardiovascular applications	
Gao et al. [218]	PI-CNN	SR of parameterised flow fields for idealised vascular problems.	Model accurately refines spatial resolution by 400 times and provides speed-up of 3364 times relative to CFD model.
Buoso et al. [235]	PINNs with RBF reduction	Left-ventricular biophysical modelling.	30 times faster than FEM including training (for evaluating only one condition). Accuracy for ejection fraction 97%, peak SBP 93%, stroke work 96%, myocardial strains 86%.
Sun et al. [228]	PINNs	Parametric flow in 2D idealised stenotic and aneurysmal vessels.	PINN evaluation is 2000 times faster than CFD model, but training takes hundreds of times longer than individual CFD simulations.

CFD, computational fluid dynamics; FEM, finite element model; PI-CNN, physics-informed convolutional neural network; PINN, physics-informed neural network; RBF, radial basis functions; SBP, systolic blood pressure; SR, super-resolution.

How suitable are PINNs for acceleration? Developing and using a PINN model often consists of three stages: (i) Generating or acquiring data from simulations or experiments (ii) Training the network whilst incorporating known physics and boundary conditions (iii) Using the model to infer solutions for new inputs. In inference mode, PINNs are usually faster than a traditional numerical model applied to the same problem. However, if the PINN relies on data generated by the numerical model and requires a potentially expensive training procedure prior to use, then the question of how to use PINNs for acceleration remains. In order to prove a useful and powerful tool for simulation acceleration, PINNs will either need to be able to generalise to unseen problems in a similar fashion to how parametric ROMs operate, or they will need to have a sufficiently small training time such that training a new PINN model is more efficient than solving a traditional numerical model.

Generalising a PINN model can require adding additional parameters into the training procedure. These parameters could describe geometry, boundary conditions, or material properties and there are various ways to incorporate this information into the PINN. The most straightforward approach is to include additional network input parameters. Arthurs and King [229] introduced two input parameters describing the peak inflow rate and diameter in a pipe flow problem. Sun et al. [228] similarly included parameters that describe geometry and viscosity as input to their PINN. When parameterising the network in this manner, an active learning strategy can reduce the cost of up-front data generation. This consists of refining the training data with additional finite element model (FEM) samples in regions of the parameter space where the PINN prediction is poor. Costabal et al. [236] used a positional encoding mechanism for PINNs that creates an input space for the network representing the geometry of a given object, improving PINN performance in complex geometries. However, for a Poisson forward problem in a simple domain, the positional encoding method was not observed to outperform traditional PINNs. de Avila Belbute-Peres et al. [237] developed a hyper-PINN approach, where an additional network is trained on sets of model input parameters (e.g. geometric parameters, boundary conditions, material properties) and network weights from previously trained PINN models for each simulation configuration. This precursor network learns how to map from the input parameter space to the weights needed for the PINN model for that particular parameter configuration. For a new parameter set, the precursor produces the weights needed to directly use the PINN in inference mode, thus bypassing the need to train a new PINN model entirely.

Alternatively to generalising PINNs, reducing training time sufficiently can mean that training a new PINN for each problem is still a tractable approach. Kissas et al. [232] suggested transfer learning to solve this problem. Transfer learning consists of intialising new PINN models with the parameters from a model previously trained on a similar problem, which can drastically reduce training time. This is similar to providing an accurate initial guess in iterative numerical methods. A transfer learning approach could allow for a new PINN to be trained for each new simulation configuration (new geometry, boundary conditions, etc.) while still providing an acceleration relative to solving the problem with traditional numerical techniques. For this approach to make sense, the new PINN must be trained without the use of training data from solving the numerical model. To this end, Desai et al. [238] proposed a one-shot transfer learning approach for PINNs, which consists of training for a selection of PDEs and then reusing some of the trained layers for an unseen PDE, thereby reducing training time. Another approach to accelerate training is to incorporate a hyper-parameter into the activation functions in the PINN [239]. The hyper-parameter dynamically changes the loss function topology throughout training and is shown to accelerate PINN convergence and increase accuracy. Residual-based adaptive refinement can also accelerate training [240, 241]. This approach aims to increase the number of network training points in regions where the PDE residual is inaccurate throughout training, thus accelerating convergence.

How fast are PINNs? Once the PINN training time is sufficiently reduced, or the network is generalised appropriately, the question of how fast PINNs are relative to traditional numerical techniques remains. Table 3.3.2 collates the literature on PINNs where the authors commented on the acceleration offered by their approach.

Arthurs and King [229] and Hennigh et al. [230] conducted design optimisation studies using PINNs. Arthurs and King [229] developed a parametric PINN model for Navier-Stokes applications and ran a parameter sweep experiment to identify the value of the geometric input parameter that would lead to a target pressure drop. This is a typical many-query problem, where repeated model evaluations are required to identify some kind of threshold in the output variable. The trained PINN required only 7.6 seconds to perform the sweep over 81 parameter points, whereas the same sweep using FEM would have taken 400 times longer. Scaling up the number of parameter queries to 1 million only increases the run time to 11.1 seconds, highlighting the scalability of the PINN due to its fast inference speed. However, it should be noted that the PINN evaluation was only performed at two spatial points, as this is all that is required to calculate the pressure drop. This demonstrates a benefit of PINNs, in that they can be used to query specific regions of interest, but the FEM model inherently evaluates the entire spatial field, so directly comparing model efficiency is not fair in this case. Hennigh et al. [230] presented NVIDIA SimNet, an AI-accelerated multi-physics simulation framework based on PINNs. They studied a design optimisation problem where SimNet is able to reduce total compute time by approximately 45,000 times compared to a commercial solver and 150,000 times compared to OpenFOAM. Gao et al. [242] trained physics-informed CNNs for super-resolution of low resolution flow field inputs using only knowledge of the conservation laws and boundary conditions. They applied this approach to 2D flow in a vascular domain and parametric super-resolution for internal flow with a parameterised inlet velocity profile. The model accurately refines the spatial resolution by 400 times for the flow fields with any new inlet BCs sampled in the 20-dimensional parameter space. The speed-up time for the trained model compared to the highly resolved CFD model is 3364 times. Sun et al. [228] used data-free parametric PINNs for flow in 2D idealised stenotic and aneurysmal vessels. They achieved accurate results in all test problems with mean test errors of order 10^{-4}–10^{-8} depending upon the problem and variable of interest. The authors noted that in the data-free PINN regime, implementing boundary and initial conditions with hard constraints improved performance when compared to the more widely used soft constraints. The trained PINN can be evaluated in 0.02 seconds, whereas the CFD model takes 40 seconds, yielding a speed-up of 2000 times. However, training the PINN took hundreds of times longer than an individual CFD simulation. The PINN will therefore only reduce total computational cost in scenarios where a large number of model evaluations are required, such as uncertainty quantification or design optimisation. Sun et al. [228] suggested that the speed-up offered by their approach will be increasingly advantageous when more complex applications are considered.

PINNs for vascular flow acceleration PINNs are inherently suited to nonlinear problems due to the nonlinear function approximating capacity of the network. In fact, the earliest applications of PINNs include nonlinear PDEs, such as the Navier-Stokes and Schrödinger equations [106]. Since then, PINNs have been successfully applied to various cardiovascular fluid dynamics problems, all of which are governed by the nonlinear Navier-Stokes equations [231, 232, 243–247].

Individual complex geometries are relatively straightforward to handle with PINNs. Instead of the usual volumetric mesh required for traditional numerical techniques, PINNs require only spatio-temporal coordinates as input and do not require connectivity between these points. Volumetric meshes may still be required in order to generate simulation data to train the PINN, but if the PINN is used to generalise across geometries, then users can forego the time-consuming meshing step for some of the geometries [228]. Raissi et al. [248] used PINNs to infer flow fields from concentration fields in an image-derived 3D aneurysm model and Sun et al. [228] applied PINNs with hard boundary condition enforcement to model flow in idealised stenosis and aneurysm models. This highlights two geometrically relevant applications of PINNs.

PINNs can also tackle multi-physics problems. Figure 3.3.1 shows a single-physics

PINN, but additional physics can be added by using a second network that maps from the same inputs as the first network (space and time) to different outputs (e.g. displacements and stresses for solid mechanics). It is therefore possible to calculate all the required derivatives in order to impose the governing equations and boundary conditions from each aspect of the multi-physics problem. This approach has been applied to an inverse Navier-Stokes and Cahn-Hilliard blood flow-thrombosis problem [246], multi-phase heat transfer [249] and FSI [250].

Basic PINNs are not commonly applied to extrapolating the associated PDE in time. Kim et al. [251] proposed a Dynamic Pulling Method (DPM) to overcome this issue. DPM manipulates the PINN's gradients to ensure the PDE's residual loss term continuously decreases during training. This is shown to improve extrapolation in time for various test problems. Basic PINNs are also not well-suited to problems spanning very large spatial regions. The issue with large spatial and temporal domains is that the domain can become arbitrarily large, leading to prohibitive training times. The primary approach to tackling these problems is incorporating domain decomposition into the PINN framework. Decomposing the large spatio-temporal domain into smaller sub-domains allows for sub-PINNs to be trained in each sub-domain. This improves training efficiency as well as reducing error propagation, allowing for domain-specific hyper-parameter tuning, increasing representation capacity and facilitating parallelisation [252].

Conservative PINNs (cPINNs), extended PINNs (XPINNs) and parallel-in-time PINNS (PPINNs) are three possible domain decomposition approaches that can tailor PINNS for multi-scale problems. cPINNs enforce conservation properties at spatial sub-domain boundaries using flux continuity and solution averaging across the interfaces [252]. XPINN is an extension to cPINN that applies to any type of PDE, not only conservation laws, and allows for decompositions in time and space [253]. Shukla et al. [254] compared cPINN and XPINN for a series of forward problems and found that for space decomposition, cPINNs are more efficient in terms of communication cost but that XPINNs are more flexible as they can handle time decomposition, a wider array of PDEs and arbitrarily shaped sub-domains. PPINNs are an extension to PINNs that mitigate the issue of long-time integration through time-domain decomposition and using a coarse-grained solver for long-time supervision [255]. The coarse-grain solver provides initial conditions for the PPINN in each time sub-domain. The coarse-grain solver needs be fast enough to solve the long-time PDE with some degree of accuracy cheaply, hence reduced-order or simplified models are viable options. Meng et al. [255] stated that the PPINN method could be extended to spatial domain decomposition, with a coarse-grained solver used to estimate the global solution and then a series of PINNs applied in parallel to spatial subdomains, thus increasing training efficiency relative to applying one PINN for the entire domain.

Conclusion PINNs offer a mixture of numerical mechanistic models and data-driven phenomenological models. Training a PINN model can be expensive compared to running a high-fidelity numerical model, so they are most useful for acceleration when a once-trained PINN can be used for numerous parameter or geometry instances. Various methods have been studied to parameterise PINNs [228, 229, 236, 237]. An alternative approach is to use PINNs in conjunction with transfer learning techniques to quickly retrain the model for a new system instance [238]. Employing techniques such as these can make PINNs a viable option for accelerating vascular flow simulations, particularly as PINNs (and extensions thereof) are well-suited to handling nonlinear, geometrically complex, multi-physics and multi-scale modelling problems.

3.3.3 Other Techniques

Given the relatively recent application of machine learning to simulation and the continued growth of the machine learning field, there are numerous other machine learning methods that have been or can potentially be applied to vascular flow acceleration. Reviewing them all in detail is beyond the scope of this study, and in most instances, there is insufficient relevant literature to do so, but herein several of these approaches are briefly discussed and how they may prove useful in the future for the target application is highlighted.

Physics-Agnostic Machine Learning Simulation

An alternative to augmenting/constructing ROMs using machine learning or attempting to encode physics into machine learning is to build a machine learning model that directly predicts the haemodynamic quantities of interest from inputs such as images or geometries [107, 224]. Some of these approaches are collated in Table 3.3.3. One of the earliest examples of this is by Itu et al. [107], who used a machine learning model

Table 3.3.3: Various machine learning simulation papers applied to vascular flow problems that mention the acceleration capability of their method.

Reference	**Method**	**Application**	**Comments on accuracy and/or acceleration**
		General applications	
Cai et al. [256]	DeepONet	Steady-state electroconvection.	Accuracy $> 99\%$. Acceleration factor $\sim 10^3$.
Mao et al. [257]	DeepONet	Coupled flow and finite-rate chemistry.	MSE is $\sim 10^{-5}$. Acceleration factor $\sim 10^5$.
		Cardiovascular applications	
Itu et al. [107]	FCNN	FFR prediction from coronary artery anatomy.	83.2% diagnostic accuracy for ischaemia. Acceleration factor > 80.
Liang et al. [225]	AE and FCNN	Steady-state haemodynamics prediction in thoracic aorta.	Velocity accuracy, 98.0%. Pressure accuracy, 98.6%. Acceleration factor ~ 900.
Morales et al. [258]	FCNN	ECAP prediction from LAA geometry.	Mean accuracy, 95.3%. Acceleration factor 144[*].
	FCNN with PCA		Mean accuracy, 94.8%. Acceleration factor 7,200[*].
Ferdian et al. [259]	Residual CNN	Super-resolution of aortic 4D flow MRI.	Flow rate prediction accuracy $> 95\%$. Prediction time 40–90 seconds.
Gharleghi et al. [260]	U-Net-style CNN	Transient WSS prediction in left main bifurcation of coronary arteries.	Accuracy $> 95\%$. Prediction time of 0.2 and 0.001 seconds with CPU and GPU, respectively[‡].
Li et al. [108]	Point-Net	Haemodynamics prediction before and after coronary artery bypass surgery.	Prediction accuracy $\sim 90\%$. Acceleration factor 600.
Li et al. [109]	Point-Net	Haemodynamics prediction before and after aneurysm treatment by FDS.	Prediction accuracy $> 87\%$. Acceleration factor 1,800.
Yin et al. [261]	DeepONet	Predicting damage progression and P-V curves in aortic dissection.	P-V accuracy $> 95\%$[†]. Prediction time is < 1 second, FOM simulation time is ~ 12 hours using 20 processors.

[*] 10-fold cross-validation used with 300 geometries. One round of cross-validation on 30 geometries took 30 seconds or 25 minutes for each model. This is used to calculate evaluation time for one geometry and compared to reported 2 hour CFD simulation time to calculate acceleration factors.

[†] P-V accuracy taken for test cases with damage included, from Table 3 of [261].

[‡] Network requires steady-state CFD result as input, which takes < 2 minutes to calculate. With this included, acceleration factor is ~ 90.

AE, autoencoder; CFD, computational fluid dynamics; CPU, central processing unit; DeepONet, deep operator network; ECAP, endothelial cell activation potential; FCNN, fully-connected NN; FDS, flow-diverting stent; FOM, full-order model; FFR, fractional flow reserve; GPU, graphics processing unit; LAA, left atrial appendage; MRI, magnetic resonance images; MSE, mean-squared error; NN, neural network; PCA, principal component analysis; P-V, pressure-volume.

to predict fractional flow reserve (FFR) given parameterised coronary artery anatomy as input. The model consists of a FCNN with inputs corresponding to features of the coronary anatomy and FFR as the solitary output. Using this approach, the authors achieved an accuracy of 83.2% in correctly diagnosing positive ischaemia and reduced model run time by a factor > 80.

Liang et al. [225] trained a deep neural network (DNN) to predict steady-state pressure and velocity fields in the thoracic aorta using 729 aorta geometries generated from a statistical shape model and CFD data generated for each geometry [262]. The DNN consisted of autoencoders to encode the aorta shapes and the fields of interest and another network to map between the encoded shapes and fields. The trained network predicted velocity and pressure fields with mean errors of 2.0% and 1.4%, respectively. DNN evaluation time is approximately one second, whereas each CFD simulation took approximately 15 minutes, giving a speed-up of $\sim$900 times. Liang et al. [263] applied this network structure to identifying the geometry corresponding to a particular pressure field, thus demonstrating an application of this method to inverse modelling. Morales et al. [258] applied two FCNNs, one with prior dimensionality reduction and one without, to predict endothelial cell activation potential (ECAP) from left atrial appendage (LAA) geometry. Their models were trained on 210 LAA geometries using CFD data. With and without dimensionality reduction, the average error was 5.8% and 4.7%, respectively. The network with dimensionality reduction was approximately 50 times faster than the other network when performing cross-validation. Gharleghi et al. [260] used a machine learning surrogate to replace a transient CFD solver in order to calculate wall shear stress (WSS) in the left main bifurcation of the coronary artery. The network requires the steady-state CFD solution for a given case as an input, but can then predict the transient WSS to an accuracy of $> 95\%$ within 0.2 seconds using a CPU and 0.001 seconds using a GPU. Rutkowski et al. [224] trained a CNN to map from 4D flow phase-contrast magnetic resonance images to highly resolved flow fields using CFD data as labels. The focus of this work was fast and accurate flow field generation directly from images, foregoing the need for time-consuming and expensive simulation set-up and execution. The network successfully de-noised flow images, improved velocity field accuracy and enhanced near-wall flow measurements. Ferdian et al. [259] similarly developed a residual network that was applied to super-resolution of 4D flow magnetic resonance images of aortic blood flow.

Their approach was able to predict flow rates in a real patient to greater than 95% accuracy within 40–90 seconds depending on the image size.

Various physics-agnostic machine learning simulation methods have been able to accurately and efficiently predict flow fields and flow-derived quantities in vascular flow applications. Provided that a FOM can be constructed and that sufficient data can subsequently be generated, the breadth of vascular flow problems that could be accelerated by these surrogate models is large. However, the vast amount of data required to generate accurate results could constrain these approaches, particularly in vascular flow applications where geometric data is typically derived from medical images that can be expensive to acquire and difficult to process. This is highlighted by Liang et al. [225], Morales et al. [258] and Gharleghi et al. [260] relying upon data augmentation strategies to extend their cohorts of real patients into larger cohorts of mostly synthetic patients. While this is necessary to create sufficiently large data sets, there is a risk that the augmentation may produce unrealistic results, as demonstrated by Morales et al. [258] discarding 30% of their initial training samples due to unrealistic flow features. It is possible that data augmentation approaches from the wider machine learning field, such as variational autoencoders or generative adversarial networks, could provide techniques to generate highly realistic synthetic data sets [264–266]. Another issue with physics-agnostic machine learning simulation methods is that the up-front cost of running CFD simulations in large cohorts to generate training data and the subsequent cost of training the complex network can lead to large overall costs. Despite these challenges, machine learning surrogate models are able to make predictions in previously unseen geometries due to being trained over an extensive array of different geometries. This is a crucial challenge in many vascular flow modelling problems that most acceleration techniques do not address with such generality.

Point Networks

Typical convolutional deep learning architectures require regular input data, such as images. Point-Net was developed to allow the direct use of irregular point cloud data with techniques typically applied to regular input data [267]. A benefit of using a Point-Net architecture is its ability to generalise well to new input point clouds. This means generalising to unseen geometries for vascular flow applications, which can lead to large savings in simulation times. Point-Net-based models have been applied to cardiovascular flow problems. Li et al. [108] used a Point-Net-based model to predict steady-state haemodynamics before and after coronary artery bypass surgery. Their approach yielded a prediction accuracy for velocity and pressure fields of around 90%. The time to evaluate the deep learning model was 600 times less than for the CFD model (1 second vs. 10 minutes), although 40 hours of training time was required prior to using the former. The same authors also applied their Point-Net-based model to predict steady-state aneurysm haemodynamics before and after treatment with a porous-medium flow-diverting stent model [109]. A similar prediction accuracy was found ($>$ 87%) and the calculation time was reduced by a factor of 1800. Kashefi and Mukerji [268] developed a physics-informed Point-Net (PIPN) and evaluated it for steady-state incompressible flow problems. The acceleration factor is approximately 35 for trained PIPN evaluation compared to the standard numerical solver. Compared to PINNs, the accuracy of PIPNs is similar when trained to the same convergence criterion, but the computational cost of PINNs is 18 times greater. This factor is increased when exploiting the inherent generalisation of PIPN to model new geometries, as in this scenario, the PINN will often need to be re-trained. PIPN is a recent technique that has not yet been applied to vascular flow.

Operator Networks

The function approximation capacity of neural networks is well known, but it is also possible for neural networks to approximate operators that map between function spaces [269]. The first and most general operator network is the deep operator network (DeepONet) [110]. DeepONet consists of a branch network, which encodes the input function space, and a trunk network, which encodes the domain of the output functions. The input to the branch network are function values at fixed sensors and the input to the trunk network are spatio-temporal coordinates at which to evaluate the operator. The output of the trunk network is a set of basis functions, and the output of the branch network is the basis coefficients [111]. Combining the basis coefficients and functions using the dot product gives the operator network output. Following training, the DeepONet approximates the underlying solution operator for the input function and coordinate spaces. Other operator learning methods include the Graph Kernel Network and Fourier Neural Operator [270, 271]. Physics-informed extensions to operator networks that can reduce the required training data have also been studied [111, 272].

Operator learning approaches have been applied to various linear and nonlinear problems involving explicit and implicit operators [110]. Cai et al. [256] used DeepONets for electroconvection, which is a multi-physics problem involving coupled flow, electric and concentration fields. They noted that training the DeepONets takes approximately 2 hours, but the evaluation time once trained is less than 1 second, representing a speed-up of approximately 1,000 times when compared to the NekTar solver used to generate training data. Mao et al. [257] used DeepONet for a hypersonic flow problem involving a coupling between flow and finite-rate chemistry. They found that the trained network was five orders of magnitude faster than the CFD solver used to generate the data. Furthermore, Cai et al. [256] and Mao et al. [257] combined multiple DeepONets to build a DeepM&MNet, which is specifically designed to handle multi-scale and multi-physics modelling. DeepONets have also been used as a surrogate for expensive microscopic models, thus accelerating the coupling between micro- and macro-scale models [273]. Recent work has also investigated using physics-informed DeepONets for long-time integration of parametric partial differential equations [274]. Applications of operator learning to vascular flow problems are limited, but two examples are by Yin et al. [261] and Arzani et al. [275]. Yin et al. [261] applied DeepONets to simulation of aortic dissection, a complex fluid-structure interaction problem. The DeepONet was able to make predictions in less than 1 second, whereas the finite element model used to produce training data took approximately 12 hours to run using 20 processors. Arzani et al. [275] applied an operator learning surrogate model to 2D cardiovascular flow applications, but the focus of this work was on the interpretability and generalisation rather than acceleration.

Compared to function-based learning strategies, a benefit of operator learning is that they demonstrate small generalisation errors [110]. Furthermore, DeepONets have been shown to overcome the curse of dimensionality, in that they do not require exponentially more training data to improve the approximation accuracy [276]. These techniques can potentially address many of the inherent complexities of vascular flow, particularly the multi-physics and multi-scale nature of the problem, but they have not yet seen widespread adoption.

3.4 Summary

This review presents simulation acceleration methods based on reduced order modelling (ROM) and machine learning for the target application of vascular flow. The review focuses on five complexities that are common in vascular flow problems, but which are also found across a multitude of other domains; namely: (i) nonlinearity, (ii) geometric complexity, (iii) multi-physics, (iv) multi-scale in time and (v) multi-scale in space. Each complexity presents unique challenges for vascular flow simulations and their acceleration. The ROM methods discussed in this review are spatial dimension reduction (SDR), proper orthogonal decomposition (POD) and dynamic mode decomposition (DMD) ROMs, as well as brief overviews of reduced basis (RB) methods and proper generalised decomposition (PGD). The machine learning approaches reviewed are machine learning-augmented ROMs, machine learning-based ROMs, physics-informed neural networks (PINNs), physics-agnostic networks, Point-Nets and operator networks. The review demonstrates that all acceleration methods are well-suited to some of the complexities of vascular flow and limited for others, as highlighted in Table 3.4.1.

3.4.1 Reduced Order Modelling

SDR methods are suitable for capturing spatial multi-scale behaviour and some nonlinear and multi-physics effects, but only in simplified geometries where axisymmetry or other assumptions are valid [115]. These methods calculate bulk quantities instead of full spatio-temporal fields and are not designed for temporal multi-scale problems. SDR methods are widely used in various vascular applications, with one of its most common uses being for deriving boundary conditions for 3D models [115, 133]. Due to their simplistic nature, SDR models can provide large acceleration ranging from two to six orders of magnitude [124].

POD-based ROMs branch into two categories depending upon whether they combine POD with projection or interpolation. POD-Projection and POD-Interpolation ROMs are able to calculate three-dimensional time-varying solution fields in individual complex geometries. POD-Projection has been applied to various vascular flow problems [100, 149–151, 161, 169, 170]. Both approaches are suitable for multi-physics problems. For nonlinear problems, the projection applied to the governing equations does not fully de-couple the ROM and the full-order model, limiting the acceleration offered by POD-Projection ROMs. POD-Interpolation does not depend upon the governing

Table 3.4.1: Reduced order modelling and machine learning acceleration methods and their suitability for modelling various vascular flow complexities. RB, PGD, physics-agnostic and Point-Net simulation acceleration approaches were briefly reviewed in this chapter but not in sufficient detail to include in this table.

Method	**Nonlinearity**	**Geometric Complexity**	**Multi-Physics**	**Multi-Scale (Time)**	**Multi-Scale (Space)**
ROMs					
SDR	✓	✗	∼	✗	✓
POD-P	∼	✓	✓	✗	✗[*]
POD-I	✓	✓	✓	✗	✗[*]
DMD	✓	✓	∼	✓	✗[*]
Machine learning-augmented ROMs					
POD-I-NN[†]	✓	✓	✓	∼	✗[*]
POD-P-NN[†]	✓	✓	✓	∼	✗[*]
Machine learning methods					
Physics-agnostic	✓	✓[★]	✓	✓[◇]	✓[◇]
PINN	✓	✓	✓	✓[‡]	✓[‡]
DeepONet	✓	✓	✓	✓	✓

Key: ✓, method is suitable; ∼, somewhat suitable; ✗, not suitable.

[*] In isolation the methods are not well-suited for spatial multi-scale problems, but they can be coupled to patient-specific SDR models so that boundary conditions are derived from large portions of the vasculature.

[†] Includes various types of NN used in conjunction with the ROM approach, such as FCNNs or RNNs.

[★] Physics-agnostic approaches are not only suitable for complex individual geometries, but are capable of generalising to previously unseen geometries.

[◇] While suitable for multi-scale problems in principle, the data-hungry nature of physics-agnostic approaches may make lead to prohibitive data requirements for problems spanning large spatial and time scales.

[‡] Basic PINNs are not designed for multi-scale problems, but extensions such as cPINNs, XPINNs and PPINNs are.

cPINNs, conservative PINNs; DeepONet, deep operator network; DMD, dynamic mode decomposition; FCNN, fully-connected NN; PGD, proper generalised decomposition; POD, proper orthogonal decomposition; POD-I, POD-Interpolation; POD-P, POD-Projection; NN, neural network; PINN, physics-informed NN; PPINNs, parallel-in-time PINNs; RB, reduced basis; RNN, recurrent NN; ROM, reduced order model; SDR, spatial dimension reduction; XPINNs, extended PINNs.

equations of the system, so it does not suffer the same limitations for nonlinear applications. However, POD-Interpolation ROMs have been shown to generalise less effectively than their projection-based counterparts [165]. Neither POD-Projection or POD-Interpolation are well-suited to multi-scale modelling in time, with the long-term stability of POD modes not guaranteed. Finally, while neither approach is inherently well-suited to spatial multi-scale modelling, coupling the POD-based ROM to an SDR ROM could produce a model that can quickly and accurately provide full spatio-temporal fields in a region of interest while capturing the influence of the systemic vasculature. Due to the non-iterative nature of POD-Interpolation, it can typically provide large accelerations ranging from two to six orders of magnitude, whereas POD-Projection acceleration ranges from one to three orders of magnitude [147, 166, 168, 169].

Similarly to POD-based ROMs, DMD ROMs can provide full spatio-temporal fields in individual geometries and could be coupled to SDR models to capture the influence of large regions of the vasculature. DMD ROMs are less common than POD-based approaches, so application to multi-physics simulation acceleration has not been thoroughly investigated. The main benefit to DMD ROMs is that they are designed to approximate the temporal dynamics of the system, which makes them well-suited to the long-time model integration required in temporal multi-scale problems.

Other techniques include RB methods and PGD. RB methods are a similar approach to POD-Projection ROMs and have been successfully applied to various nonlinear, multi-physics, geometrically complex problems [195–197]. RB methods have been applied to vascular flow problems such as flow field calculation in 2D parameterised carotid arteries, inverse modelling in stenosed arteries and flow in femoropopliteal bypass problems [192–194]. The acceleration offered by RB methods ranges from two to three orders of magnitude. PGD sits apart from most ROM methods, as it uses separated representations and successive enrichment *a priori* instead of applying dimensionality reduction to snapshots from the full-order model *a posteriori* in order to construct the reduced basis [198]. PGD has been applied to Navier-Stokes and rheology applications with acceleration ranging from one to two orders of magnitude [200, 202]. This approach is well-suited to separable problems, whether the separation is in space or time [199, 203, 204]; however it has not been applied as widely as other ROM methods and has seen no application to vascular flow simulation acceleration.

3.4.2 Machine Learning Simulation Acceleration

Machine learning offers an array of approaches for simulation acceleration. A common approach is to use machine learning in conjunction with ROM methods, where the learning algorithm augments or replaces part of the ROM method. Neural networks can be used to provide a powerful high-dimensional interpolation algorithm in the POD-Interpolation ROM approach [212, 213, 215] or to overcome the difficulties POD-Projection ROMs encounter for nonlinear equations [217, 218]. Autoencoders can also replace the dimensionality reduction that is common across most ROM methods [104, 105]. Another approach is to build a machine learning ROM based on autoencoders and feedforward neural networks whilst utilising POD for dimensionality reduction of the data passed to the machine learning ROM [214, 219, 220]. Machine learning can overcome some of the limitations of traditional ROMs and broaden the scope of problems for which the ROM methods are suitable.

PINNs are a machine learning-based simulation method that lies at the intersection of equation-based and data-driven modelling [106]. To be used for simulation acceleration, PINNs needs to be able to generalise across new input parameters and/or geometries or they need to be sufficiently fast to train that a new PINN can be constructed for each new problem instance. The former can be achieved by adding extra inputs to the network or by constructing a pre-cursor network that handles the parametric dependence in the problem [229, 236, 237]. Faster training times can be achieved through techniques such as transfer learning, trainable activation functions and residual-based adaptive refinement [232, 238, 240, 241]. When used in an acceleration context, such as many-query parameter sweeps, PINNs have been demonstrated to reduce total simulation time by two to five orders of magnitude, depending upon the application and the number of queries [229, 230]. PINNs and its extensions are suitable for all of the complexities that commonly occur in vascular flow problems and have been successfully applied to aneurysm flow modelling and synthesis of non-invasive flow measurements in a bifurcating vessel model [232, 248].

Alternative machine learning-based simulation techniques include physics-agnostic methods, Point-Nets and operator networks. Physics-agnostic simulation methods have been applied to vascular flow problems such as fractional flow rate prediction in coronary arteries, steady-state pressure and velocity prediction in the thoracic aorta, inverse geometry prediction in the aorta, endothelial cell activation potential prediction and prediction of flow fields from magnetic resonance images [107, 224, 225, 258, 263]. While these approaches can accelerate solution evaluations by two to three orders of magnitude and tend to generalise well to previously unseen geometries, they require large data sets and the network outputs do not necessarily respect the underlying physics in the problem. Point-Nets facilitate the use of powerful convolutional deep learning architectures on data sets consisting of point clouds. They have been used for steady-state haemodynamics predictions before and after coronary artery bypass surgery and aneurysm flow diversion, producing accurate predictions and reducing prediction time by two to three orders of magnitude compared to the computational fluid dynamics model [108, 109]. Point-Nets generalise well to new geometries despite paying no attention to underlying governing equations, but require large data sets for training. Physics can inform Point-Nets, but this is a new technique with very few use cases to date [268]. Operator learning techniques, such as DeepONets, are another powerful simulation technique that have demonstrated strong generalisation capabilities, the ability to accelerate by two to five orders of magnitude, and the ability to overcome the curse of dimensionality [110, 256, 257]. However, operator learning is an emerging technique that has only seen a small number of applications to vascular flow problems to date [261, 275].

CHAPTER 4

In-Silico Flow Diverter Performance Assessment in Posterior Communicating Artery Aneurysms

Abstract

In-silico trials use clinical data and mathematical modelling to evaluate the performance of medical devices in cohorts of virtual patients. Utilising virtual patients allows in-silico trials to test devices in scenarios and demographic groups that may be underrepresented in clinical trials. This chapter presents the FD-PComA in-silico trial into flow diverter (FD) treatment performance in posterior communicating artery (PComA) aneurysms. PComA aneurysm flow diversion has not been studied in clinical trials to date and is further complicated by the presence of fetal posterior circulation (FPC). FPC has an estimated prevalence of 4–29% and is more common in people of black (11.5%) than white (4.9%) ethnicity, the former of which can be underrepresented in clinical trials. Therefore, FD-PComA demonstrates the use of in-silico trials to provide insight in lesser studied scenarios (FPC) and demographics (black ethnicity).

In the FD-PComA trial, flow diverter treatment was modelled in 118 virtual patients with 59 distinct anatomies, using computational fluid dynamics to assess post-treatment outcome. Boundary conditions were prescribed to model the effects of non-fetal and fetal posterior circulation (FPC), allowing for comparison between these subgroups. FD-PComA predicted reduced treatment success in FPC patients, with an aneurysm space-and-time-averaged velocity reduction of 67.8% across non-fetal patients and 46.5% across fetal ($p < 10^{-11}$). Space and time-averaged wall shear stress on the device surface was 29.2 Pa in fetal patients and 23.5 Pa in non-fetal ($p < 0.05$), suggesting flow diverter endothelialisation may be hindered in FPC patients. The study found that morphological variables such as the size and shape of the aneurysm and PComA size did not affect the treatment outcome.

4.1 Introduction

In-silico trials (ISTs) are an alternative to clinical trials for generating evidence of medical device or drug safety and efficacy. In-silico methodologies are being adopted for drug [14, 15] and medical device development [277] in the treatment of various pathologies, such as diabetes [16], acute ischemic stroke [17, 18] and COVID-19 [19]. In clinical trials, some treatment scenarios may be unfeasible or unethical to impose in real life patients and demographic biases due to study type, location or other selection criteria (e.g. age) may be unavoidable [278]. ISTs can be used to generate evidence in scenarios that are difficult to assess in clinical trials and/or in demographics that may be less well-represented.

Previous work has established that ISTs for flow diversion of intracranial aneurysms can replicate and expand upon results from clinical trials of intracranial aneurysm flow diversion [20]. Flow diversion is an established treatment method for intracranial aneurysms, with various devices available on the market. The Pipeline Embolization Device (PED), produced by Medtronic, is the most widely used flow diverter but it is not FDA-approved for use in posterior communicating artery (PComA) aneurysms, which is one of the most common aneurysm locations [49]. Flow diversion aims to promote stasis-induced thrombosis in the aneurysm sac and endothelial proliferation of the neointima along the device itself, ultimately leading to aneurysm occlusion. Safety concerns regarding branch vessel occlusion led the FDA to not approve the use of the PED in PComA aneurysms, but studies since then have consistently found no neurological deficits following the treatment (i.e. the safety of the treatment has been demonstrated) [279–281].

PED efficacy in the treatment of PComA aneurysms in certain scenarios remains in question despite its safety being established. Fetal posterior circulation (FPC) is defined by absence (true fetal) or atrophy (fetal-type) of the P1 segment of the posterior cerebral artery (PCA) [282]. In these scenarios, the PComA is required to provide some or all of the blood supply to the P2 PCA. The increased demand on the PComA leads to increased flow rates and potentially subsequent PComA hypertrophy, both of which could influence PED performance in the treatment of PComA aneurysms. The incidence of the fetal PComA variant is 4–29% unilaterally and 1-9% bilaterally [283]. FPC is more common in patients with PComA aneurysms than those without [284]. FPC is also more prevalent in patients of black ethnicity than white [285]. Multiple case studies [286, 287] have suggested treating fetal-type PComA aneurysms with alternative methods to the PED. A retrospective review of 49 patients by Rinaldo et al. [2] found that PED occlusion rates were 43.7% for patients with fetal-type PComA aneurysms and 81.8% for patients without the fetal-type variant. This review defined FPC as a PComA diameter greater than that of the P1 PCA, but increased flow rates are also a characteristic of FPC. This poses the question of whether it is the PComA diameter or the increased flow that hinders PED treatment [4].

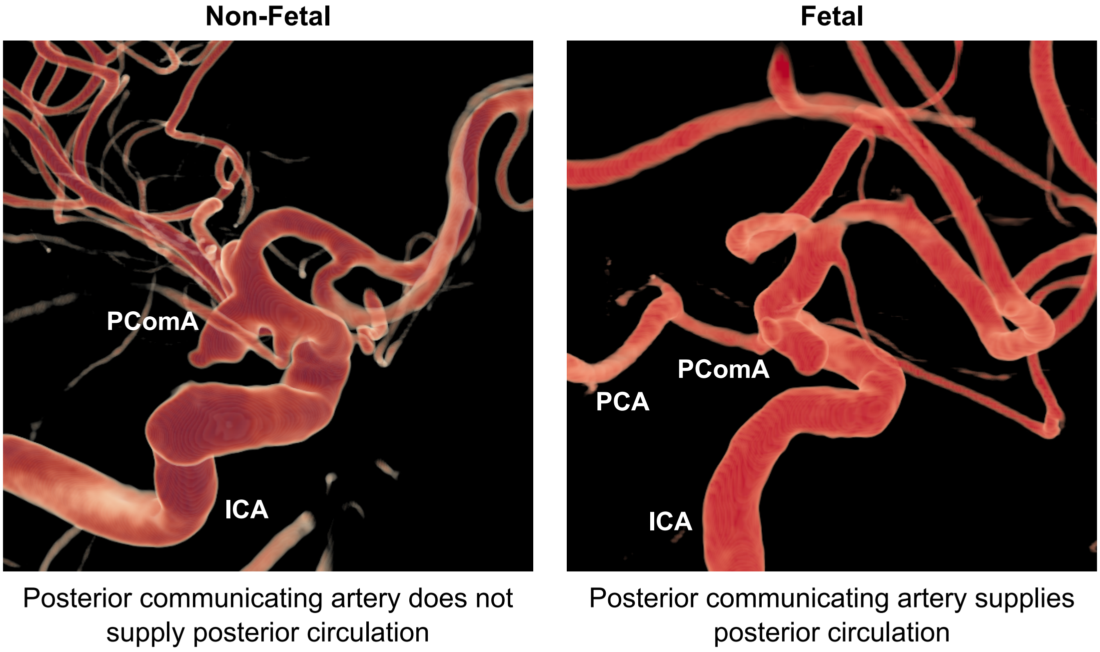
Prospective clinical trials into PED treatment have not focused on PComA aneurysms to date, despite them accounting for roughly 25% of all intracranial aneurysms [283]. The PUFS (Pipeline for Uncoilable or Failed Aneurysms [288–290]), PREMIER (Prospective Study on Embolization of Intracranial Aneurysms With Pipeline Embolization Device [291]) and ASPIRe (Aneurysm Study of Pipeline in an Observational Registry [292]) prospective clinical trials into PED flow diversion collectively contained 23/456 (5.0%) PComA aneurysms. The IntrePED (International Retrospective Study of the Pipeline Embolization Device [293]) retrospective study contained 61/906 (6.7%) PComA aneurysms. None of these studies reported specific findings for PComA subgroups. PComA aneurysms with FPC have similarly not been specifically reported on in large-scale clinical trials.

The aim of this work is to perform an IST of PComA aneurysm PED flow diversion and to investigate the impact of FPC on device performance. Herein, I present the FD-PComA IST, which consists of the following steps: (1) Select PComA aneurysms from the @neurIST and AneuX studies [294, 295]; (2) Simulate PED deployment using the algorithm presented in [296]; (3) Generate volumetric lumen meshes pre- and post-treatment; (4) Apply patient-specific time-varying boundary conditions for normotensive inflow conditions [91]; (5) Apply patient-specific outflow conditions for non-fetal and fetal posterior circulation; (6) Simulate transient haemodynamics pre- and post-treatment; (7) Extract haemodynamic markers for aneurysm flow reduction and endothelialisation on the device struts and examine correlations between non-fetal/fetal flow conditions, aneurysm morphology and PComA size.

This work is an exemplar IST demonstrating the application of computer modelling and simulation to generating evidence for lesser studied scenarios (FPC) and demographics (FPC prevalence varies across demographics). This IST also targets an off-label use of the PED flow diverter and as such represents the largest scale study, clinical or otherwise, into PED treatment of PComA aneurysms [50].

The chapter is outlined as follows. Section 4.2 provides background on fetal posterior circulation. Section 4.3 details the methodology, including the IST design (4.3.1), the simulation pipeline (4.3.2), and the automation of the pipeline (4.3.3). Section 4.4 presents results from the IST and Section 4.5 draws conclusions. Discussion for the chapter can be found in the thesis discussion (Section 6.2).

4.2 Fetal Posterior Circulation



Posterior communicating artery does not supply posterior circulation

Posterior communicating artery supplies posterior circulation

Figure 4.2.1: Example of fetal and non-fetal anatomies for two patients from the trial cohort.

Figure 4.2.1 shows images of fetal and non-fetal patients from the @neurIST database. In the fetal case, the PComA is larger and supplying blood to the posterior circulation. In the non-fetal case, the PComA is less visible in the image. This is due to less contrast agent being drawn into the PComA as it supplies no distal vessels and therefore has a reduced flow rate. These images highlight the two factors that could reduce flow diverter efficacy in PComA aneurysms in patients with FPC - the increased flow rate and the increased PComA size.

Fetal posterior circulation (FPC) is estimated to occur in 4-29% of the population [283] but its prevalence can vary in different demographics. Shaban et al. [285] retrospectively reviewed 532 PComA aneurysms and provided statistics for the prevalence of PFC in a number of ethnicities. This data was collated (4.2.1) to show that full FPC prevalence is significantly lower in white patients (4.9%, 8/164) than black patients (11.5%, 40/349, $p = 0.008$). The FD-PComA in-silico trial therefore not only generates evidence in less-studied scenarios (i.e. PComA aneurysms with FPC), but also provides insights that are beneficial to demographic groups that can be under-represented in clinical trials (i.e. black ethnicity) [297]. This demonstrates the capacity of in-silico trials to generate new evidence in medical device testing and to improve health equity.

Table 4.2.1: Fetal posterior circulation prevalence for different ethnicities.

Reference	FPC Classification	Ethnicity		
		White	Black	Hispanic
Shaban et al. [285]	Fetal, % (n/N)	4.9 (8/164)	11.5 (40/349)	0 (0/12)
	Partial fetal, % (n/N)	15.2 (25/164)	14.9 (52/349)	16.7 (2/12)
	Fetal or partial fetal, (n/N)	20.1 (33/164)	26.4 (92/349)	16.7 (2/12)
	Non-fetal, % (n/N)	79.9 (131/164)	73.6 (257/349)	83.3 (10/12)

FPC, fetal posterior circulation.

4.3 Methodology

4.3.1 In-Silico Trial Design

Hypotheses and sub-analyses FD-PComA investigated the following hypotheses:

1. Maintained PComA flow following PED flow diversion reduces treatment success (measured as post-operative aneurysm flow reduction) in patients with FPC.

2. PComA radius influences PComA aneurysm treatment success to a greater extent than aneurysm morphology (characterised by maximum diameter, aspect ratio, neck diameter and non-sphericity index).

In addition to flow reduction, endothelialisation plays an important role in flow diverter treatment. Therefore, the influence of FPC on endothelialisation was also investigated, using high wall shear stress on the device as a haemodynamic marker for inhibited endothelialisation. Similarly to flow reduction, the influence of morphological parameters (PComA radius, aneurysm maximum diameter, aneurysm aspect ratio, aneurysm neck diameter and aneurysm non-sphericity index) on endothelialisation was analysed.

Power calculation Based on a retrospective study, PED occlusion rate for PComA aneurysms is thought to be 43.7% for patients with FPC and 81.8% for patients without [2]. Considering a type I error of 0.05 and a power of 90%, the number of subjects required to observe this disparity in treatment success in our trial is 64, with 32 from each sub-group (FPC and non-FPC).

Inclusion criteria and virtual patient cohort The inclusion criteria in the trial were that the patient must only have one aneurysm and that the aneurysm must arise from the PComA. Patients were selected from the @neurIST [298] and AneuX [295] cohorts. A number of cases were initially processed directly from @neurIST images. The @neurIST cohort consists of 3D rotational angiography images for 800 patients, 143 of which are for PComA aneurysms. From these images, 29 were selected and automatically segmented using an algorithm developed by Lin et al. [299]. Following segmentation and subsequent surface mesh processing and device deployment steps, 13/27 patients were added to the trial. The reasons for failure in these 14 cases were: (i) absent or poor quality PComA following segmentation (10/14), (ii) multiple aneurysms present (1/14), (iii) surface mesh processing (1/14) and (iv) device deployment (1/14). The AneuX cohort contains surface meshes for 668 vessels and 750 aneurysm domes gathered by processing patients from the AneuX1, AneuX2, @neurIST and Aneurisk data sets. Of the 130 PComA cases in AneuX, 60 were randomly selected, ensuring no duplication with the successful cases already processed from @neurIST images. Eventually, 14/60 cases were omitted due to issues with: (i) geometric configuration e.g. multiple branch vessels, PComA absent/small/stenosed/mislabelled (8/14), (ii) aneurysm not arising from the same side of the internal carotid artery (ICA) as the PComA (2/14), (iii) surface mesh quality (2/14) and (iv) device deployment (2/14). In total, 59 cases were included in the FD-PComA trial, which exceeds the number required to achieve statistical power in the results. Details of the cohort can be found in Table 4.3.1.

Table 4.3.1: FD-PComA in-silico trial cohort characteristics for the 59 distinct anatomies included. The demographic and morphological characteristics were identical in the non-fetal and fetal patients.

FD-PComA in-silico trial	
Number of virtual patients	118
Number of distinct geometries	59
Aneurysm location	
PComA, % (n/N)	100 (59/59)
Non-fetal posterior circulation, % (n/N)	50 (59/118)
Fetal posterior circulation, % (n/N)	50 (59/118)
Female sex, % (n/N)	74.6 (44/59)
Male sex, % (n/N)	25.4 (15/59)
Age, years	
Mean ± SD (N)	53.4 ± 10.9
Median	53.0
Range	22.0–78.0
Aneurysm size, mm	
Mean ± SD (N)	6.0 ± 3.3
Median	5.0
Range	1.8–18.9
Aneurysm neck, mm	
Mean ± SD (N)	3.8 ± 1.5
Median	3.5
Range	1.6–8.8
Aneurysm aspect ratio	
Mean ± SD (N)	1.3 ± 0.5
Median	1.3
Range	0.4–2.9
Aneurysm non-sphericity index	
Mean ± SD (N)	0.18 ± 0.07
Median	0.17
Range	0.03–0.32
PComA diameter, mm	
Mean ± SD (N)	1.5 ± 0.5
Median	1.5
Range	0.5–2.2

FD, flow diverter; PComA, posterior communicating artery; SD, standard deviation.

In-silico trial end points and other metrics Clinical trials for flow diverters typically use endpoints such as neurological morbidity and mortality, intracranial haemorrhage and ischaemic stroke as measures of safety, and angiographic occlusion rate at 6- and 12-month follow up as measures of efficacy. While it is not possible to simulate long-term treatment response currently, several studies suggest that post-treatment flow diversion is an appropriate proxy haemodynamic metric for predicting aneurysm occlusion [300–302]. Furthermore, Sarrami-Foroushani et al. [20] demonstrated that an in-silico end point of 35% reduction in aneurysm mean velocity led to accurate recreation of existing clinical trial results in a cohort of 82 ICA and PComA aneurysms. For this reason, a 35% reduction in aneurysm space-and-time-averaged velocity (STAV) was also considered as a haemodynamic surrogate end point for angiographic occlusion in this study. As well as reducing the mean aneurysm velocity, flow diverter treatment aims to reduce the maximum inflow velocity through the neck to minimise the impact of impinging jets [303]. As such, reduction in maximum time-average velocity (MTAV) at the aneurysm neck has also been used as a haemodynamic proxy for successful occlusion [20]. This variable was calculated in addition to STAV and used as another metric to assess aneurysm occlusion.

As well as stasis-induced aneurysm thrombosis, endothelial cell growth along the device plays an important role in flow diverter treatment of aneurysms [304]. Wall shear stress (WSS) levels along the device struts have been shown to indicate the pattern of neointimal growth on the device [305, 306]. Increased WSS was found in regions that remain patent, whereas longitudinal proliferation of neointimal cells was found in regions of low WSS. Therefore, space-and-time-averaged WSS (STAWSS) on the device surface was used as an indicator for treatment performance. However, there are insufficient clinical or computational studies into this metric to formulate a suitable end-point to distinguish successful or unsuccessful treatments.

4.3.2 Semi-Automatic In-Silico Trial Simulation Pipeline

Image segmentation The @neurIST database contains 3D rotational angiography images for each patient. These images were automatically segmented using a multi-task convolutional neural network and a patch-based learning pipeline designed to segment both vessel and aneurysm simultaneously [299]. In the AneuX database, surface meshes for vessel and aneurysm are provided and segmentation was not necessary.

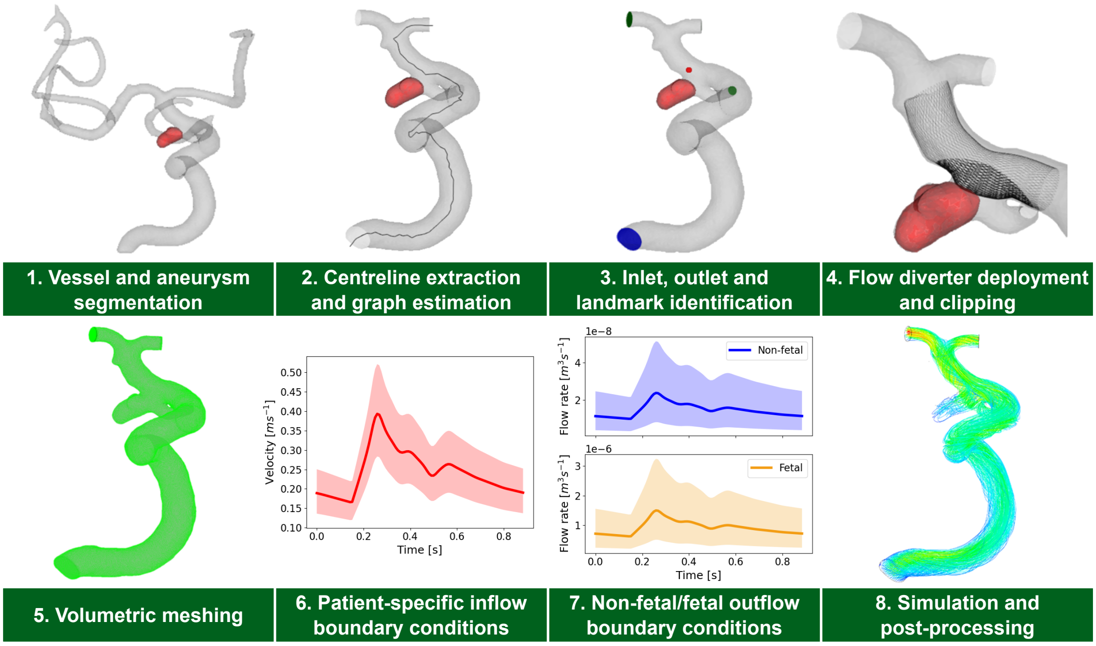


Figure 4.3.1: Simulation pipeline.

Surface mesh pre-processing Following automatic segmentation of the 3DRA images, the segmented masks were processed to prepare vascular surface meshes for subsequent volume meshing. Centrelines were extracted using Laplacian mesh contraction and spectral clustering. An undirected graph of centreline points was constructed by exhaustive recursive path finding. The graph was used to identify bifurcation and terminal points in the vasculature and the segmented aneurysm mask (@neurIST) or sac mesh (AneuX) was used to identify an aneurysm landmark point on the centreline. The aneurysm landmark point was later used to position the flow diverter during device deployment. The ICA inlet landmark point was identified by calculating the vessel radii at all terminal centreline points and assuming that the largest radius corresponds to the inlet. The ICA-PComA bifurcation point was found by assuming the path length to this bifurcation is shorter than the path to any distal bifurcations. The middle cerebral artery (MCA) and anterior cerebral artery (ACA) bifurcation is assumed to be the second shortest path from the inlet to the bifurcation points. Manual surface corrections were applied where required, particularly to remove small vessels for which appropriate boundary conditions were not known, such as the anterior choroidal artery. The PComA, MCA and ACA were clipped or extruded as required to ensure that they had a comparable length while retaining as much of the PComA as possible. These steps produced surface meshes for the inlet (ICA), three outlets (PComA, MCA, ACA), and vessel (including the aneurysm).

Flow diverter deployment All patients were treated with a single PED flow diverter. Each PED consisted of 48 wires with a 30 μm thickness and was deployed using a fast virtual stent (FVS) placement method, which was presented and validated by Larrabide et al. [296]. The deployment algorithm acts on a simplified simplex mesh that describes the stent. To ensure the deployment is realistic, the geometrical characteristics of the stent are used to guide the deformation. These characteristics include the stent design (strut pattern), the strut length, the angle between the struts and the deployed stent radius. Using this information and the geometry of the vessel in which the device is being deployed, the algorithm expands the stent until it reaches a minimal energy state when the struts are in close proximity to the vessel walls. A limitation of this approach is that the stent can bulge into the aneurysm, as there is no vessel wall at the aneurysm neck to restrain the expansion of the device. For this reason, manual checks were performed to check that there was minimal bulging of the deployed stent through the neck surface. Each flow diverter was positioned using the aneurysm location landmark and flow diverter diameter was selected using the mean of the parent vessel radii proximal and distal to the aneurysm landmark. To reduce computation costs, the flow diverter was clipped in the vicinity of the aneurysm. This has been shown to have negligible effect on intra-aneurysmal haemodynamics [307]. When clipping, the portion of the flow diverter covering the aneurysm neck and the branch PComA vessel was retained.

Volumetric meshing Following surface mesh pre-processing and device deployment, volumetric meshes were generated using ANSYS ICEM CFD v19.1 (Ansys Inc., Canonsburg, PA, USA). Tetrahedral elements with maximum edge size of 0.5 mm were used to discretise the computational domain. A maximum edge size of 0.2 mm was used on the vessel wall and 0.05 mm was used on the inlet and outlet surfaces. Where PED was present, a maximum edge size of 0.01 mm was set on the wires. This led to an average number of mesh elements across all geometries of approximately 1 million without the device and 14 million with. Mesh independence of the solutions at these resolutions was verified by Sarrami-Foroushani et al. [23]. Images of the surface meshes for one case with and without the device are presented in Appendix A.2.1. These surface meshes demonstrate the difference in mesh resolution required in the vicinity of the device due to small length scale of the struts.

Inflow boundary conditions Meshed anatomies were combined with virtual physiological flow conditions to create virtual patients. Flow conditions were applied through a patient-specific inlet flow waveform generated using a Multivariate Gaussian Model (MGM) [21, 91]. For each patient, a waveform representing rest/normotensive conditions was produced. Normotensive ICA flow waveforms were taken as the mean of the MGM, which was trained and calibrated by patient-specific phase-contrast magnetic resonance imaging (MRI) measurements of ICA flow in 17 healthy young adults (age $= 28 \pm 7$ years) [308].

Non-fetal and fetal outflow boundary conditions To investigate the hypotheses, outflow boundary conditions that model the effect of non-fetal and fetal vascular physiology were developed. Alastruey et al. [3] developed a 1D model of the Circle of Willis and calculated flow rates in the large vessels after removing various vascular segments from the model. Using data from this paper (see Table 4.3.2), the mean volume flow rate in the ICA was calculated by summing the flow rates of the ACA, MCA and PComA for a given side of the vasculature (i.e. left or right). The ratio of ICA inflow to PComA outflow was used to calculate a ratio of PComA to ICA flow for non-fetal and fetal cases. The fetal/non-fetal flow splits are multiplied by the ICA inflow derived from the MGM to calculate outflow rates for the PComA in fetal and non-fetal conditions. For fetal patients, the inflow rate was multiplied by 21.7% to get the PComA outflow rate. For non-fetal patients, the inflow rate was multiplied by 0.34% to get the PComA outflow rate. This allows us to model distinct physiological conditions for each geometry.

Alastruey et al. [3] developed their 1D model based on one set of vessel parameters. To check that the derived flow splits fall within the expected variability across a range of patients, data from a clinical study was also analysed [4]. Zarrinkoob et al. [4] used phase-contrast MRI to assess cerebral blood flow (CBF) in 94 patients, 17 of which were fetal. In their results, they found that the percentage of total CBF (tCBF) in the P1 PCA was unchanged for fetal vs. non-fetal patients ($8 \pm 1\%$). For fetal patients this blood can only be supplied by the PComA, so it can be deduced that the PComA

Table 4.3.2: Mean volume flow-rates (ml/s) at the outlet of the efferent arteries and in the middle of the communicating arteries for different study cases. Table recreated from [3]. Complete circle is non-fetal, and RPCA/LPCA absent is fetal. ICA flow is calculated as the sum of ACA, MCA and PComA outflow.

Study case	Side	ACA	MCA	PComA	PComA:ICA [%]
Complete circle	Right	1.16	1.73	-0.01	0.34
Complete circle	Left	1.16	1.72	0.01	-0.35
PCA (P1) absent	Right	1.15	1.70	-0.79	21.7
PCA (P1) absent	Left	1.15	1.70	-0.79	21.7

ACA, anterior cerebral artery; ICA, internal carotid artery; MCA, middle cerebral artery; PCA, posterior cerebral artery; PComA, posterior communicating artery.

also accounts for 8 ± 1% of the tCBF. In fetal patients, the ipsilateral ICA accounts for 40 ± 3% of the tCBF. The PComA therefore accounts for 20 ± 3% of the ICA flow. From Zarrinkoob et al. [4], the PComA flow ratio in non-fetal patients can be calculated to be 2.6 ± 2.3% (see Table 4.3.3). The flow splits from Alastruey et al. [3] fall within the confidence intervals of the flow splits from Zarrinkoob et al. [4] for the fetal and right-sided non-fetal cases. As the left side non-fetal flow split (-0.35%) from Alastruey et al. [3] was outside the bounds of the flow split calculated from Zarrinkoob et al. [4], this scenario was not considered. The flow splits were used to calculate mass flow rates for non-fetal and fetal cases which were then imposed as mass flow outlet boundary conditions on the PComA.

Computational fluid dynamics simulations Blood flow simulations were performed by solving the unsteady Navier-Stokes equations in ANSYS CFX v19.1 (Ansys Inc., Canonsburg, PA, USA) using a finite volume method. Arterial wall distensibility was not considered and blood was modelled as an incompressible, Newtonian fluid with density 1066 kgm^{-3} and dynamic viscosity of 0.0035 Pa.s. The cardiac cycle was discretised into 200 equal steps. Timestep independence studies were performed previously by Villa-Uriol et al. [294] and Cebral et al. [309]. In these studies, the solution algorithm was iterated to convergence within each timestep, with the residuals required to converge by three orders of magnitude. Timestep independence was established by monitoring flow velocity, pressure and WSS at several points in the computational domain [294]. Each simulation was run for three cardiac cycles and results from the last

Table 4.3.3: Mean percentage of total cerebral blood flow measured in each artery all patients ($N = 94$) and fetal ($N = 17$) patients in each vessel with standard deviations [4]. Values for "All" and "Fetal" rows are directly from [4]. "Non-fetal" mean values are calculated using $\mu_{nf} = (\mu_{all} N_{all} - \mu_f N_f)/(N_{all} - N_f)$. Standard deviations for "Non-fetal" are taken as the standard deviation of "All". The "PComA" column values for "All" and "Non-fetal" are the remaining ICA flow percentage once the percentages for OA, MCA and ACA are subtracted. For "Fetal", the PComA flow percentage is simply the PCA flow percentage, as the PComA is the only supplier of the PCA. PComA to ICA flow ratios and standard deviations are calculated from the PComA and ICA mean flow percentages and standard deviations.

Study case	ICA	OA	ACA	MCA	PCA	BA	PComA	PComA:ICA [%]
All ($N = 94$)	36±4	2±1	11±4	21±3	8±1	20±4	2±1	5.6 ±3
Fetal ($N = 17$)	40±3	2±1	10±2	21±3	8±1	15±4	8±1	20±3
Non-fetal ($N = 77$)	35.1±4	2±1	11.2±4	21±3	8±1	21.1±4	0.9±1	2.6±2.3

ACA, anterior cerebral artery; BA, basilar artery; ICA, internal carotid artery; MCA, middle cerebral artery; OA, ophthalmic artery; PCA, posterior cerebral artery; PComA, posterior communicating artery.

cycle were analysed to reduce the effect of initial transients. No-slip boundary conditions were applied on vessel walls and zero pressure was applied at the ACA and MCA outlets. ICA Inflow and PComA outflow conditions were as described previously.

Post-processing and analysis Aneurysm velocity was extracted on the mesh nodes within the aneurysm sac and interpolated onto a linearly spaced 3D grid. The space-and-time-average of the interpolated velocity was calculated pre- and post-treatment and used to calculate aneurysm STAV reduction as a percentage using $100\% \times (\text{STAV}_{pre} - \text{STAV}_{post})/\text{STAV}_{pre}$. Aneurysm neck surfaces were included in the AneuX data and manually extracted for the @neurIST data. Velocity was extracted on mesh nodes in close proximity to the neck surface and similarly interpolated onto a uniform grid. The time-average of the neck velocity was calculated and the maximum value of the time-averaged velocity was taken to be the 99^{th} percentile. This value is referred to as MTAV and it was extracted pre- and post-treatment in order to calculate aneurysm neck MTAV reduction similarly to STAV reduction. The WSS field on the flow diverter surface was also extracted in the post-treatment simulations and STAWSS was calculated by taking the space-and-time-average of WSS across the entire clipped stent and all timesteps.

The three primary variables of interest are aneurysm STAV reduction, neck MTAV reduction and STAWSS. A T-test was performed for each variable with the null hypothesis that the independent samples (non-fetal and fetal) have identical means assuming identical variances. p-values were calculated and a value of $p < 0.05$ was used to determine statistical significance. Each variable of interest was compared with the morphological quantities of interest, namely (i) aneurysm maximum diameter, (ii) aneurysm neck diameter, (iii) aneurysm aspect ratio, (iv) aneurysm non-sphericity index, (v) PComA radius, and (vi) PComA and ICA radius ratio. A best-fit line was found for each data set and p and R^2 values were computed. The occlusion rate in non-fetal and fetal groups was calculated using using the $> 35\%$ STAV reduction haemodynamic end point for successful treatment. Alternative end point thresholds were tested for STAV and MTAV to determine the sensitivity of the results to this parameter.

4.3.3 Automation of the simulation pipeline

In order to improve the scalability and reproducibility of in-silico trials, a key challenge to address is the large amount of expert user input required in the simulation pipelines. Throughout this study, the steps in the simulation pipeline were automated where possible. Herein, details are provided of how each step was automated and the common failure modes.

Image segmentation Image segmentation was automated through development of a multi-task convolutional neural network and a patch-based learning algorithm that takes images as input and returns the segmented masks for vessel and aneurysm as output [299]. The most common reason for failure at this stage was the PComA not being segmented by the algorithm. This could be attributed to the PComA being small in size or the algorithm focusing on large vessels, such as the ICA.

Surface mesh pre-processing Surface mesh pre-processing was partially automated through the development of an algorithm that takes segmented masks as input and returns the vessel centreline, landmark points for terminal, bifurcation and aneurysm locations, and surface meshes for the vessel, inlet and outlet surfaces. The most common issue in this step was that the centreline algorithm was often unable to identify small thin vessels. The centreline is the key component in the surface mesh pre-processing as it is used to identify the inlet and outlet locations that are essential for applying boundary conditions. When the centreline algorithm does not identify all of the branch vessels, it is not possible to automatically clip the vessels and define surfaces for the outlets of the vessels. To overcome this, manual surface corrections had to be made for some cases to clip or remove small vessels.

Flow diverter deployment Device deployment was automated through an algorithm that takes the vessel centreline, vessel surface and device deployment landmark point as inputs and outputs the deployed device [296]. Device diameter was selected automatically by calculating the proximal and distal radii of the parent vessel and choosing a device that had a comparable radius to the vessel radii. Device length did not need to be chosen carefully as the subsequent clipping procedure removes the majority of the device. To automate the clipping procedure, the Vascular Modelling Toolkit (VMTK) was used to reconstruct the parent vessel without the aneurysm and the PComA branch vessel [310]. The intersection between the original vessel and the reconstructed VMTK vessel can then be found, which gives the aneurysm and PComA ostium. The ostium surface lies adjacent to the clipped PED and can be used to ensure the clipped PED has full coverage of the ostium, such that the influence of clipping is minimal. As well as having sufficient coverage, the ostium surface can be used to ensure the device is not over-deployed (i.e. does not bulge excessively into the aneurysm sac). The main issue with this step was finding a balance between not having an excessively large clipping radius and retaining a sufficient portion of the device to cover the ostium. Manual quality control was performed to identify cases with under/over-sized clipped devices and then the device was re-clipped with a larger/smaller clipping radius as required.

Volumetric meshing Volumetric meshing was automated through the creation of ANSYS ICEM CFD macro scripts for each mesh configuration based upon a template macro. Each macro script contains instructions for the meshing software to read in the surface mesh files and generate the volumetric mesh using the pre-defined meshing protocols and the pre-selected meshing parameters. This step did not have any common failure modes.

Inflow boundary conditions Inflow boundary condition generation was automated by the creation of an algorithm that takes the age and sex of a patient as input and outputs ICA flow-rate waveforms for various physiological conditions (rest, hypertension, exercise) [91]. This step did not have any common failure modes.

Non-fetal and fetal outflow boundary conditions Outflow boundary condition generation was automated through the development of code that takes the inflow boundary condition as input and then uses the non-fetal and fetal flow splits derived from literature studies to generate PComA outlet flow-rate waveforms for the different conditions. This step did not have any common failure modes.

Computational fluid dynamics simulations Simulation set up was automated through the creation of a template file defining the array of simulation settings and parameters. The patient-specific boundary conditions were then applied to the template before the patient-specific simulation set up file was combined with the volumetric mesh for the given case. This process was automated through the use of shell scripts interfacing with the ANSYS software. Simulation execution was automated through the generation of job scripts to submit the simulations to a high-performance computing cluster. These steps did not have any common failure modes.

Post-processing and analysis Post-processing was automated by first converting the ANSYS simulation results files into VTK files and then using various Python libraries and Paraview to extract the quantities of interest from each simulation. The second post-processing stage performed multi-simulation analysis, such as the calculation of flow reduction using the pre- and post-treatment results. The final post-processing stage analysed the quantities of interest using various Python libraries for statistical analysis and plotting. These steps did not have any common failure modes.

4.4 Results

4.4.1 Qualitative Results

Visualisations of the velocity flow fields in non-fetal and fetal patients can help to develop an understanding of the impact that the different physiological conditions have on the aneurysm flow reduction. Figure 4.4.1 shows post-treatment systolic velocity streamlines for one patient under non-fetal and fetal flow conditions from two viewing angles. These images demonstrate how the increased flow imposed through the PComA in the fetal patient draws a greater amount of blood flow across the flow diverter and leads to higher residual flow in the aneurysm than in the non-fetal case. For this patient, the non-fetal aneurysm STAV reduction was 91.0% whereas it was 62.1% in the fetal case. Given all other factors are identical (geometry, inflow conditions, material properties, etc.), this highlights the large discrepancy in aneurysm flow reduction that is caused by non-fetal and fetal flow conditions. However, this result is only for one patient and more significant results are found when analysing the entire cohort.

4.4.2 Flow Variables vs. Presence of Fetal Posterior Circulation

The cohort was stratified into non-fetal and fetal patients by imposing distinct outflow boundary conditions at the PComA outlet. The results demonstrated that aneurysm STAV reduction was significantly lower in fetal cases than non-fetal cases ($p < 10^{-11}$): mean STAV reduction was 67.8% in non-fetal patients but only 46.5% in fetal patients. Similarly, aneurysm neck MTAV reduction was significantly lower in fetal than non-fetal patients ($p = 0.001$). Flow reduction is a key feature of aneurysm treatment by flow diverting devices; these results (Figure 4.4.2) correspondingly suggest that treatment success will be lower in patients with FPC. Post-treatment stent STAWSS was also found to be significantly higher in fetal than non-fetal patients ($p < 0.05$): mean STAWSS was 23.5 Pa in non-fetal patients and 29.2 Pa in fetal patients. Higher stent WSS suggests that endothelialisation is more likely to be inhibited in patients with fetal posterior circulation.

4.4.3 Treatment Success Rate vs. Successful Treatment Threshold

Prediction of treatment success requires specification of threshold values for relevant variables of interest. In a previous in-silico study [20], a 35% reduction in aneurysm STAV was used as a success criterion. With such a value, treatment success rates of 98.4% and 85.3% were found in non-fetal and fetal patients, respectively. These are substantially higher than the corresponding success rates (81.8% and 43.7%, respectively) reported by Rinaldo et al. [2]. Applying a 35% MTAV reduction threshold yielded treatment success rates of 63.9% and 32.8% in non-fetal and fetal patients, respectively. Figure 4.4.3 shows the predicted treatment success rates for a range of aneurysm STAV

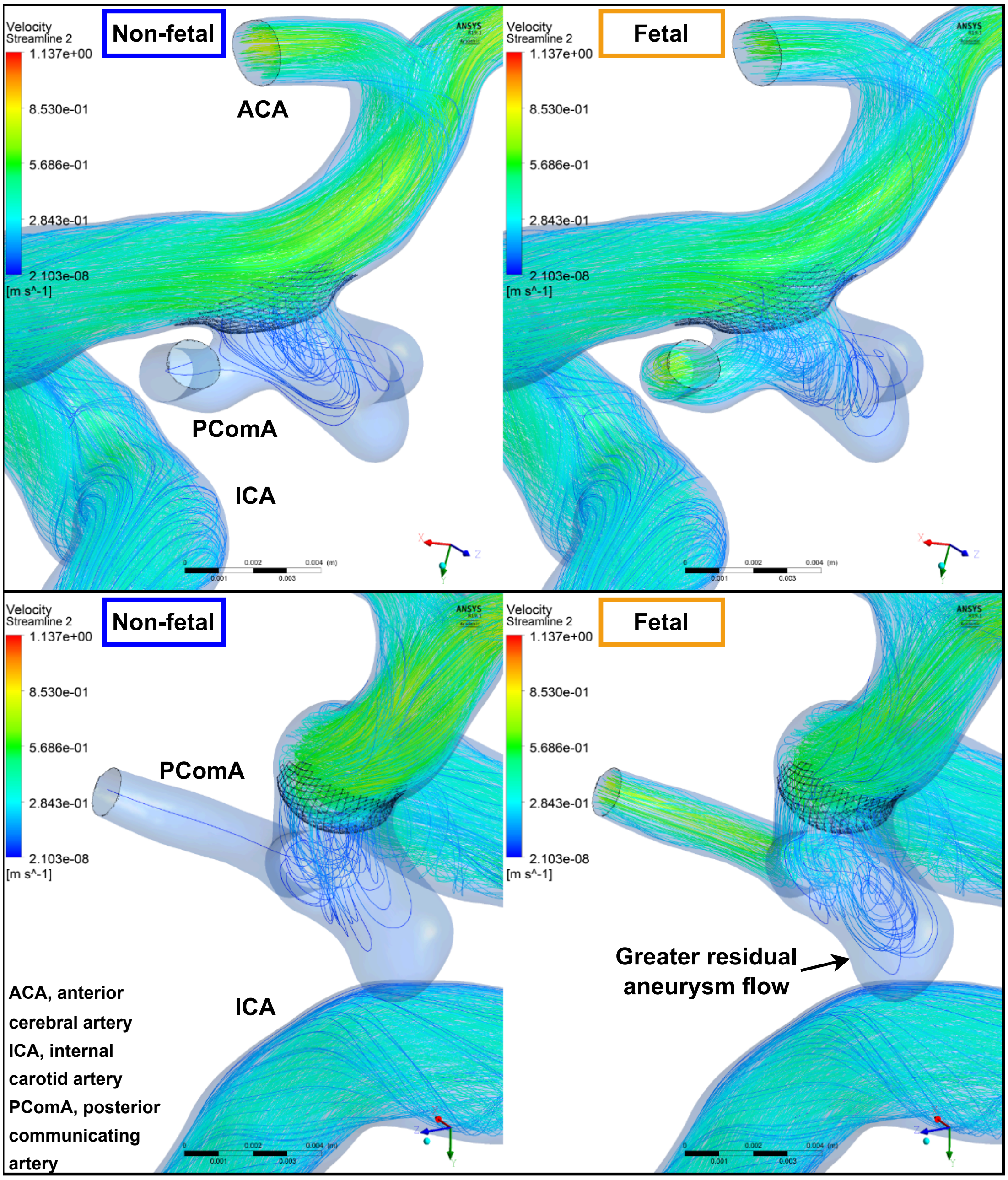


Figure 4.4.1: Systolic velocity streamlines for one patient from FD-PComA under non-fetal and fetal flow conditions. Aneurysm STAV reduction was 91.0% for the non-fetal case and 62.1% for the fetal case.

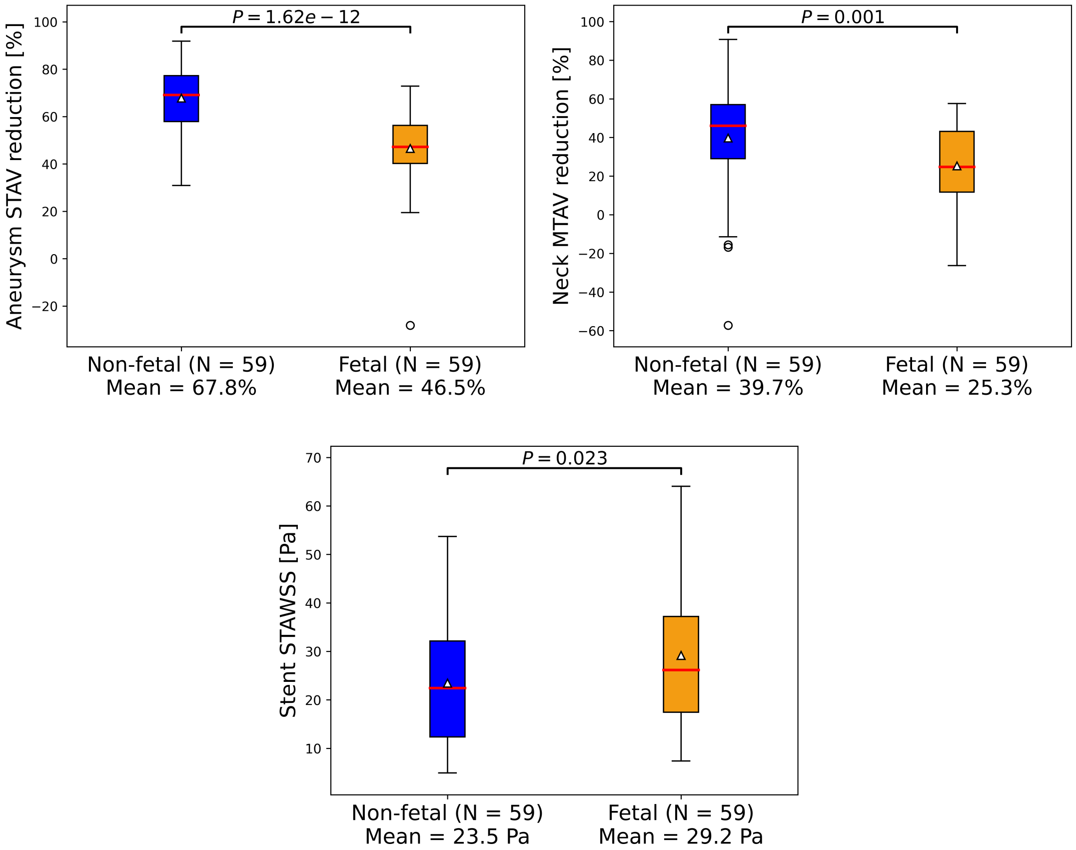


Figure 4.4.2: Flow reduction characterised by aneurysm STAV and MTAV reduction and stent STAWSS plotted for non-fetal and fetal physiology. In the box plots, the triangle marker is the mean and the red line is the median. The box extends from the first quartile (Q1) to the third quartile (Q3). The whiskers extend to the farthest data point lying within 1.5× the inter-quartile range (IQR) from the edge of the box (i.e. $Q1 - 1.5IQR$ to $Q3 + 1.5IQR$). The circles represent the data points that lie outside the range of the whiskers. STAV, space-and-time-averaged velocity; MTAV, maximum time-averaged velocity; STAWSS, space-and-time-averaged wall shear stress.

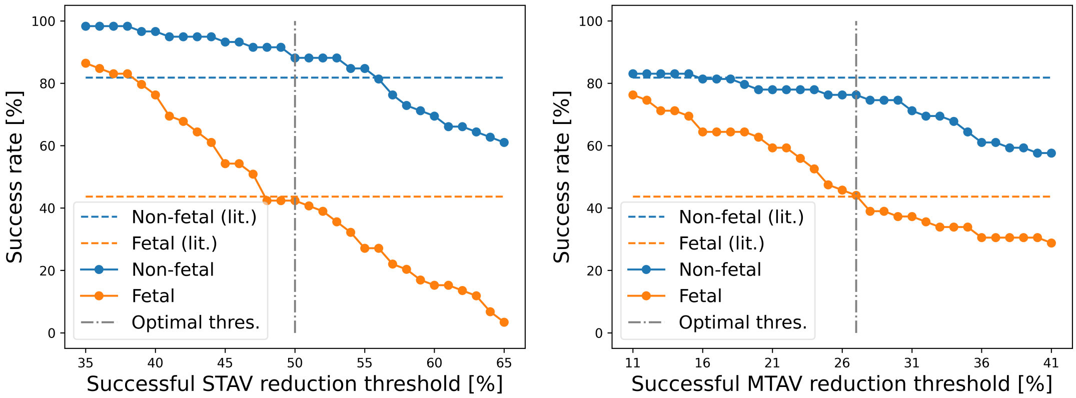


Figure 4.4.3: Treatment success rates against different STAV/MTAV reduction thresholds. Literature treatment success rates from Rinaldo et al. [2] are included as dashed lines. STAV, space-and-time-averaged velocity; MTAV, maximum time-averaged velocity.

and MTAV threshold values, with the corresponding success rates from [2]. These results suggest optimal matches between the predicted success rates and the literature values are obtained with the following thresholds: STAV reduction of 50%, yielding non-fetal and fetal success rates of 88.5% and 42.6%, respectively; and MTAV reduction of 26%, yielding non-fetal and fetal success rates of 74.4% and 44.3%, respectively. These results demonstrate two key points: (i) there is a distinct difference in non-fetal vs. fetal treatment success for a wide range of thresholds, (ii) 35% STAV reduction is not applicable across different aneurysm sub-groups or for different measures of flow reduction.

4.4.4 Flow Variables vs. Morphological Parameters

Statistical analyses were performed to test for correlation between flow variables and morphological parameters describing the aneurysm and the PComA. The flow variables tested were aneurysm STAV reduction, aneurysm neck MTAV reduction and post-treatment stent STAWSS. Aneurysm morphology was characterised by maximum diameter, aspect ratio, neck width and non-sphericity index (see Figure 2.2.2 for definitions of these parameters). PComA size was characterised by its radius and the ratio of the PComA radius to the ICA radius. The analyses were performed for non-fetal, fetal and all (i.e. non-fetal and fetal combined) patients. Linear regression was performed between each set of variables for each physiology. R^2 and p values are presented in Table 4.4.1. R^2 was low across all combinations and p values were typically > 0.05, which demonstrates that there was typically low correlation between flow variables and morphological characteristics and low significance associated to the best fit lines between the variables. Only PComA to ICA radius ratio consistently provided p values that were < 0.05 in the case of fetal patients. However, this variable is implicitly encoded into the outflow boundary conditions, as both the ICA inlet and PComA outlet areas are used in the calculation of the outflow boundary condition applied at the PComA. The significance found between the flow variables and the PComA to ICA radius ratio is therefore thought to be associated to the PComA outflow boundary conditions, which is consistent with the results presented in Figure 4.4.2. Scatter plots of the flow variables and the morphological quantities and the best fit lines can be found in Appendix A.3.

Table 4.4.1: Statistical values (R^2 and p value for best fit line) were calculated to quantify the correlation between flow variables (STAV/MTAV reduction, STAWSS) and morphological parameters (aneurysm maximum diameter, neck width, aspect ratio, NSI; PComA radius, PComA to ICA radius ratio) for different physiologies (non-fetal, fetal, or both).

		Physiology and Statistics					
		Non-fetal		Fetal		Non-fetal and fetal	
Morphological Variable	Flow Variable	R^2	p	R^2	p	R^2	p
Aneurysm max. diameter	STAV reduction	0.084	0.026	0.009	0.474	0.006	0.420
Aneurysm max. diameter	MTAV reduction	0.035	0.158	0.032	0.174	0.001	0.733
Aneurysm max. diameter	Stent STAWSS	0.015	0.354	0.001	0.806	0.002	0.668
Aneurysm aspect ratio	STAV reduction	0.009	0.482	0.001	0.772	0.000	0.823
Aneurysm aspect ratio	MTAV reduction	0.023	0.253	0.000	0.97	0.007	0.359
Aneurysm aspect ratio	Stent STAWSS	0.000	0.913	0.007	0.525	0.001	0.693
Aneurysm neck width	STAV reduction	0.209	0.000	0.008	0.506	0.021	0.122
Aneurysm neck width	MTAV reduction	0.042	0.121	0.046	0.104	0.001	0.763
Aneurysm neck width	Stent STAWSS	0.044	0.111	0.000	0.946	0.010	0.275
Aneurysm NSI	STAV reduction	0.004	0.634	0.006	0.548	0.000	0.932
Aneurysm NSI	MTAV reduction	0.001	0.793	0.000	0.918	0.001	0.791
Aneurysm NSI	Stent STAWSS	0.012	0.410	0.001	0.798	0.005	0.465
PComA radius	STAV reduction	0.021	0.279	0.023	0.257	0.000	0.953
PComA radius	MTAV reduction	0.002	0.734	0.112	0.009	0.011	0.248
PComA radius	Stent STAWSS	0.001	0.775	0.017	0.321	0.003	0.589
PComA:ICA radius ratio	STAV reduction	0.000	0.945	0.087	0.024	0.014	0.204
PComA:ICA radius ratio	MTAV reduction	0.000	0.907	0.138	0.004	0.024	0.094
PComA:ICA radius ratio	Stent STAWSS	0.060	0.061	0.175	0.001	0.108	0.000

ICA, internal carotid artery; MTAV, aneurysm neck maximum time-averaged velocity; STAWSS, space-and-time-averaged wall shear stress; STAV, aneurysm space-and-time averaged velocity; PComA, posterior communicating artery.

4.4.5 Computational Resources

The FD-PComA in-silico trial required simulation of pre- and post-treatment configurations in 118 virtual patients, giving a total of 236 simulations. Each simulation was performed using 32 Central Processing Unit (CPU) cores on a High Performance Computing cluster. The mean run-time was 0.96±1.02 hours for pre-treatment simulations and 13.5 ±6.09 hours and for post-treatment simulations (see Table 4.4.2). The total simulation execution time for the complete IST was 1722 hours, or 72 days (rounded

Table 4.4.2: Simulation run times and their standard deviations (SDs) in hours for the FD-PComA IST. Each simulation was parallelised across 32 cores and run on a High Performance Computing cluster.

		Run time (hours)		
Case	Number of simulations	Mean	SD	Total
No device	118	0.96	1.02	113
Device	118	13.5	6.09	1589

to the nearest day). This highlights the demand for simulation acceleration, as medical device design optimisation could potentially require simulations of multiple devices in significantly higher numbers of patients under a wider array of physiological states.

4.5 Conclusion

The FD-PComA in-silico trial provided evidence suggesting: (1) PED flow diverter treatment of PComA aneurysms is less effective in patients with fetal posterior circulation and (2) this is due to the increased flow rate through the PComA in this scenario. Morphological variables like PComA size, aneurysm maximum diameter, aneurysm aspect ratio, aneurysm neck width and aneurysm non-sphericity index did not influence the treatment outcome. Relating the findings to the initial trial hypotheses, the following conclusions are drawn:

1. Maintained PComA flow following flow diversion reduces treatment success rates in patients with FPC.
2. PComA aneurysm flow diverter treatment success is most affected by the presence of FPC and was not significantly influenced by PComA diameter, aneurysm maximum diameter, aneurysm aspect ratio, aneurysm neck width or aneurysm non-sphericity index.

The sub-analyses demonstrated that endothelialisation, as characterised by wall shear stress on the device surface, is hindered by the presence of FPC and is not influenced by the morphology of the PComA or aneurysm. These findings lead to the conclusion that PComA aneurysm patients with fetal posterior circulation should be treated by alternative means to single PED flow diverter treatment.

CHAPTER 5

Reduced Order Modelling of Intracranial Aneurysm Flow Using Proper Orthogonal Decomposition and Neural Networks

Abstract

Reduced order models (ROMs) reduce the dimensionality of high-fidelity mathematical models, facilitating large gains in simulation execution speed. Recently, machine learning (ML) has been used to overcome limitations with some ROM techniques, proving particularly adept at improving parametric ROMs that depend upon interpolation. Despite this, the performance of ML-ROMs are yet to be evaluated in a wide array of applications and questions remain regarding the optimal design of ML-ROMs. This chapter develops a non-intrusive parametric ML-ROM and applies it to a nonlinear, time-dependent fluid dynamics problem in a complex 3D geometry, contributing to optimal ML-ROM design in future studies. The ML-ROM was constructed using Proper Orthogonal Decomposition (POD) for dimensionality reduction and neural networks for interpolation of the ROM coefficients. Different network designs and parametric sampling regimes are compared in terms of approximation accuracy. The ML-ROM was applied to physiological flow variation in intracranial aneurysms, which is important when considering rupture risk and treatment performance. Physiological variation was parameterised by introducing three scale factors into the inflow velocity waveform that control flow magnitude, pulsatility and heart rate. Across the various sampling regimes tested, similar model performance was found. The best-performing network design used a two-level POD reduction, a technique rarely used in previous studies. The best-performing ML-ROM achieved mean test accuracy of 98.6% and 97.6% in vessel and aneurysm regions, respectively, while providing speed-up of order 10^5 times.

5.1 Introduction

The many-query or real-time solution of parameterised partial differential equations (PDEs) is a common scenario that arises for example in the design optimisation, uncertainty quantification and optimal control of problems relating to the modelling of physical systems. Using high-fidelity computational models in these scenarios remains challenging due to the cost of solving high-dimensional PDEs. Reduced order models (ROMs) are low-order representations of high-order models that preserve essential input-output behaviour at the cost of some model accuracy and are commonly used to accelerate expensive computational models [96, 97]. Machine learning (ML) is increasingly being used to improve upon traditional ROM techniques, but questions remain about the optimal design of ML-ROMs and their suitability to real-world applications [104, 105, 167, 212, 215–217].

Traditional ROM construction for nonlinear PDEs typically involves: (i) solving the full order model (FOM) using standard numerical techniques for a number of model configurations to generate data, or *snapshots*, for the high-order system; (ii) extracting a low-order representation of the solution manifold using dimensionality reduction techniques such as Proper Orthogonal Decomposition (POD), Dynamic Mode Decomposition (DMD) or auto-encoders [99, 103, 104]; and (iii) applying projection or interpolation methods in the low-order space to build a ROM [165]. Step (iii) of the ROM construction can be performed either with *intrusive* approaches that project the underlying governing equations onto the low-dimensional manifold, or with *non-intrusive* approaches that interpolate the ROM coordinates in the low-dimensional space. Machine learning using neural networks has improved upon classical ROM methods by offering alternative dimensionality reduction techniques [104, 105, 214, 311], approximating algebraic operators in projection-based ROMs applied to nonlinear problems [100, 218], improving stability and long-term prediction accuracy [217], and providing high-dimensional interpolation techniques in interpolation-based ROMs [212, 213, 215].

In this chapter, a machine learning ROM is applied to parameterised physiological flow variation in intracranial aneurysms. Intracranial aneurysms are pathological bulges in blood vessels in the brain that are estimated to occur in 5–8% of the general population [26]. While aneurysms are often asymptomatic, their rupture causes subarachnoid haemorrhage, an event with high rates of mortality, morbidity and disability, as well as high healthcare costs [28]. In aneurysm haemodynamics, adverse physiological conditions can increase flow velocity, wall pressure and wall shear stress inside the aneurysm sac, increasing the risk of rupture [21]. Different physiological conditions have also been shown to affect aneurysm treatment performance [20]. Modelling physiological flow variability in aneurysms is a time-dependent, geometrically complex problem that requires repeated evaluations of an expensive 3D Navier-Stokes model to solve. This gives us an excellent real-world complex application with which to establish ROM performance.

Various ROM techniques have been applied to haemodynamic problems, including approaches based on POD [93, 100, 148, 150–152, 216], DMD [174, 177, 184] and reduced basis methods [192–194]. The approach taken in this chapter is to use POD for dimensionality reduction and fully-connected neural networks (FCNNs) for interpolation of the ROM coefficients. Non-intrusive interpolation-based ROMs were chosen for their ability to solve parameterised time-dependent problems while providing larger computational speed-ups than their projection-based counterparts [212, 213, 215]. Three different network designs are investigated to improve on previous work on this type of ROM, including a design where a double-POD is utilised for further dimensionality reduction [146]. To generate training data, a validated computational fluid dynamics (CFD) model of blood flow in a 3D aneurysm derived from 3D rotational angiography (3DRA) images is used. Physiological flow variation is parameterised by introducing three inlet flow-rate waveform scale factors that control flow magnitude, pulsatility and heart rate. Five approaches for sampling the parameter space are investigated: (i) Uniform grid, (ii) Latin Hypercube with bounds, (iii) Latin Hypercube without bounds, (iv) Random with bounds, (v) Random without bounds. For each sampling regime, a distinct set of snapshots are generated and used to train a ROM and evaluate its performance. For the best-performing network design and hyperparameters, results are presented on the accuracy of the ROM relative to the FOM and on the acceleration factor and data storage reduction provided by the ROM.

The chapter is outlined as follows. Section 5.2 details the methodology used in the CFD model (5.2.1) and POD-Interpolation ROM (5.2.3). Section 5.3 presents results from the multi-stage hyperparameter optimisation (5.3.2, 5.3.3), the network design comparison and sampling method investigation (5.3.4), the best-performing model (5.3.5), and the acceleration and data storage reduction quantification (5.3.6). Section 5.4 provides conclusions from the chapter. Discussion for the chapter can be found in the thesis discussion (Section 6.3).

5.2 Methodology

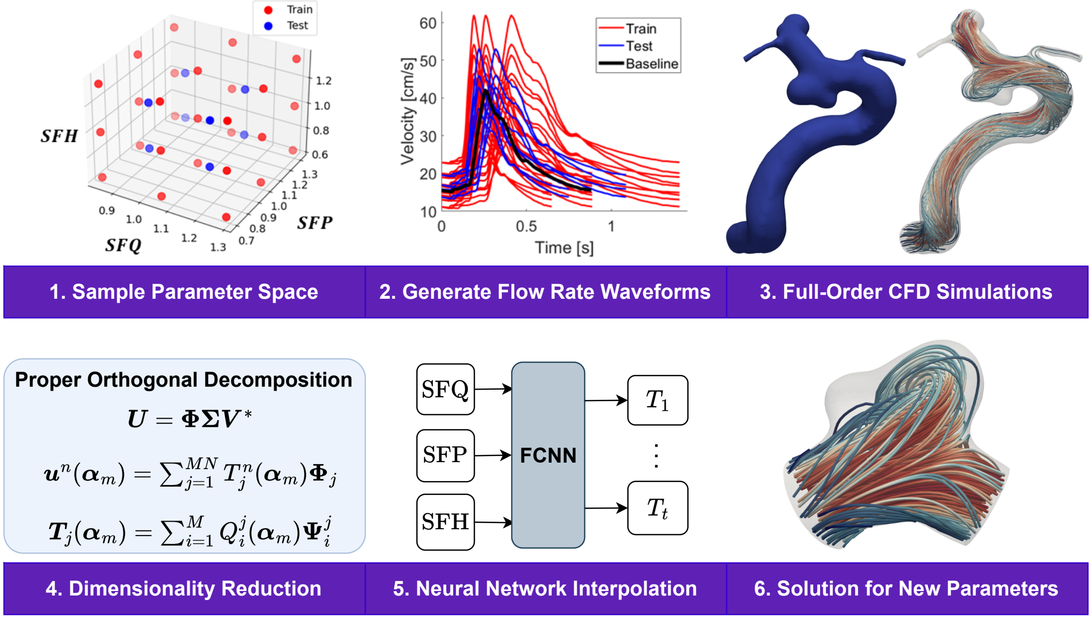


Figure 5.2.1: The ML-ROM methodology uses Proper Orthogonal Decomposition for dimensionality reduction and fully-connected neural networks for interpolation of the ROM coefficients.

5.2.1 Intracranial Aneurysm Computational Fluid Dynamics

Geometry

Figure 5.2.2 shows the 3D aneurysm geometry used in this study. The anatomic surface model of the geometry was obtained from 3D rotational angiography images using an automatic segmentation method based on geodesic active regions, details of which can be found in [298, 312]. The aneurysm diameter is 6.1 mm and its aspect ratio is 0.722. The full geometry was used for the CFD model, but clipped geometries in the parent vessel and aneurysm locations were used for the ROM, as non-intrusive ROMs can be constructed for regions of interest to provide greater speed-up and reduce data storage requirements. The aneurysm model used in this chapter is for a side wall internal carotid artery (ICA) aneurysm, which is geometrically simpler than the PComA aneurysms investigated in Chapter 4. The flow is also simpler for the side wall aneurysm, as the branching vessels are further from the aneurysm and therefore have less influence on the flow in the aneurysm. The simpler problem was chosen for this chapter, as the goal is to construct an ML-ROM rather than to answer clinically relevant questions about aneurysm treatment as was the goal in Chapter 4.

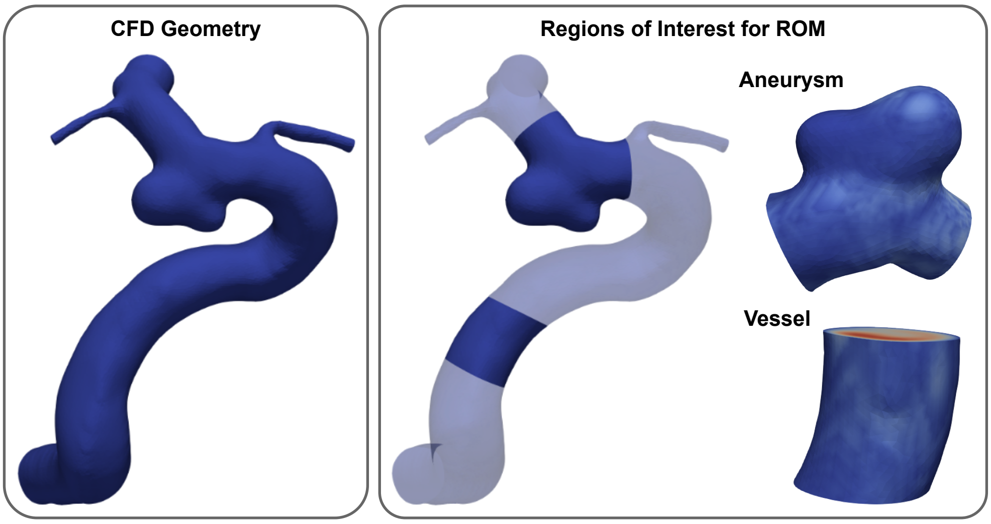


Figure 5.2.2: Full geometry used in the computational fluid dynamics (CFD) simulations and the truncated vessel and aneurysm geometries used for the reduced order model (ROM).

Computational Fluid Dynamics Model

The three-dimensional Navier-Stokes equations for incompressible Newtonian fluid were used to describe blood flow:

$$\rho \left(\frac{\partial \boldsymbol{u}}{\partial t} + (\boldsymbol{u} \cdot \nabla) \boldsymbol{u} \right) = -\nabla p + \mu \nabla^2 \boldsymbol{u}, \tag{5.2.1}$$

where $\boldsymbol{u}$ and p are velocity and pressure, respectively. Blood is assumed to be a Newtonian fluid with constant density $\rho = 1066$ kgm^{-3} and viscosity $\mu = 0.0035$ Pa·s. A volumetric mesh was constructed using ANSYS ICEM CFD v19.1 (Ansys Inc., Canonsburg, PA, USA) and the Navier-Stokes equations were solved with ANSYS CFX v19.1 using a finite volume method.

A flow rate waveform (FRW) is prescribed at the inlet and zero pressure conditions at all outlets. Vessel walls were no-slip boundaries. The Reynolds number did not exceed 1000 across the various physiologies, thus all simulations were laminar. Each simulation was run for three cardiac cycles. The period of the baseline cardiac cycle was 0.883 seconds. Each cycle was discretised into 100 timesteps, with the velocity and pressure fields taken from only the final cycle to reduce initial transients.

Parameterisation of Physiological Flow Variability

Physiological variability was parameterised through the inlet FRW, which was derived from a Multivariate Gaussian Model (MGM) trained on patient-specific phase-contrast magnetic resonance imaging measurements of internal carotid artery flow in 17 healthy young adults (age 28 ± 7 years). Further details on the MGM are reported in [21, 313].

From Geers et al. [314], the FRW is of the form

$$Q(t) = aQ^0(ct) + b, \tag{5.2.2}$$

$$a = \underbrace{\frac{Q_{TA}}{Q^0_{TA}}}_{SFQ}\underbrace{\frac{PI}{PI^0}}_{SFP}, \quad b = Q_{TA}\Big(1 - \underbrace{\frac{PI}{PI^0}}_{SFP}\Big) \quad c = \underbrace{\frac{HR}{HR^0}}_{SFH}, \tag{5.2.3}$$

where Q^0 is the MGM-derived time-varying waveform, Q_{TA} is the time-averaged flow rate, PI is the pulsatility index (the difference between peak systolic and minimum diastolic flow velocity divided by the time-averaged flow velocity) and HR is the heart rate for a particular physiology. Superscript 0 (e.g. Q^0_{TA}) represents the value of the given variable for the baseline waveform derived from the MGM. Three scale factors for flow magnitude (SFQ), pulsatility (SFP) and heart rate (SFH) are defined through equation (5.2.3), as

$$SFQ = \frac{Q_{TA}}{Q^0_{TA}}, \quad SFP = \frac{PI}{PI^0}, \quad SFQ = \frac{HR}{HR^0}. \tag{5.2.4}$$

A range of configurations of SFQ, SFP and SFH can be sampled based upon physiologically realistic bounds and then the FRW for each configuration is calculated through equations (5.2.2) and (5.2.3). The inlet FRWs are applied through a parabolic velocity profile at the inlet to the geometry. The inlet velocity condition drives blood flow into the aneurysm for a given parameter configuration or physiology.

Flow Rate Bounds Ford et al. [308] measured left and right ICA flow rates in 17 young, healthy volunteers at rest. They found mean total (left + right) ICA flow of 549 mL min^{-1} with $\sigma_{inter} = 103$, $\sigma_{intra} = 21$, minimum flow rate of 398 mL min^{-1} and maximum flow rate of 850 mL min^{-1}. Using maximum and minimum flow rate values gives SFQ bounds of $[0.72, 1.54]$. Using only intra-patient variability gives SFQ bounds of $[0.96, 1.04]$. The ROM approach is only applicable to intra-patient variability, but to induce additional variation into the snapshots, the bounds are extended to $[0.84, 1.29]$, which lies midway between the bounds for intra- and inter-patient variability.

Pulsatility Index Bounds Pulsatility index is an easy to obtain (non-invasive) parameter that has been used to assess macrocirculation in highly prevalent medical conditions, such as hypertension, type 1 and 2 diabetes, and thyroid disorders [315]. Pulsatility index varies with age - 1.08 ± 0.29 for ICA blood flow in adults [316]. This is a variance of ±31%, so the bounds of SFP are chosen to be $[0.69, 1.31]$.

Heart Rate Bounds Ford et al. [308] measured mean heart rate as 68±8 bpm (range 56-83) in 17 young, healthy volunteers at rest. Matsuo et al. [317] found that during moderate steady-state and incremental exercise for 12 young, healthy volunteers, mean heart rate did not exceed 120 bpm. Therefore, 56 was selected as the minimum heart rate and 120 as the maximum, with 88 as the mean. This gives SFH bounds of $[0.63, 1.36]$.

5.2.2 Sampling

Various sampling techniques are investigated. For each technique, 27 configurations are generated, which corresponds to a $3 \times 3 \times 3$ grid in a uniform-grid sampling regime. The sampling methods used are shown illustratively in two dimensions in Figure 5.2.3, alongside the test cases created for ROM evaluation.

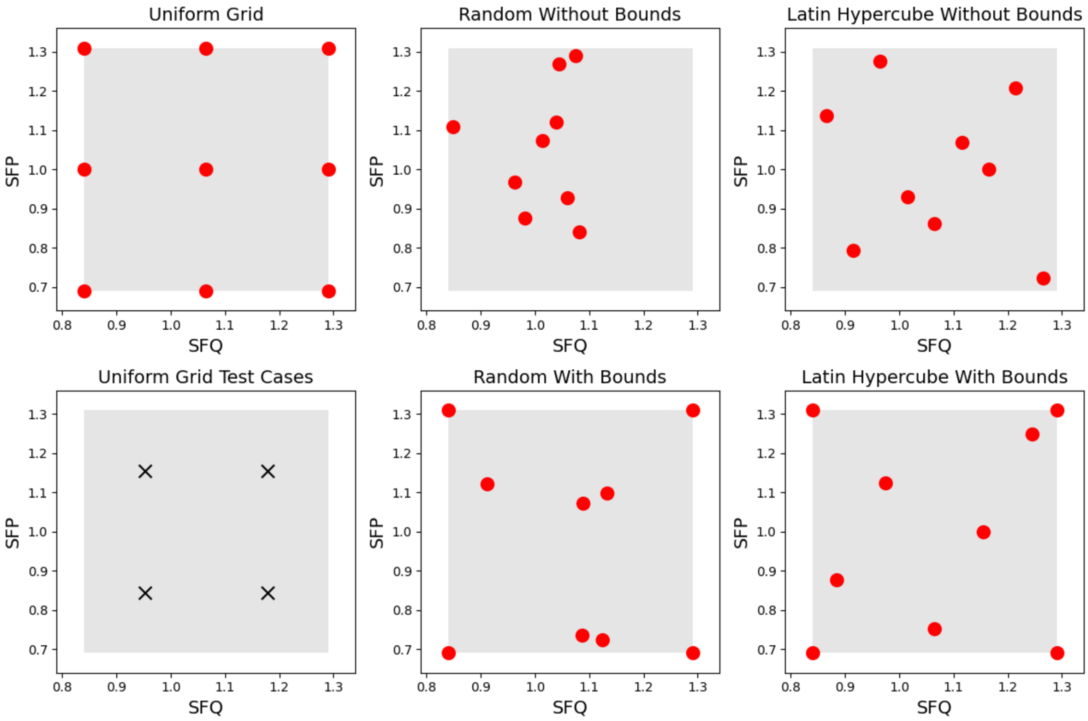


Figure 5.2.3: Exemplar 2D sample points for various methods investigated in this study.

Uniform Grid Uniform Grid sampling consists of discretising each input parameter with a number of points and generating all possible combinations of the input parameters. Figure 5.2.3 shows uniform sampling with two input parameters and three points in each parametric direction. For a parameter space with a parameters and b points in each direction, Uniform Grid sampling will give a^b samples. This exponential dependence on the number of parameters means that the number of samples grows rapidly as additional parameters are incorporated.

Latin Hypercube Latin Hypercube sampling consists of quasi-randomly populating the parameter space with a chosen number of points, with no two points residing in the same row or column of the discretised parameter space. Latin Hypercube sampling provides better uniformity when compared with random sampling [318]. Similarly to random sampling, the domain bounds can be included, as shown in Figure 5.2.3.

Random Random sampling consists of defining a chosen number of points and randomly populating the parameter space with this many points. The bounding points of the parameter space may also be included, as shown in Figure 5.2.3. Random sampling can suffer from insufficient distribution of points, which can lead to the ROM being inaccurate in some regions of the parameter space.

5.2.3 POD-Interpolation ROM

Herein, the POD-Interpolation method used in this work is described. The approach and notation for POD is the similar to that used by Walton et al. [146]. Another description of this method can be found in Wang et al. [165]. The benefits of this approach are: (i) it is non-intrusive; (ii) it is relatively straightforward to implement and solve; (iii) it is faster than projection-based ROMs; (iv) it can be applied to regions of interest instead of the full geometry; (v) when compared to element-wise interpolation between full order solutions, it drastically reduces the amount of data and the number of interpolation operations required.

POD

The full order CFD model described in section 5.2.1 is used to generate velocity fields for a range of parameter configurations. $M = 27$ parameter configurations are used, with $\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_M$ denoting the configuration vectors and $\boldsymbol{\alpha}_m = (SFQ_m, SFP_m, SFH_m)$. Each time-varying solution of the FOM for one parameter configuration is taken as a snapshot. Each snapshot is an $D \times N$ matrix, where D is the number of mesh nodes and N is the number of timesteps. After running the FOM simulations for all configurations, a snapshot matrix $\boldsymbol{U}$ is generated with the form:

$$\boldsymbol{U} = \begin{bmatrix} u_1^1(\boldsymbol{\alpha}_1) & \ldots & u_1^N(\boldsymbol{\alpha}_1) & \ldots & u_1^N(\boldsymbol{\alpha}_M) \\ \vdots & \ddots & \ldots & \ldots & \\ u_D^1(\boldsymbol{\alpha}_1) & \ldots & u_D^N(\boldsymbol{\alpha}_1) & \ldots & u_D^N(\boldsymbol{\alpha}_M) \end{bmatrix}. \tag{5.2.5}$$

A column of $\boldsymbol{U}$ lists the values of the velocity magnitude at each of the mesh points $1, \ldots, D$ at one timestep n for a certain parameter configuration $\boldsymbol{\alpha}_m$. This snapshot matrix will typically have many more rows than columns.

$\boldsymbol{U}$ is factorised using an economic singular value decomposition (SVD):

$$\underset{D\times MN}{\boldsymbol{U}} = \underset{D\times MN}{\boldsymbol{\Phi}} \times \underset{MN\times MN}{\boldsymbol{\Sigma}} \times \underset{MN\times MN}{\boldsymbol{V}^*}. \tag{5.2.6}$$

The dimensions of each matrix are included in equation (5.2.6), with $MN = M \times N$. $\boldsymbol{\Phi}$ are the left singular vectors of $\boldsymbol{U}$, $\boldsymbol{\Sigma}$ are the singular values and $\boldsymbol{V}^*$ are the right singular vectors. The columns of $\mathbf{\Phi}$, denoted $\mathbf{\Phi}_j$ with $j = 1, \dots, MN$, are the mutually orthogonal POD modes. $\mathbf{\Sigma}$ is a diagonal matrix with each of the singular values, denoted σ_j, indicating the energetic contribution of each POD mode.

It is possible to reconstruct each column of $\boldsymbol{U}$ using

$$\boldsymbol{u}^n(\boldsymbol{\alpha}_m) = \sum_{j=1}^{MN} T_j^n(\boldsymbol{\alpha}_m)\mathbf{\Phi}_j, \tag{5.2.7}$$

where $\boldsymbol{T}^n(\boldsymbol{\alpha}_m)$ are a set of parameter-dependent coefficients that can be considered as a path through the coordinate system given by $\mathbf{\Phi}$ [146]. Each of the simulated parameter configurations will have a corresponding matrix $\boldsymbol{T}^n(\boldsymbol{\alpha}_m)$ and the goal of POD-Interpolation is to predict the path taken by a new set of parameter values. To do this quickly, it is first necessary to reduce the order of the system.

The system order can be reduced by considering the energy contained within each POD mode and then retaining r nodes up to a certain energy threshold. The sum of the squares of all of the singular values in $\mathbf{\Sigma}$ gives a measure of the total energy contained in the POD modes, so the first r modes will contain a fraction of the total energy. Provided the majority of the energy in the system is captured in the first several modes, a large number of modes can be discarded and thus the order of the system reduced. Mathematically, this equates to choosing a value for the energy fraction E_{frac} to be retained and then finding r such that equation (5.2.8) is satisfied.

$$E_{\text{frac}} \leq \frac{\sum_{j=1}^{r} \sigma_j^2}{\sum_{j=1}^{M\times N} \sigma_j^2} \tag{5.2.8}$$

The snapshot matrix can then be truncated as follows:

$$\boldsymbol{U} = \begin{bmatrix} \underbrace{\mathbf{\Phi}_1 \quad \dots \quad \mathbf{\Phi}_r}_{\text{Retained}} & \underbrace{\mathbf{\Phi}_{r+1} \quad \dots \quad \mathbf{\Phi}_{MN}}_{\text{Discarded}} \end{bmatrix} \begin{bmatrix} \sigma_1 & 0 & \dots & \dots & 0 \\ 0 & \ddots & \ddots & & \vdots \\ \vdots & \ddots & \sigma_r & \ddots & \vdots \\ \vdots & & \ddots & \ddots & 0 \\ 0 & \dots & \dots & 0 & \sigma_{MN} \end{bmatrix} \begin{bmatrix} \mathbf{v}_1^* \\ \vdots \\ \mathbf{v}_r^* \\ \vdots \\ \mathbf{v}_{MN}^* \end{bmatrix}. \tag{5.2.9}$$

After discarding the high-order modes, the system becomes:

$$\boldsymbol{U} \approx \begin{bmatrix} \mathbf{\Phi}_1 & \dots & \mathbf{\Phi}_r \end{bmatrix} \begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_r \end{bmatrix} \begin{bmatrix} \mathbf{v}_1^* \\ \vdots \\ \mathbf{v}_r^* \end{bmatrix}. \tag{5.2.10}$$

Figure 5.2.4 demonstrates the vast truncation that can be achieved in the vessel and aneurysm models.

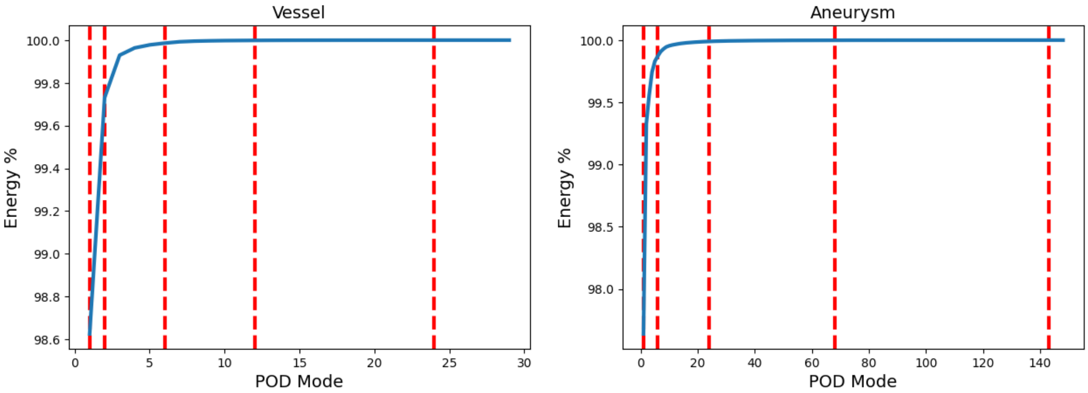


Figure 5.2.4: Cumulative energy contained in the POD modes for the aneurysm and vessel models. The red dashed lines from left to right represent the number of truncated POD modes in order to capture 99%, 99.9%, 99.99%, 99.999% and 99.9999% of the energy in the system, which is defined by the cumulative sum of the squares of all singular values. The total number of singular values is $M \times N = 27 \times 101 = 2727$, demonstrating the vast reduction following the truncation.

Low-Dimensional Representation

Now that the order of the system has been reduced, equation (5.2.7) can be modified to

$$\boldsymbol{u}^n(\boldsymbol{\alpha}_m) = \sum_{j=1}^{r} T_j^n(\boldsymbol{\alpha}_m)\boldsymbol{\Phi}_j, \tag{5.2.11}$$

with the sum up to $j = r$ rather than $j = MN$. Before interpolating between the T_j^n coefficients, they are calculated by exploiting the orthogonality of the POD basis vectors (equation (5.2.12)).

$$\boldsymbol{\Phi}_i^{\mathsf{T}}\boldsymbol{\Phi}_j = \begin{cases} 1, & \text{if } i = j \\ 0, & \text{otherwise} \end{cases} \tag{5.2.12}$$

For each parameter configuration in turn, the inner product is taken of the transpose of each $\boldsymbol{\Phi}$ mode with both sides of equation (5.2.11). To exploit orthogonality, each $\boldsymbol{\Phi}$ mode is looked at in turn. Taking only the first $\boldsymbol{\Phi}$ mode ($j = 1$), equation (5.2.11) is now simply

$$\boldsymbol{u}^n(\boldsymbol{\alpha}_m) = T_1^n(\boldsymbol{\alpha}_m)\boldsymbol{\Phi}_1. \tag{5.2.13}$$

Multiplying both sides of (5.2.13) by the transpose of $\boldsymbol{\Phi}_1$ gives

$$\boldsymbol{\Phi}_1^\intercal \boldsymbol{u}^n(\boldsymbol{\alpha}_m) = \boldsymbol{\Phi}_1^\intercal T_1^n(\boldsymbol{\alpha}_m)\boldsymbol{\Phi}_1, \tag{5.2.14}$$

however, since T_1^n are simply multiplying factors it is possible to use multiplicative commutativity to rewrite the equation as

$$\boldsymbol{\Phi}_1^\intercal \boldsymbol{u}^n(\boldsymbol{\alpha}_m) = T_1^n(\boldsymbol{\alpha}_m)\underbrace{\boldsymbol{\Phi}_1^\intercal\boldsymbol{\Phi}_1}_{=1} \tag{5.2.15}$$

This leaves an equation for T_1^n:

$$\underset{1\times N}{T_1^n(\boldsymbol{\alpha}_m)} = \underset{1\times D}{\boldsymbol{\Phi}_1^\intercal}\, \underset{D\times N}{\boldsymbol{u}^n(\boldsymbol{\alpha}_m)}. \tag{5.2.16}$$

Each coefficient in T_1^n gives the weighting of the first $\boldsymbol{\Phi}$ mode at timestep n. A vector of T_1^n coefficients can be found for each parameter configuration for the first $\boldsymbol{\Phi}$ mode. These coefficients are arranged into a second snapshot matrix $\boldsymbol{S}_j$, as per Walton et al. [146]:

$$\boldsymbol{S}_j = \begin{bmatrix} T_j^1(\boldsymbol{\alpha}_1) & \dots & T_j^1(\boldsymbol{\alpha}_M) \\ \vdots & \dots & \vdots \\ T_j^N(\boldsymbol{\alpha}_1) & \dots & T_j^N(\boldsymbol{\alpha}_M) \end{bmatrix}. \tag{5.2.17}$$

Each column of $\boldsymbol{S}_j$ gives the values of the T_1^n coefficients at time $n = 1, \dots, N$, for a given parameter configuration $\boldsymbol{\alpha}_m = \boldsymbol{\alpha}_1, \dots, \boldsymbol{\alpha}_M$, for a given POD mode $\boldsymbol{\Phi}_j$. A distinct $\boldsymbol{S}_j$ is found for each $\boldsymbol{\Phi}$ mode. The goal is now to interpolate between these $\boldsymbol{S}_j$ matrices to find the representation for a new set of parameters, $\boldsymbol{\alpha}_k$.

Performing a second SVD on $\boldsymbol{S}_j$ gives a second set of POD modes, denoted $\boldsymbol{\Psi}_i^j$. This presents an opportunity for a second truncation of POD modes to further reduce the order of the system and simplify the interpolation. Similarly to the previous case, is it possible to reconstruct each column of $\boldsymbol{S}_j$ using

$$\boldsymbol{T}_j(\boldsymbol{\alpha}_m) = \sum_{i=1}^{r_Q} Q_i^j(\boldsymbol{\alpha}_m)\boldsymbol{\Psi}_i^j, \tag{5.2.18}$$

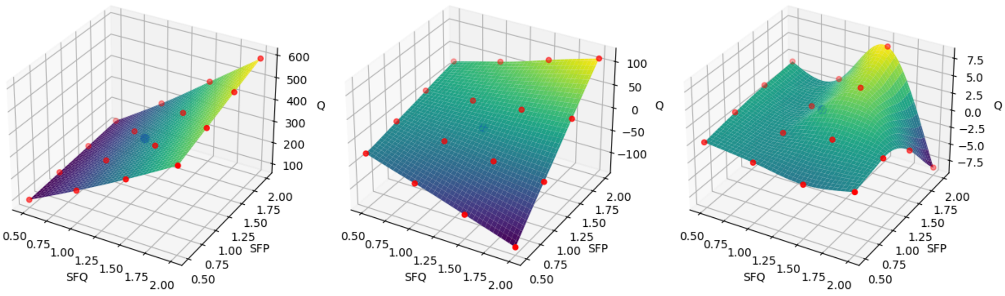
where r_Q is the number of second POD modes retained following truncation. The purpose of this second SVD is to split $\boldsymbol{T}_j(\boldsymbol{\alpha}_m)$ into parameter-dependent terms, $Q_i^j(\boldsymbol{\alpha}_m)$, and parameter-independent terms, $\boldsymbol{\Psi}_i^j$. This means that only the Q_i^j values need to be interpolated between to find a low-dimensional representation of the solution at a new parameter configuration. The Q_i^j coefficients can be calculated, similarly to the T_j^n coefficients previously, using

$$\underset{M\times M}{Q_i^j(\boldsymbol{\alpha}_m)} = \underset{M\times N}{\boldsymbol{\Psi}_i^{j\top}} \underset{N\times M}{\boldsymbol{S}_j(\boldsymbol{\alpha}_m)}. \tag{5.2.19}$$

Each column of Q_i^j gives the weighting coefficient for the corresponding $\boldsymbol{\Psi}_i^j$ mode. A parameter space is then constructed for the Q coefficients for each $\boldsymbol{\Psi}$ mode and the data points are interpolated to find Q for a new parameter configuration.

Neural Network Interpolation

Linear, cubic spline or radial basis function interpolation schemes are all possible approaches to use for interpolating the ROM coefficients. Figure 5.2.5 shows interpolation surfaces for different $\boldsymbol{\Phi}$ and $\boldsymbol{\Psi}$ modes for a two-parameter problem that was used in initial testing of the POD-Interpolation ROM. The Q values can be seen to decrease in magnitude as the $\boldsymbol{\Psi}$ mode increases.



(a) First $\boldsymbol{\Phi}$ mode, first $\boldsymbol{\Psi}$ mode.

(b) First $\boldsymbol{\Phi}$ mode, second $\boldsymbol{\Psi}$ mode.

(c) First $\boldsymbol{\Phi}$ mode, third $\boldsymbol{\Psi}$ mode.

Figure 5.2.5: Q surfaces for the first three Ψ modes and the first Φ mode. Red points show where the full-order CFD simulation data points lie, and the surface is generated by interpolating over these red points onto a finer grid (cubic spline interpolation). The blue point at $SFQ = SFP = 1.25$ represents the parameter configuration used for preliminary testing.

Performing interpolation as shown in Figure 5.2.5 can become more complex as the number of parameters increases. With the development of machine learning, researchers often turn to neural networks for interpolation as they excel at approximating functions mapping from high-dimensional inputs to outputs. In this study, the inputs are the scale factors (and potentially time) and the outputs are the corresponding ROM coefficients (either T or Q values). Three different networks were constructed, highlighted in Figure 5.2.6.

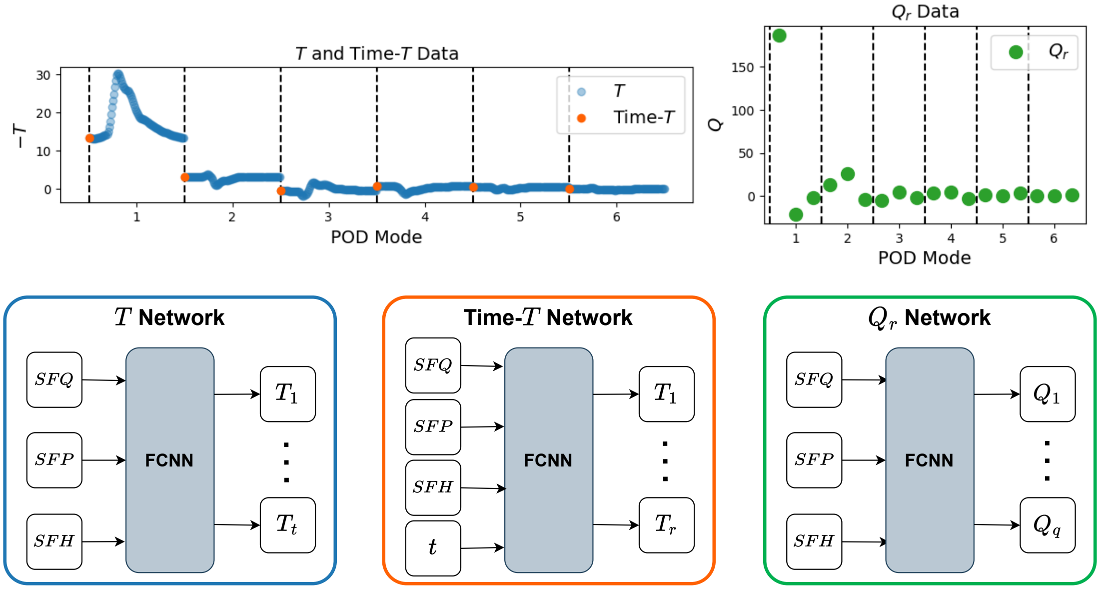


Figure 5.2.6: Three fully-connected neural network (FCNN) designs for the POD-Interpolation ROM that map from simulation parameters $\{SFQ, SFP, SFH, t\}$ to ROM coefficients $\{T, Q\}$. Exemplar T data for one simulation and six POD modes is shown top left, with blue data points output by the T network and orange data points output by the Time-T network, which correspondingly includes time as an additional input parameter. Q_r data for six POD modes and three second POD modes is shown top right and is output by the Q_r network.

The number of neurons in the output layer of each network in Figure 5.2.6 is different. The T network has $r \times N$ outputs, where r is the number of truncated POD modes and N is the number of timesteps, the Time-T network has r outputs, and the Q_r network has $r \times r_Q$ outputs, where r_Q is the number of modes retained in the second POD truncation. An in-depth hyperparameter study for the network parameters was performed, details and results for which can be found in Sections 5.3.2 and 5.3.3.

Once the networks are optimised and trained, the goal is to evaluate the ROM for a previously unseen parameter configuration $\boldsymbol{\alpha_k}$. Inputting $\boldsymbol{\alpha_k}$ into the Q_r network gives us $Q(\boldsymbol{\alpha_k})$. The corresponding $\boldsymbol{T}$ values can then be found by multiplying by the stored $\boldsymbol{\Psi}$ modes:

$$\boldsymbol{T}_j(\boldsymbol{\alpha_k}) = \sum_{i=1}^{r_Q} Q_i^j(\boldsymbol{\alpha_k})\boldsymbol{\Psi}_i^j. \tag{5.2.20}$$

In this case of the T and Time-T networks, the new parameter configuration is input to those networks to calculate the above $\boldsymbol{T}$ values directly. The new $\boldsymbol{T}$ values are then multiplied by the stored $\boldsymbol{\Phi}$ modes and summed over r to give the solution for the new parameter configuration:

$$\boldsymbol{u}^n(\boldsymbol{\alpha_k}) = \sum_{j=1}^{r} T_j^n(\boldsymbol{\alpha_k})\boldsymbol{\Phi}_j. \tag{5.2.21}$$

This model allows for quick evaluation of the solution field for new values of SFQ, SFP and SFH, which is essentially for new physiological conditions.

To highlight and concisely state the steps involved in the POD-NN ROM, Algorithms 1 and 2 are presented for the offline and online stages of the ROM, respectively.

Algorithm 1 POD-NN Offline Phase

```
Use FOM to simulate a number of parameter configurations $\boldsymbol{\alpha}_m$
Construct snapshot matrix $\boldsymbol{U}$.
Economic SVD of $\boldsymbol{U} = \boldsymbol{\Phi\Sigma V}^*$ to generate POD modes $\boldsymbol{\Phi}_j$.
Retain first $r$ POD modes based upon energy criteria: $E_{\text{frac}} \leq \frac{\sum_{j=1}^{r}\sigma_j^2}{\sum_{j=1}^{M\times N}\sigma_j^2}$.
Take inner product of snapshot matrix and each POD mode to generate time coefficients: $\boldsymbol{u}^n(\boldsymbol{\alpha}_m) = \sum_{j=1}^{r} T_j^n(\boldsymbol{\alpha}_m)\boldsymbol{\Phi}_j$.
Construct time coefficient matrix $\boldsymbol{S}_j$ for each POD mode.
SVD of $\boldsymbol{S}_j$ to generate $Q_i^j(\boldsymbol{\alpha}_m)$ and $\boldsymbol{\Psi}_i^j$.
Save the retained POD modes $\boldsymbol{\Phi}_r$, all $\boldsymbol{\Psi}_i^j$ modes and low-dimensional representations $Q_i^j(\boldsymbol{\alpha}_m)$ of each full order simulation.
Train one of the following FCNNs: (i) $T$ Network: $(SFQ, SFP, SFH) \mapsto T_t$; (ii) Time-$T$ Network: $(SFQ, SFP, SFH, t) \mapsto T_r$; (iii) $Q_r$ Network: $(SFQ, SFP, SFH) \mapsto Q_r$
```

Algorithm 2 POD-NN Online Phase

```
1: For a new parameter configuration $\boldsymbol{\alpha_k}$, evaluate $T_j^n(\boldsymbol{\alpha_k})$ or $Q_i^j(\boldsymbol{\alpha_k})$ using the trained FCNNs.
2: For $Q_r$ network, calculate $T$ coefficients for new parameter configuration: $T_j^n(\boldsymbol{\alpha_k}) = \sum_{i=1}^{M} Q_i^j(\boldsymbol{\alpha_k})\boldsymbol{\Psi}_i^j$.
3: Calculate solution for new parameter configuration: $\boldsymbol{u}^n(\boldsymbol{\alpha_k}) = \sum_{j=1}^{r} T_j^n(\boldsymbol{\alpha_k})\boldsymbol{\Phi}_j$.
```

5.3 Results

The results section for the three-parameter POD-NN ROM is outlined as follows. First, preliminary tests and the two-stage hyperparameter study performed on each network design shown in Figure 5.2.6 are presented. The hyperparameter study was only performed for one sampling regime to drastically reduce the number of networks that had to be trained. Furthermore, using the same optimised network parameters for different sampling techniques allows for direct comparison of the ROM with the only difference being in the sampling technique. Second, for the best-performing network design, results are presented to quantify the different sources of error in the ROM, namely the errors due to POD truncation, network generalisation and parameter space sampling regime. Finally, for the best-performing sampling regime and network design, qualitative and quantitative results demonstrating the accuracy and acceleration the approach offers relative to the high-fidelity CFD model are presented.

5.3.1 Preliminary Tests

Given the difference between the three network designs, it is likely that each network will require a different set of hyperparameters to perform optimally. Preliminary tests were performed to identify suitable data processing steps, activation functions and loss calculation methods. Details of the tests performed for data processing and activation function choices can be found in Appendix B.1. From these preliminary tests, good network performance was found for: (i) global normalisation and standardisation of the data; (ii) Leaky ReLU activation functions on hidden layers and Tanh on the output layer; (iii) L_1 training loss (as opposed to L_2). For all networks trained, Adam optimisation was used with a learning rate of 10^{-3} [207] and trained until either 1000 epochs or until the loss plateaued such that the mean loss over the ten most recent epochs is within 10^{-5} of the mean loss over the ten preceding epochs. A batch size of one gave the most accurate results for the T and Q_r networks. A slightly larger batch size was better for the Time-T network (both in terms of improving ROM accuracy and reducing training time), so a batch size of ten was used in this case.

These choices were applied to all networks and then the remaining hyperparameters that the network performance was sensitive to were optimised, namely the network size, the loss function construction, the number of POD modes, and the number of second POD modes in the Q_r network case. Throughout all testing of the networks, the same set of eight unseen test cases were used to evaluate ROM accuracy. For each test case, the velocity magnitude field was evaluated in the region of interest (i.e. vessel/aneurysm) and the L_1 error between the ROM and CFD solution fields was calculated. The mean error was then calculated across the eight test cases and used as the metric for ROM performance.

5.3.2 Hyperparameter Study: Stage 1

For the first hyperparameter study, the parameters listed in Table 5.3.1 were varied through the specified values. Various hidden layer sizes were tested, including a small network with one hidden layer and ten neurons, a wide network with three hidden layers and 1000 neurons per layer, and a deep network with six hidden layers and ten neurons per layer. For each network, single-component and multi-component loss functions were tested. The single component loss was simply the prediction error between the ROM coefficients (T_t, T_r or Q_r) and their ground truth values, whereas the multi-component loss also included the prediction error for the solution field compared to its ground truth. For the T and Q_r networks, the full temporal solution field can be predicted in each training iteration, so the multi-component loss function evaluated the error on the full spatiotemporal solution field. The Time-T network only predicts the solution for a batch of ten timesteps, so for this network the multi-component loss only calculated the solution field error for these ten timesteps. The number of POD modes used in the model was varied according to the energy truncation criteria. To retain {99%, 99.9%, 99.99%, 99.999%, 99.9999%} requires {1, 6, 24, 68, 143} POD modes for the aneurysm model, whereas the vessel model requires {1, 2, 6, 12, 24} modes. This is shown in Fig. 5.2.4. Note that in the first hyperparameter study, three second POD modes were retained for all Q_r networks, however this parameter is investigated in more detail in the second hyperparameter study.

Table 5.3.1: First hyperparameter study details.

Hyperparameter	Values
Hidden layers	[10], [10, 100], [10, 100, 1000], [100, 100, 100], [1000, 1000, 1000], [10, 10, 10, 10, 10, 10]
Loss construction	Single component, multi-component
No. of POD modes (aneurysm)	1, 6, 24, 68, 143
No. of POD modes (vessel)	1, 2, 6, 12, 24
No. of second POD modes (Q-network only)	1, 3, 5, 10, 20

From the first hyperparameter study results (Figure B.2.1, Table B.2.2, Table B.2.1), the following conclusions are drawn: (i) Smaller network sizes ([10], [10, 100]) provide the best results for the T and Q_r networks. For the Time-T network, larger networks provide the best results ([1000, 1000, 1000]). (ii) Loss construction is the least important of the hyperparameters investigated, with similar results for single- and multi-component loss functions. However, the multi-component loss does produce a smaller minimum error in the Q_r network. (iii) The accuracy is best for somewhere around 99.9%-99.99% energy retention. For the T and Time-T networks, adding more POD modes increases the error significantly, whereas the Q_r network can retain more POD modes without as much accuracy degradation.

5.3.3 Hyperparameter Study: Stage 2

Based on the conclusions from the first hyperparameter study, the hyperparameter ranges in the second study were restricted to those presented in Table 5.3.2. Note that for the Q_r network, an additional hyperparameter is investigated for the number of second POD modes retained. Following the second hyperparameter study, results for which can be seen in Tables B.3.1 and B.3.2, the optimal network configurations were identified (Table 5.3.3).

Table 5.3.2: Second hyperparameter study details.

Hyperparameter	Values
T Network	
Hidden layers	[10], [10, 100]
Loss construction	Multi-component
Number of POD modes (aneurysm)	4, 8, 10, 12, 18
Number of POD modes (vessel)	4, 8, 10
Time-T Network	
Hidden layers	[10, 100, 1000], [1000, 1000, 1000], [1000]
Loss construction	Single-component
Number of POD modes (aneurysm)	4, 6, 8, 12, 18
Number of POD modes (vessel)	2, 4, 6, 8, 10, 12
Q_r Network	
Hidden layers	[10], [10, 100]
Loss construction	Multi-component
Number of POD modes (aneurysm)	12, 16, 20, 28, 32
Number of POD modes (vessel)	4, 8, 10, 12, 16
Number of second POD modes	1, 3, 5, 10

Table 5.3.3: Optimal hyperparameters for each network design in each region of interest.

Model	Network	Network Size	Loss Construction	No. of POD Modes	No. of second POD Modes
Vessel	T	[10, 100]	Multi-component	4	NA
	Time-T	[1000, 1000, 1000]	Single-component	4	NA
	Q_r	[10]	Multi-component	12	5
Aneurysm	T	[10]	Multi-component	10	NA
	Time-T	[10, 100, 1000]	Single-component	6	NA
	Q_r	[10]	Multi-component	16	3

5.3.4 Error Quantification

Three main sources of error exist in the ROM: (i) POD truncation error, (ii) network generalisation error and (iii) snapshot sampling error. POD truncation error depends upon the number of POD modes retained in the ROM (and the number of second POD modes retained for the Q_r network). In traditional ROMs, increasing the number of POD modes will typically lead to increased accuracy and reduced efficiency. However, for ROMs that use neural networks for coefficient interpolation, there is additional error dependent on how well the network is able to generalise. In a POD-NN ROM, a compromise must be found between the truncation and generalisation errors. The final source of error depends on how the parameter space was sampled. Again, a balance must be struck between the up-front cost to generate the snapshots and the demand for sufficient snapshots to adequately capture the solution variability throughout the parameter space.

POD Truncation and Network Generalisation Errors

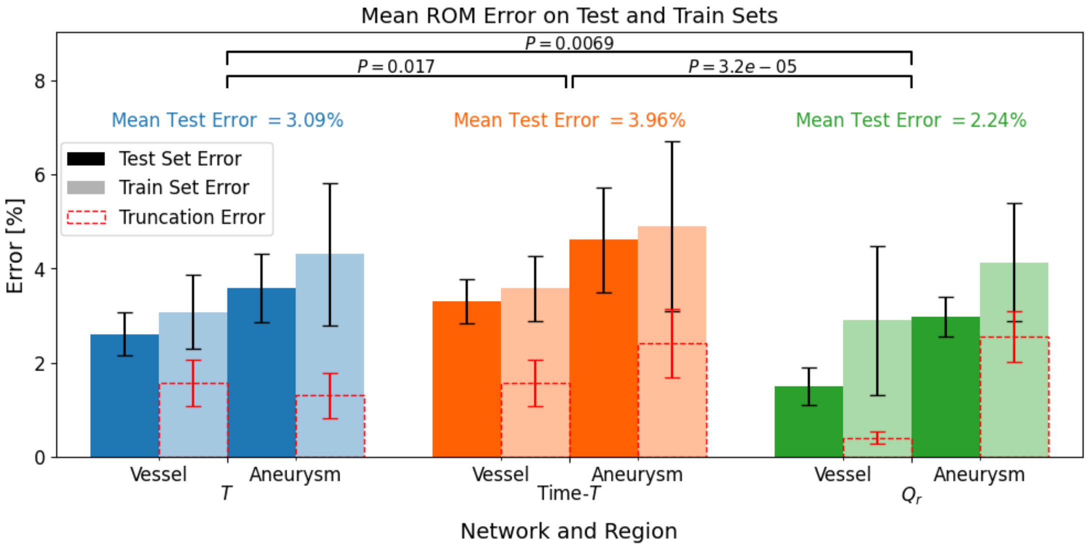


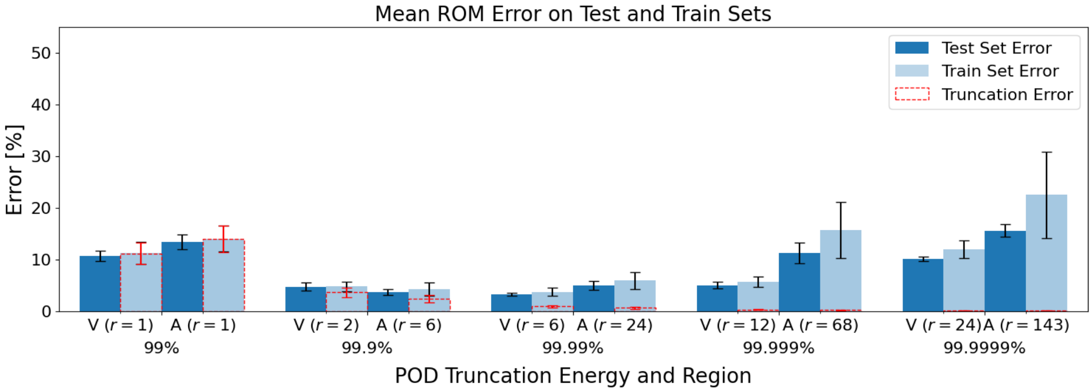
Figure 5.3.1: Mean ROM testing errors on the test and train sets of eight and 27 parameter configurations, respectively, for each of the three network designs. The truncation error refers to the error due to the POD truncation. The remainder of the error above the red dashed bar is attributed to the network generalisation error. The black error bars represent the standard deviation of the error across all parameter configurations for that network, region and data set. P values are calculated using a T-test for the means of independent samples.

For the optimal network designs presented in Table 5.3.3, model performance was evaluated on the test and training sets of data for vessel and aneurysm regions. The contributions of the POD truncation and network generalisation errors can be quantified for the training data test cases, as the ground truth values for T and Q were previously calculated for these parameter configurations. Calculating the solution using the ground truth T and Q values means there is no network generalisation error and therefore the error is entirely due to the POD truncation. The network generalisation error was then calculated as the difference between the ROM error using the network predicted and ground truth T and Q coefficients.

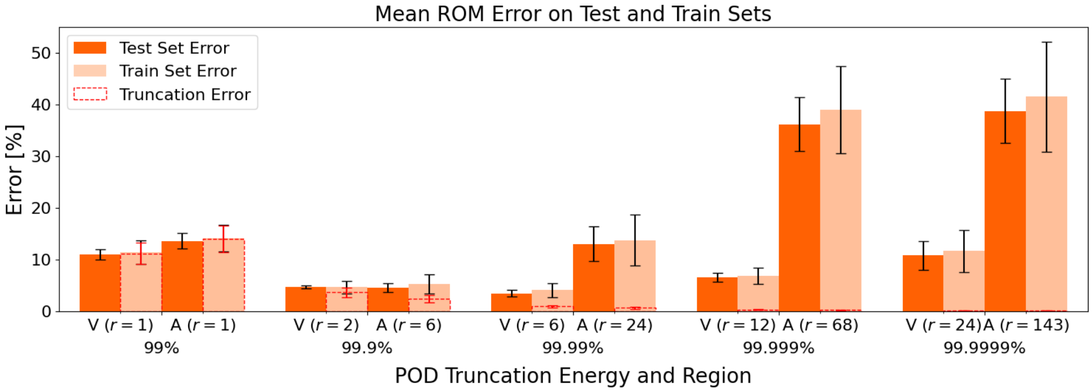
Figure 5.3.1 shows the ROM errors for each network design for the testing and training data sets for each geometry. The Q_r network was found to significantly outperform the T and Time-T networks. The T network performed second best and the Time-T network third best. All networks achieve mean test errors of $< 5\%$ in all geometries and for all data sets. In most cases, the network generalisation error contributes more to the total error than the POD truncation error.

To understand why the Q_r network gives the most accurate results, networks were trained with different values of POD truncation energy to see how the weighting between POD truncation and network generalisation error varies. The optimal network parameters (Table 5.3.3) were used but with varying values of r based on the number of POD modes required to retain 99%, 99.9%, 99.99%, 99.999% and 99.9999% of the cumulative energy. For the Q_r network, the same number of second POD modes ($r_Q = 3$) was used so that the variation in truncation error with additional POD modes would not be affected by this parameter for the different regions.

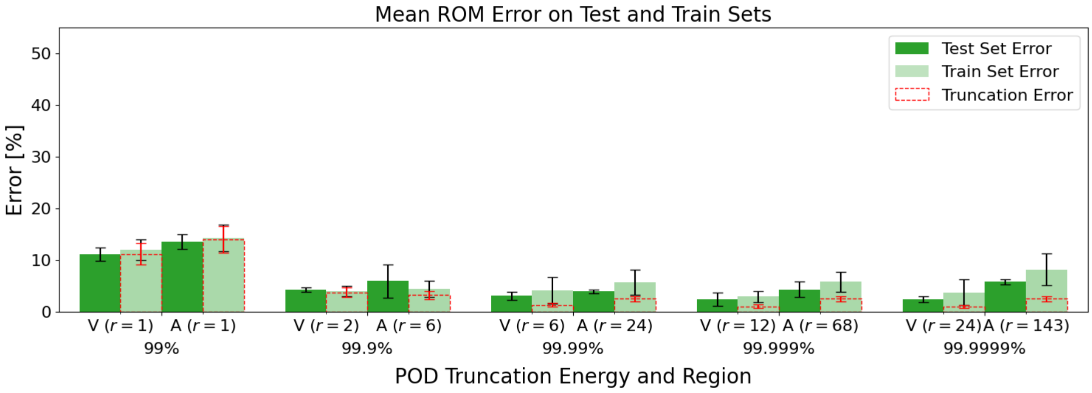
Figure 5.3.2 shows that the POD truncation error dominates the total error when the number of POD modes retained is small. Adding more POD modes decreases the POD truncation error but increases the network generalisation error due to the increased number of parameters in the output layer of the network. The POD truncation error does not decrease as much for the Q_r network as for the T and Time-T networks. Despite this, the total error remains lower for the Q_r network as the network generalisation error does not grow as large as it does for the other networks.



(a) T network.



(b) Time-T network.



(c) Q_r network.

Figure 5.3.2: Mean ROM testing errors on the test and train sets of 8 and 27 parameter configurations, respectively, for each of the three network designs with different values for the POD truncation energy threshold. r is the number of POD modes for a given energy cut-off and region. V and A refer to the vessel and aneurysm geometries, respectively.

Sampling Method Error

For the Q_r network, which provided the best accuracy of the three network designs tested (Fig. 5.3.1), a new network was trained for each different parameter space sampling regime. Five sampling methods were investigated, namely Uniform Grid, Latin Hypercube With Bounds, Latin Hypercube Without Bounds, Random With Bounds and Random Without Bounds. For each of the Latin Hypercube and Random sampling methods, three distinct data sets were generated and a Q_r ROM was trained for each data set. Results are presented from the best-performing sampling regime among the three distinct data sets for each sampling regime.

Figure 5.3.3 shows the ROM test errors on testing and training data sets for the uniform grid sampling approach and the best-performing Latin Hypercube (with and without bounds) and Random (with and without bounds) sampling regimes. The performance is similar across all regimes, with only a slight improvement upon uniform grid sampling for the best-performing sampling regime, which was Latin Hypercube with bounds. Despite using the same number of POD modes for each sampling regime, it is notable that the POD truncation error varies from regime to regime. This suggests that in some cases the sampling regime leads to a better distribution of snapshots across the parameter space.

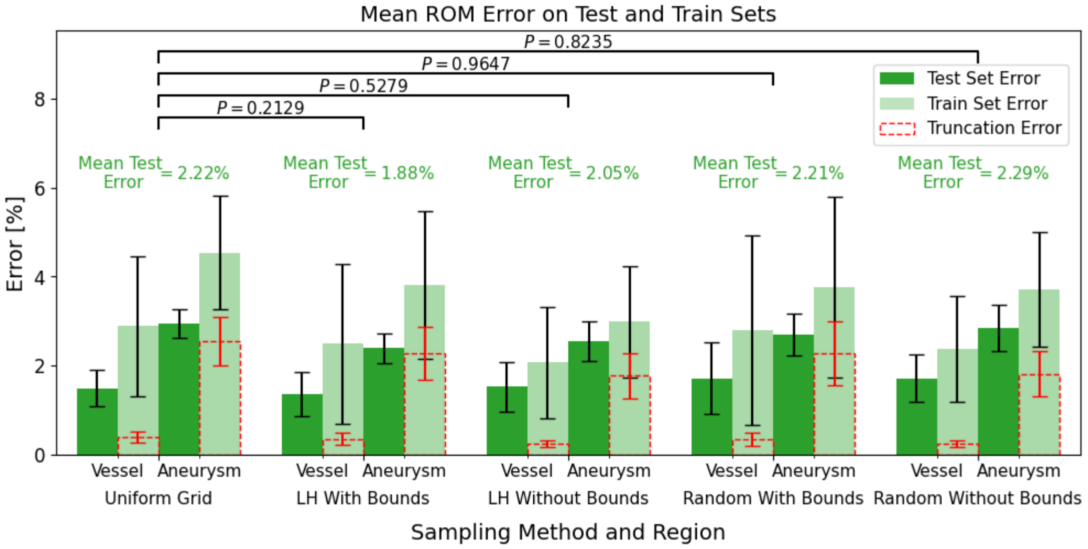


Figure 5.3.3: Mean Q_r network ROM testing errors on the test and train sets of eight and 27 parameter configurations, respectively, for different parameter space sampling regimes. The truncation error refers to the error due to the POD truncation. The remainder of the error above the red dashed bar is attributed to the network generalisation error. The black error bars represent the standard deviation of the error across all parameter configurations for that network, region and data set. P values are calculated using a T-test for the means of independent samples.

5.3.5 Best-Performing Model

To further demonstrate the quantitative accuracy of the ROM predictions, as well as show the qualitative agreement between ROM and FOM, several visualisations and additional figures were created using the best-performing ROM approach. These extra results are only presented for the aneurysm model as the flow is more complex in this scenario. The best-performing ROM was the Q_r ROM using Latin Hypercube sampling with bounds included. Figure 5.3.4 shows the velocity magnitude field in a 2D slice of the aneurysm geometry at systole and diastole. The relative error for the velocity magnitude U is calculated as:

$$\text{Relative Error} = \frac{U_{ROM} - U_{FOM}}{\overline{U}_{FOM}} \times 100\%, \tag{5.3.1}$$

where $\overline{U}_{FOM}$ is the space-and-time-averaged FOM velocity magnitude.

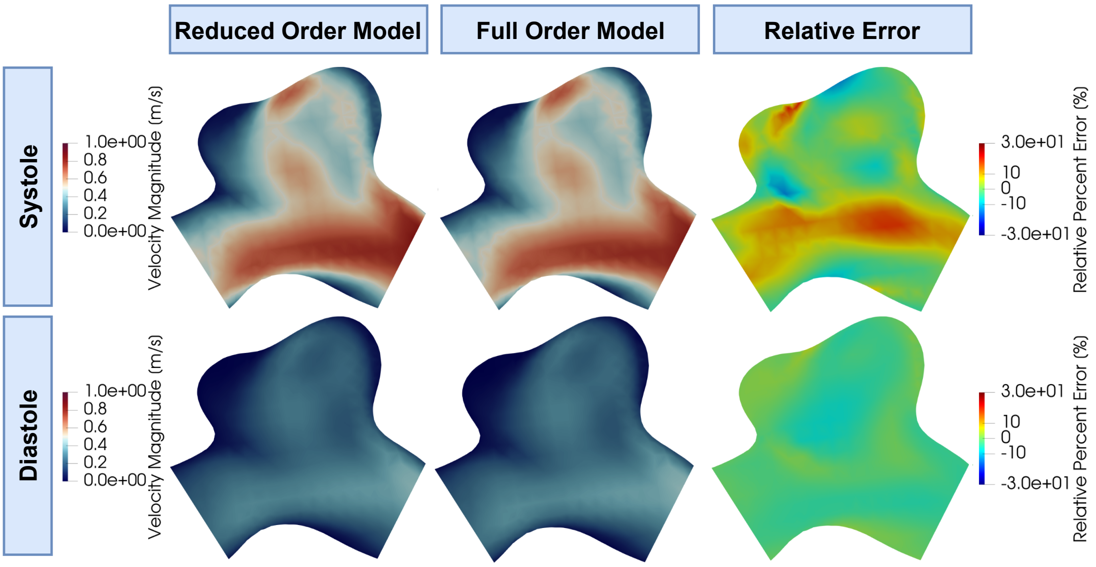
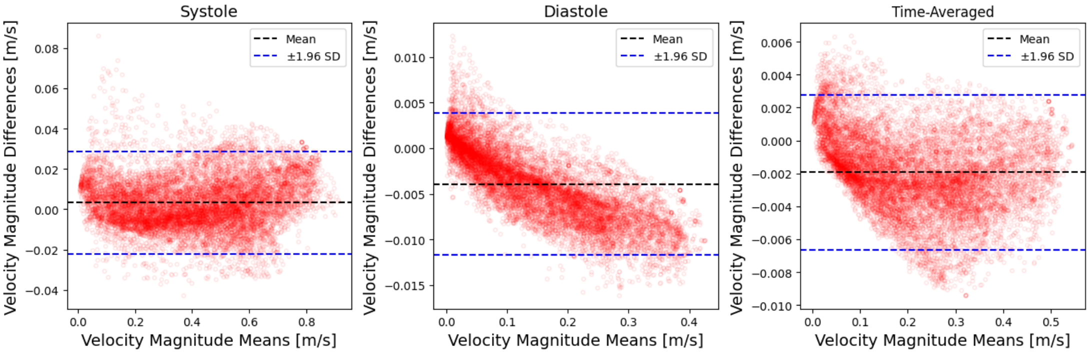
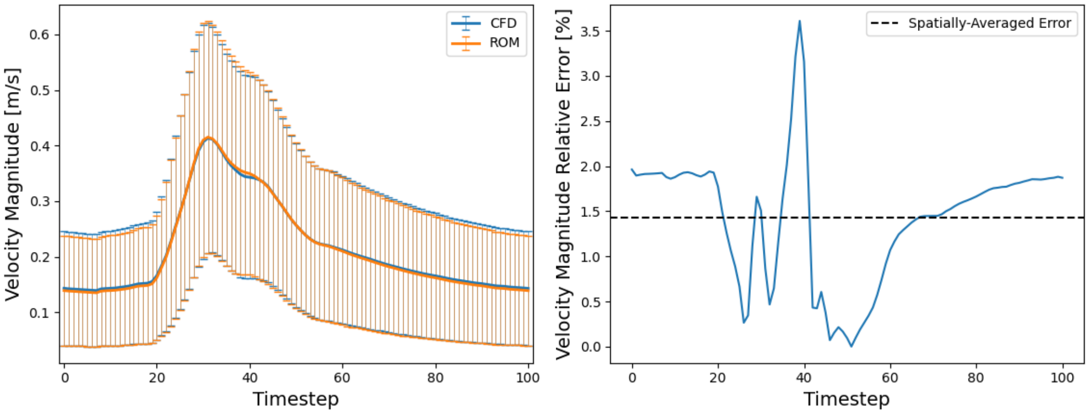


Figure 5.3.4: Visualisations of systolic (timestep 30) and diastolic (timestep 90) ROM and CFD velocity magnitude fields and the relative error between them in a 2D slice of the aneurysm geometry for a median error test case using the Q_r network.

The largest relative errors occur at systole ($\sim \pm 30\%$). Figure 5.3.4 demonstrates the qualitative agreement between ROM and FOM at systole and diastole. A median error test case with an error of 2.2% was chosen as a representative case for Figure 5.3.4. Across the eight test cases, the mean error was 2.4±0.3% and the maximum error was 3.0%, which highlights the suitability of the median case to represent the overall performance of the ML-ROM. Figure 5.3.5 shows Bland-Altman plots in the aneurysm region for systolic (timestep 30), diastolic (timestep 90) and time-averaged ROM and CFD velocity magnitude. At systole, the mean difference line (black dashed) is above the $y = 0$ line, which implies that the ROM slightly over-predicts the solution at this time. Conversely, the ROM slightly under-predicts the diastolic and time-averaged solutions. However, the Bland-Altman points largely fall in close proximity to the $y = 0$ line and the scale of the differences is smaller than that of the means, implying low errors. Figure 5.3.5 also shows spatially-averaged velocity magnitude field waveforms for ROM and FOM, the standard deviation across the spatial points and the absolute relative error between the two waveforms. These waveforms show good agreement throughout the cardiac cycle, with the largest discrepancy in velocity magnitude occurring in the post-systolic period of the cardiac cycle, where the ROM slightly smooths the gradients of the CFD waveform. The Bland-Altman plots in Figure 5.3.5 demonstrate the agreement between the spatially-varying ROM and CFD velocity magnitude fields at different time points and for the time-averaged fields. The waveform plots in Figure 5.3.5 demonstrate the agreement between the temporally-varying solutions that are averaged in space.



(a) Bland-Altman plots for systolic (timestep 30), diastolic (timestep 90) and time-averaged velocity magnitude. The means of the ROM and CFD velocity magnitude at the spatial points in the mesh are plotted on the x-axis. The differences between the ROM and CFD velocity magnitude fields ($U_{\mathrm{ROM}} - U_{\mathrm{CFD}}$) at the spatial points in the mesh are plotted on the y-axis. The blue dashed lines represent a 95% confidence interval for the differences and the black dashed line represents the mean difference between the ROM and CFD solutions.



(b) Velocity magnitude waveforms for ROM and CFD (left) and the error between them (right). The error is calculated by taking the absolute values of the relative error (equation 5.3.1).

Figure 5.3.5: Bland-Altman plots for ROM and CFD velocity magnitude at systole, diastole and for the time-averaged fields and spatially-averaged waveform plots in the aneurysm model for a median error test case using the best-performing Q_r network.

5.3.6 Acceleration, Data Storage Reduction and Training Times

Acceleration

ROM speed was quantified by calculating the mean run-time across the eight test cases and the 27 train cases for each network design. Across all network designs and both geometries, the mean ROM run-time was 0.037 seconds, which highlights that the POD-NN ROM methods can provide very fast solution inference. The ROM took approximately twice as long in the aneurysm as in the vessel, which is likely due to the optimal aneurysm models using more POD modes and therefore having more ROM coefficients being output in the final layer of the network, thus slowing down the evaluation speed slightly. To calculate acceleration, the mean ROM run-time was compared with the time it took to run the FOM locally and on a high-performance computing (HPC) cluster. One local evaluation of the FOM was performed using one CPU. This simulation took 1 hour 39 minutes and 32 seconds. For the HPC simulations, 16 cores were used and the mean run-time across the eight test cases was 25 minutes and 51 seconds. The acceleration of the various ROM designs was then calculated for each geometry relative to these FOM run-times. The results are presented in Table 5.3.4. The ROMs provide acceleration of roughly 10^4–10^5 times. However, it is important to note that the simulation times are for the full geometry, whereas the ROM evaluations are only in smaller regions of interest. This inflates the ROM acceleration factors to some degree.

Table 5.3.4: Acceleration offered by each ROM relative to running the full order model on one CPU locally and on 16 CPUs using a high-performance computer.

Model	Network	Mean Test Time [seconds]	Acceleration Factor* (relative to 1 CPU)	Acceleration Factor* (relative to 16 CPUs)
Vessel	T	0.012	5.1×10^5	1.3×10^5
	Time-T	0.031	1.9×10^5	5.0×10^4
	Q_r	0.022	2.7×10^5	7.1×10^4
Aneurysm	T	0.043	1.4×10^5	3.6×10^4
	Time-T	0.055	1.1×10^5	2.8×10^4
	Q_r	0.062	9.7×10^4	2.5×10^4

*Note that the run-time used to calculate the acceleration factors was taken from a simulation of the full geometry, whereas the ROM run-times are only for the regions of interest.

Data Storage Reduction

Retaining all POD modes in the ROM equates to performing direct interpolation between the high-order solution snapshots. When considering data storage requirements, a comparison can therefore be made between the amount of data that must be stored in order to construct a POD-Interpolation ROM and the data that must be stored in order to perform direct solution interpolation (DSI). DSI requires that all of the full order simulation data is stored, so that each spatiotemporal point can be interpolated. Constructing a POD-Interpolation ROM only requires that the POD modes (and potentially the second POD modes) and the ROM coefficients are stored. The total size of the stored variables for DSI and the ROMs are shown in Table 5.3.5. The amount of data stored by the reduced order models compared to that required for direct solution interpolation is approximately 120 times less for the aneurysm model. This reduction factor would be further increased if the number of POD modes was decreased (for example, for 5 POD modes, the reduction factor would be roughly 380).

Table 5.3.5: Number of variables involved in POD-Interpolation and direct solution interpolation.

Method	Total size of stored data	Aneurysm model*
DSI	$M \times D \times N$	17,272,818
T and Time-T ROMs	$\underbrace{(r \times D)}_{\phi \text{ modes}} + \underbrace{(r \times N \times M)}_{T \text{ coefficients}}$	144,976
Q_r ROM	$\underbrace{(r \times D)}_{\phi \text{ modes}} + \underbrace{(r \times N \times M)}_{\psi \text{ modes}} + \underbrace{(r \times r_Q \times M)}_{Q \text{ coefficients}}$	146,272

D, number of mesh nodes; N, number of timesteps; M, number of parameter configurations; r number of retained first POD modes; r_Q, number of retained second POD modes.

*For the aneurysm model, $D = 6334$, $N = 101$ and $M = 27$. For the calculation, $r = 16$ and $r_Q = 3$, which are the optimal parameters for the Q_r network.

Training Times

All of the networks were trained using a single CPU in Google Colab. The training times for the optimal networks for each geometry and network design are shown in Table 5.3.6. The optimal T network trained fastest, with the Q_r network second fastest and the Time-T network the slowest. The mean and standard deviations for were also calculated across all of the hyperparameters investigated in the first hyperparameter study for each geometry and network design. In this case, the Q_r network trained the fastest, with the T network second fastest and the Time-T network the slowest. All networks were trained with the same stopping criteria, so the Q_r network training faster for a large range of hyperparameters highlights how this network design can typically make network training more efficient.

Table 5.3.6: Network training times for each network design and each problem. The hyperparameters for the optimal network design are those presented in Table 5.3.3. The mean and standard deviations are calculated across all training runs performed in the first hyperparameter study (see Table 5.3.1 for details).

Model	Network	Optimal network training time [mins]	Mean network training time [mins]	SD network training time [mins]
Vessel	T	6	10	9
	Time-T	43	24	22
	Q_r	8	9	6
Aneurysm	T	7	43	46
	Time-T	28	70	77
	Q_r	18	32	30

5.4 Conclusion

A POD-NN ML-ROM has successfully been developed and applied to physiological blood flow variation in intracranial aneurysms. A two-stage POD reduction was found to improve model performance compared to the widely used one-stage POD approach, however similar performance was found for the various sampling regimes tested. The best-performing ROM achieved mean test accuracy of 98.6% and 97.6% in vessel and aneurysm regions, respectively, while providing speed-up of order 10^5 times. This level of acceleration means that the ROM could be used for real-time flow field inference.

CHAPTER 6

Discussion and Conclusion

6.1 Chapter 3: Literature Review

6.1.1 Challenges

Despite years of research on ROMs and the recent application of machine learning to simulation acceleration, applying these techniques to real-world vascular flow problems remains challenging. Three key challenges to address that have been identified by this review are:

1. The development of accelerated simulation methods that can handle large geometric variability, facilitating their application to previously unsimulated and dynamically varying geometries.
2. The development of accelerated simulation methods for multi-scale problems, enabling seamless evaluation of small- and large-scale processes over short- and long-term time scales.
3. The development of a benchmarking framework for accelerated simulation methods, allowing for systematic quantification and comparison of new approaches whilst driving transparent progress in the field.

A critical challenge to widespread adoption of simulation acceleration in vascular flow applications is incorporating large geometric variability into the models. Whether performing large-scale testing of medical devices in cohorts with varying anatomy, simulating medical device responses as part of treatment planning for an individual patient, or providing real-time surgical feedback during operation, the ability of the accelerated model to accurately evaluate haemodynamics in a previously unsimulated or dynamically changing geometries is essential. Efforts to introduce geometric variability into vascular flow ROMs have mainly focused on developing parameterised models [100, 151, 161, 169]. While these approaches yielded accurate results, acceleration was only of one order of magnitude in most cases, with the largest acceleration roughly three orders of magnitude. Furthermore, models typically only used a small number of parameters describing features such as vessel diameter or stenosis severity and position [100]. In pathologies with highly complex shapes, such as aneurysms, identifying descriptive parameterisations with few parameters may not be possible. This would be further exacerbated by device modelling or fluid-structure interaction. A possible approach to overcome this is to use domain decomposition ROMs that can partition an unseen geometry into sub-geometries that bear resemblance to the geometries for which snapshots were previously calculated [319, 320]. This approach has been applied to flow over urban landscapes and pipe flow problems so far, but could potentially be applied to vascular flow problems, where the sub-geometries could be a set of commonly required vascular segments and configurations. ML approaches such as physics-agnostic simulation methods [225, 258, 260] and Point-Nets [108, 109] have demonstrated the ability to generalise to unseen geometries by using large sets of mostly synthetic geometries and corresponding simulation data for training. These are the most promising attempts to provide generalisation across geometries in vascular simulation acceleration, but they are still hampered by the amount of data required and the risk that data augmentation strategies can lead to unrealistic results. Informing these approaches with physics could potentially reduce the data requirement and increase the reliability of the results but there have been few studies into this to date [268].

Multi-scale problems represent the second challenge for accelerated simulation of vascular flow models. When using computational models to inform treatment decisions or in assessing medical device safety and efficacy, short- and long-term metrics are likely to be required. Depending upon the specific problem, models of small-scale processes like thrombosis or endothelialisation may need to be coupled to models of large-scale haemodynamic effects. In principle, DMD ROMs are well-suited to long-term solution evaluation, but the few studies using this approach for vascular flow applications have focused on solution reconstruction rather than long-term prediction [174, 178]. Domain decomposition PINN methods, such as cPINNs, XPINNs, and PPINNs, are suitable for multi-scale problems in time and space, but have also seen little use in vascular flow applications [252–254]. DeepONets have also shown great potential for multi-scale applications. Wang and Perdikaris [274] used DeepONets for long-time prediction of partial differential equations, while Cai et al. [256] and Mao et al. [257] used modular DeepONets trained individually on single-physics single-scale problems to facilitate multi-physics and multi-scale modelling for electroconvection and flow-chemistry applications. Modular DeepONets are referred to as DeepM&MNets (Deep Multi-Physics & Multi-Scale Networks) and represent a promising approach towards the challenge of long-time evaluation of multi-physics and multi-scale models which are crucial in vascular flow applications.

The final challenge highlighted is the need for a benchmarking framework for assessing simulation acceleration methods. Throughout Chapter 3, quantitatively comparing different approaches has proved challenging due to the the following factors that vary across studies: (i) amount of training data; (ii) training details, e.g. stopping/convergence criteria, number of modes retained in model; (iii) accuracy and acceleration metrics, e.g. error metrics and variables of interest, acceleration relative to FOM or entire offline cost; (iv) target applications. To overcome this challenge, the development of a benchmarking framework for use in the simulation acceleration community is proposed. This should consist of a series of example problems of varying nature and complexity, data sets for each example problem for use in training, specified allowances and/or metrics for the computational cost of data generation and training, and metrics defined for assessment of accuracy and acceleration. The example problems should also be motivated by real-world problems where a balance often must be struck between the amount of training data available for the machine learning model and the task for which it is to be used (e.g. many-query tasks, control problems, real time prediction, etc.). Development and subsequent use of this framework would enable objective assessment and comparison of methodological advances in the field. Inspiration could also be taken from the medical image analysis field, where challenge problems are commonly proposed with publicly available data and predefined metrics to assess model performance for tasks like registration and segmentation [321, 322].

6.1.2 Outlook

Accelerated vascular flow models are essential for applications such as in-silico trials (ISTs), patient-specific treatment planning and real-time surgery feedback. ISTs can require the evaluation of nonlinear, multi-physics, multi-scale models in large cohorts of virtual patients, which are anatomically and physiologically diverse, undergoing treatment with different devices [11, 20]. Patient-specific treatment planning requires similarly complex models that can be evaluated for an individual patient in a reasonable time frame given the prognosis of the pathology in question. Real-time surgery feedback requires complex model evaluation for individual patients at a sufficient speed to provide haptic feedback or visualisations to the surgeon performing the procedure [114]. These three applications highlight some of the impact that accurate and efficient vascular flow models can have on patient care, which makes developing these approaches a worthwhile endeavour. This review has identified that the key challenge to be addressed is the development of multi-scale simulation acceleration methods that can handle the large geometric variability inherent to vascular flow problems. To achieve quantifiable and transparent progress in simulation acceleration, it is also suggested that the community should develop a benchmarking framework consisting of a series of exemplar problems with standardised metrics for assessing acceleration and accuracy. This framework would benefit both the simulation acceleration and the vascular flow modelling communities.

6.2 Chapter 4: FD-PComA In-Silico Trial

6.2.1 Discussion

In-silico trials of internal carotid artery (ICA) and posterior communicating artery (PComA) intracranial aneurysm flow diversion (FD) have been demonstrated to replicate and expand upon results from conventional clinical trials [20]. In addition, each component of the FD-PComA in-silico trial modelling pipeline has been validated independently through a series of studies [21, 23, 91, 294, 296, 299]. Through the component-wise validations performed in these studies and the soft validation of the complete modelling pipeline performed by Sarrami-Foroushani et al. [20], there is sufficient trust in the modelling choices to use the FD-PComA pipeline to investigate hypotheses for flow diverter treatment indications that have not been studied in clinical trials and are not well understood. In particular, the FD-PComA in-silico trial investigates Pipeline Embolization Device (PED) flow diversion of PComA aneurysms with and without fetal posterior circulation.

Fetal posterior circulation (FPC) is estimated to occur in 4–29% of the population [283], but its prevalence can vary across different demographics. Shaban et al. [285] retrospectively reviewed 532 PComA aneurysms and provided statistics for the prevalence of FPC in a number of ethnicities. This data was collated and it was found that full FPC prevalence is significantly lower in white patients (4.9%, 8/164) than in black patients (11.5%, 40/349, $p = 0.008$). In patients with FPC, the PComA supplies the posterior cerebral artery and will typically be larger in diameter than in non-fetal patients. This highlights two factors that could reduce flow diverter efficacy in PComA aneurysms in patients with FPC: the increased flow rate and the increased PComA size. The results demonstrate that it is the increased flow rate associated to FPC that is the primary determinant of treatment success. In the FD-PComA in-silico trial, computational fluid dynamics simulations were performed for 118 virtual patients pre- and post-treatment by the PED flow diverter and haemodynamic variables (aneurysm STAV/MTAV and stent STAWSS) were used to assess treatment performance. Significantly lower flow reduction and treatment success rates were found in patients with FPC, which was characterised by an increased flow rate through the PComA. Patients with FPC were also found to have significantly higher STAWSS on the device struts than non-fetal patients. This could inhibit endothelialisation and neointimal proliferation along the device struts [305, 306]. This will aid in maintaining PComA patency following the treatment, which could explain why neurological complications following the treatment are rare [279–281]. By construction, the PComA size distributions were identical in the non-fetal and fetal sub-groups since the same anatomies were used in each. The difference in predicted treatment success is therefore attributable to the difference in flow rates in the sub-groups. Aneurysm morphological characteristics (maximum diameter, aspect ratio, neck width, non-sphericity index) were also identical across the non-fetal and fetal sub-groups, so it is concluded that aneurysm morphology does not affect treatment success.

The results indicate that fetal PComA aneurysms treated by PED flow diversion will have low aneurysm flow reduction due to the persistent flow through the PComA. This could explain the slow or failed occlusion that is observed more predominantly in patients with FPC than in those without [2]. Coiling of PComA aneurysms tends to be relatively uncomplicated compared to aneurysms located in other regions, so in patients with FPC, using endovascular coils in conjunction with flow diversion may improve aneurysm flow reduction and accelerate occlusion [283]. Stent-assisted coiling has already been shown to be effective in treating PComA aneurysms, but further studies are required to ascertain the efficacy of flow diverter-assisted coiling [323].

6.2.2 Limitations

Two limitations in this study relate to the outflow boundary conditions. Firstly, the posterior cerebral artery was not included in the simulation geometries for the 59 patient anatomies used in the trial. FPC was modelled using boundary conditions that mimic flow conditions for non-fetal and fetal posterior circulation, but including the posterior cerebral artery explicitly where possible could allow for additional analysis and further stratification of the cohort (e.g. comparisons between true fetal and fetal-type variants). Secondly, the same PComA flow split was used for all patients, which meant that the study could not account for any intra- and inter-patient variability that may be present in the flow split.

Biological processes such as stasis/device-driven thrombosis and PComA remodelling were also not modelled. Stasis-driven thrombosis plays an important role in aneurysm treatment success and clot composition has been linked to ischaemic/haemorrhagic stroke. Device-induced thrombosis is an alternative clotting pathway that links to endothelial cell growth along the device, which is another component of successful flow diverter treatment [304]. PComA remodelling can occur as a long-term result of flow diverter treatment affecting the PComA flow rate and the vessel subsequently adapting to the increased/decreased flow. Modelling stasis/device-induced thrombosis would require coupling biochemical models to the blood flow model, such as in [23], but this was not deemed necessary to address the FD-PComA trial hypotheses. In FD-PComA, PComA flow rates were assumed to be the same pre- and post-treatment to address the hypotheses regarding fetal posterior circulation, which meant that investigating remodelling was not feasible. Addressing each of these limitations could form the basis for additional in-silico trials.

Finally, a methodological limitation in the calculation of STAWSS on the stent surface is noted. In this calculation, the WSS field is evaluated for all timesteps and all mesh points on the surface of the clipped stent. This will lead to the results being somewhat dependent on the size of the clipped stent. Oversized clipped stents are likely to have more of the struts lying in close proximity to the vessel walls, which is likely to lead to lower WSS values in these regions and therefore a reduction in the mean WSS for this configuration. Performing the clipping manually helped to reduce the impact that this effect has on the overall results due to the more optimal sizing that can be achieved. However, the results could have been made more accurate by identifying a sub-region of the stent that lies adjacent to the neck surface and only using WSS values from this region in the calculation of STAWSS instead of using WSS values evaluated on the entire clipped stent.

6.2.3 Outlook

FD-PComA has demonstrated the power of in-silico trials for investigating less-studied scenarios that may not be analysed in clinical trials, such as PED treatment of PComA aneurysms with FPC. The finding that FPC is more prevalent in certain demographics highlights how in-silico trials can be used to generate evidence in groups that may be under-represented in clinical trials. For these two reasons, FD-PComA represents an exemplar trial that demonstrates the usefulness of in-silico trials, which is one of the challenges that this thesis aimed to address.

6.3 Chapter 5: Machine Learning Reduced Order Model

6.3.1 Discussion

In Chapter 5, a non-intrusive parametric machine learning reduced order model (ML-ROM) using proper orthogonal decomposition (POD) for dimensionality reduction and neural networks (NN) for interpolation of the ROM coefficients was developed. The POD-NN ML-ROM was applied to intracranial aneurysm fluid dynamics with physiological variability characterised by inflow parameters for flow magnitude (SFQ), pulsatility (SFP) and heart rate (SFH). Typically POD-NN ROMs utilise a single POD reduction and evaluate the ROM coefficients using a network that takes the varying parameters (i.e. SFQ, SFP, SFH) and the time coordinate as inputs and outputs the corresponding coefficients [167, 215, 216]. This approach (referred to as the Time-T network) and two other approaches were investigated, one that outputs the entire time trajectory of coefficients in one forward pass through the network (T network) and another that performs a second POD reduction and trains using the doubly reduced coefficients (Q_r network). The latter approach has not yet been used in POD-NN ROMs and is rarely used even in POD methods that do not utilise machine learning [146].

The findings indicate that the Q_r network outperforms the T and Time-T networks. For the vessel model, the best-performing Q_r network achieves an error of 1.50%, whereas the T network achieves 2.61% and the Time-T network achieves 3.31%. For the aneurysm model, the best-performing Q_r network achieves an error of 2.97%, whereas the T network achieves 3.58% and the Time-T network achieves 4.61%. In addition, the Q_r network appears to be more robust to the number of modes retained in the model. Figure 5.3.2 shows that the ML-ROM errors grow as an increasingly large number of POD modes are added to the model, particularly for the T and Time-T networks. The POD truncation error decreases with additional modes as expected but the network generalisation error grows at a faster rate and so the total error increases. For the Q_r network, the truncation error does not decay as much as for the T and Time-T networks, but in turn the network generalisation error does not grow so rapidly. The double-POD performed in the Q_r network therefore seems to make the network more robust to the number of POD modes retained. This makes the network more straightforward to optimise as there are a greater range of hyperparameters that can give close to optimal performance.

While the results demonstrated that the Q_r network performs best, a more extensive and therefore costly investigation into optimal network design could have led to improved results for the T and Time-T networks. In principle, the minimum error a POD-NN ROM can achieve is bounded by the POD truncation error (assuming the interpolation error approaches zero). The truncation error is larger in the Q_r network case than the T or Time-T cases, so in theory it should be possible to train the latter two approaches to a higher accuracy. In practice, a balance must be struck between the ML-ROM accuracy and the resources put into searching for the optimal network. If the aim is to streamline the network optimisation process, then the findings demonstrate that using a two-stage POD reduction can provide accurate results with a more straightforward network optimisation procedure. Given that simplifying the ML-ROM data appears to improve the network robustness, it may be that using more advanced machine learning-based dimensionality reduction tools such as autoencoders could further simplify the training data and further improve ML-ROM performance or simplify training [223].

Each ML-ROM that was investigated provided a speed-up of order 10^5 relative to the full-order computational fluid dynamics simulation performed on one CPU. ML-ROM evaluation time was less than a tenth of a second for all network designs, although each evaluation took approximately twice as long in the aneurysm as in the vessel. This is likely because the optimal aneurysm models use more POD modes and therefore have more coefficients to output in the final layer of the network and more calculations to perform in order to produce the solution field. In a vascular flow context, the ML-ROM solution fields can be calculated quicker than the period of the heartbeat, which constitutes a model that is fast enough to respond to real-time measurements of physiological changes. Using a POD-NN ROM also reduces the amount of stored data by approximately 100 times compared with retaining full-order model solutions and directly interpolating them. This means that the POD-NN ROM provides fast and accurate predictions while requiring minimal storage capacity.

6.3.2 Outlook

The ML-ROM could be useful in scenarios such as inverse modelling (e.g. to identify boundary conditions that produce outputs matching experimental data [216]), design optimisation (e.g. identifying input parameters that induce desired flow features) or in uncertainty quantification (i.e. analysing model output sensitivity to its inputs). An example use of the ROM for an in-silico trials application could be for assessing the sensitivity of the flow fields in a given geometry to the ICA-PComA flow split ratio used for the non-fetal and fetal boundary conditions that were developed in Chapter 4. The ML-ROM could also be suitable for real-time flow monitoring in various applications due to its speed. However, in its current design, the ML-ROM is not suitable for problems where capturing geometric variability is essential. Two examples of this in a vascular flow context are in real-time surgical feedback where the geometry changes dynamically and in performing in-silico trials where flow needs to be evaluated in previously unsimulated geometries at low cost. Some attempt has been made to include geometric parameters into POD-NN ROMs but only for relatively simple geometries [212]. In the wider ROM field, higher dimensional parameterisations have been used for vascular flow applications but these are often limited in their generality [100]. Due to the limitations of the ML-ROM in evaluating flow in new geometries, the approach is not seen as an alternative to traditional numerical methods for performing complete in-silico trials, but instead as a tool that could be used for uncertainty quantification in problems related to in-silico trials.

6.4 Conclusion

This thesis set out to achieve the following aims:

1. Perform an in-silico trial investigating flow diverter treatment of intracranial aneurysms and automate the steps involved in the simulation pipeline.

2. Critically review accelerated simulation methodologies for vascular flow problems, identifying the types of application each method is suitable for and directions for future research.

3. Develop accelerated simulation methodology to facilitate efficient in-silico trials of endovascular medical devices and demonstrate the application of the methodology to a relevant problem.

The in-silico trial (IST) into flow diversion (FD) of posterior communicating artery (PComA) aneurysms presented in Chapter 4 achieved the first aim in part. The FD-PComA IST successfully generated evidence for flow diverter efficacy in PComA aneurysms with and without fetal posterior circulation (FPC). Flow diverter treatment of PComA aneurysms using the Pipeline Embolization Device is not approved by the U.S. Food and Drug Administration and treatment of PComA aneurysms with FPC has not been investigated in clinical trials to date, so FD-PComA demonstrates the use of ISTs to generate evidence in off-label scenarios that have not been widely studied. FPC was found to be more prevalent in particular demographics, so FD-PComA also demonstrates how ISTs can be used to generate evidence in demographic groups that may be under-represented in clinical trials. For these reasons, FD-PComA is an exemplar trial that demonstrates the usefulness of ISTs. In the FD-PComA trial, device deployment, boundary condition assignment, simulation set up and execution, and post-processing were fully automated. Surface mesh pre-processing required some manual corrections in cases that had small thin vessels which were not detected in the centreline calculation. Device clipping also required manual input in some cases to ensure the clipping radius was not over/undersized. To create a fully automatic simulation pipeline would require additional work into improving these steps.

The literature review presented in Chapter 3 achieved the second thesis aim in full. The review covered reduced order modelling and machine learning techniques for simulation acceleration and found that each method has strengths and limitations depending upon the specific application. A taxonomy of each reviewed approach's suitability to the complexities of vascular flow modelling can be found in Table 3.4.1. Various avenues for future research emerged from this review, a number of which are outlined in Section 6.5.

The machine learning reduced order model (ML-ROM) study presented in Chapter 5 achieved the third aim in full. This chapter made a methodological contribution to ML-

ROM design by showing that a two-stage reduction of the training data using successive applications of POD improves the ROM performance. The ML-ROM was then applied to evaluating flow fields in intracranial aneurysms subject to physiological variation, where the model demonstrated high levels of accuracy and acceleration compared to a computational fluid dynamics model.

6.5 Future Work

This thesis has made significant contributions in a number of areas related to computational modelling and simulation for in-silico trials applications. Various studies could be undertaken as a continuation of this work.

From the literature review into accelerated simulation methodologies (Chapter 3), possible future studies include:

- The development of accelerated simulation methods that can evaluate flow fields for geometries that were not included in the training data. Techniques such as Point-Nets could be explored for capturing geometric variability [108, 109]. Informing the Point-Net with physics could reduce the amount of training data required and improve the accuracy of the inferred solution fields [268].

- If the development of the aforementioned methods are successful, then using these methods and traditional numerical methods to perform an in-silico trial would allow for comparison between the two approaches in terms of accuracy and efficiency.

- The development of a benchmarking framework for reduced order modelling and machine learning simulation acceleration techniques. This framework would include carefully selected exemplar problems to align with the common complexities in computational modelling (see Section 2.4), data for these problems, and specified metrics for assessing the accuracy and acceleration of the simulation acceleration techniques.

From the FD-PComA in-silico trial (Chapter 4), future studies could include:

- An in-silico trial for PComA aneurysm treatment comparing single PED flow diversion with other treatments, such as PED flow diversion with multiple PEDs, PED flow diversion with adjunct coiling, and intrasaccular devices such as the Woven Endovascular Bridge [324] or Contour Neurovascular System [325].

- An in-silico twin trial for PED treatment in PComA aneurysms. This would require collecting clinical data for patients with PComA aneurysms that were treated by flow diversion, constructing models of the patients and comparing the simulation results with the clinical outcomes.
- Analysis of the simulation pipeline using the ASME V&V 40 framework for establishing model credibility based on the question of interest and context of use [326].

From the ROM study (Chapter 5), future studies could include:

- Application of the ROM developed in Chapter 5 to real-time flow monitoring in cardiovascular applications with the ROM input parameters derived from easily measured quantities such as blood pressure and heart rate.
- Investigation of machine learning-based dimensionality reduction techniques, such as autoencoders, for interpolation-based ROMs [223].

APPENDIX A

Chapter 4: Additional Information and Results

A.1 Fetal Posterior Circulation

Table A.1.1: Fetal posterior circulation prevalence in various studies undertaken in different countries. Fetal/fetal-type are taken as the pooled hypoplastic P1 PCA types.

Study details						
Reference	[327]	[328]	[329]	[330]	[331]	[332]
Location	USA (1)	Morocco	USA (2)	France	Iran	Sri Lanka
Total number of patients	994	100	414	200	102	225
Circle of Willis variation	Number of patients					
Typical (full circle)	192	18	20	29	29	32
UL HP P1	47	3	4	5	1	2
BL HP P1s	33	1	16	6	0	1
HP P1 and CL A1	2	0	10	0	0	0
HP P1 and IL A1	20	1	1	3	1	4
BL HP P1s and A1	5	0	3	1	0	1
HP AComA and P1	35	4	0	4	1	6
HP PComA and CL P1	26	0	46	3	2	1
HP PComA, AComA and CL P1	17	1	28	1	1	0
HP P1, CL PComA and IL A1	10	1	5	0	0	0
BL HP P1s and AComA	13	0	10	1	0	1
Fetal/non-fetal	Prevalence, % (n/N)					
Fetal/fetal-type (HP P1 types pooled)	20.9	11.0	29.7	12.0	5.9	7.1
Non-fetal (remaining cases)	79.1	89.0	70.3	88.0	94.1	92.9

A1, A1 anterior cerebral artery; AComA, anterior communicating artery; BL, bilateral; CL, contralateral; IL, ipsilateral; HP, hypoplastic; P1, P1 posterior cerebral artery; PComA, posterior communicating artery.

De Silva et al. [332] compared different Circle of Willis variations in their study and a series of similar studies. Hypoplasia of the P1 posterior cerebral artery (PCA), which is a feature of fetal posterior circulation (FPC), was included in a number of these variations. Hypolastic P1 PCA can be used as an indicator of the patient having FPC and the patients for each study can be pooled into those with and without P1 PCA hypoplasia to find the prevalence of fetal and non-fetal vasculature in the different studies (see Table A.1.1). The findings indicate that FPC occurs in roughly 25.3% of people in America (USA (1) and USA (2)), 11.5% in Europe and North Africa (Morocco and France) and 6.5% in Asia (Iran and Sri Lanka). This highlights the variance in FPC prevalence across different demographics (ethnicities/races).

A.2 Methodology

A.2.1 Volumetric Meshing

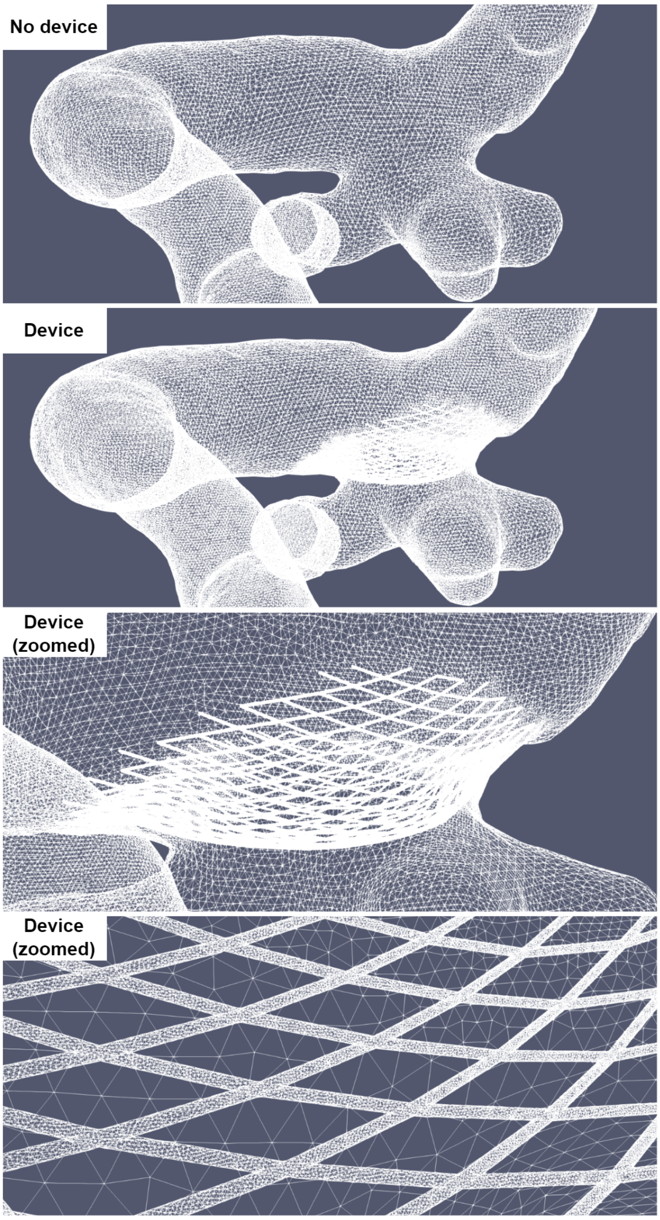


Figure A.2.1: Pre- and post-treatment surface meshes for one case used in the FD-PComA in-silico trial. The volumetric mesh is similarly more highly refined in the vicinity of the clipped flow diverter than in the vessel.

The volumetric meshes in the FD-PComA in-silico trial were more highly refined in the vicinity of the device. Figure A.2.1 shows the surface meshes for one case with and without the clipped flow diverter. The highly refined surface mesh for the device illustrates the high density of mesh elements required to explicitly model the device struts.

A.3 Results

A.3.1 Flow Variables vs. Aneurysm Characteristics

The three flow variables of interest (aneurysm STAV reduction, aneurysm neck MTAV reduction, stent STAWSS) were plotted against four aneurysm morphological parameters (maximum diameter, neck diameter, aspect ratio, non-sphericity index). A line of best fit was found between the morphological parameters and the pooled non-fetal and fetal flow variables. The R^2 and p values associated to the best fit line were calculated for each plot. Figure A.3.1 shows aneurysm STAV reduction against the morphological parameters. Figure A.3.2 shows aneurysm neck MTAV against aneurysm morphology. Figure A.3.3 shows stent STAWSS against aneurysm morphology. In each figure, the R^2 values are typically small and the p values are large. The lowest p value is found for aneurysm STAV reduction against aneurysm neck width ($p = 0.028$). These results suggest that aneurysm morphology typically does not play an important role in the assessment of treatment success using haemodynamic metrics.

Aneurysm STAV Reduction

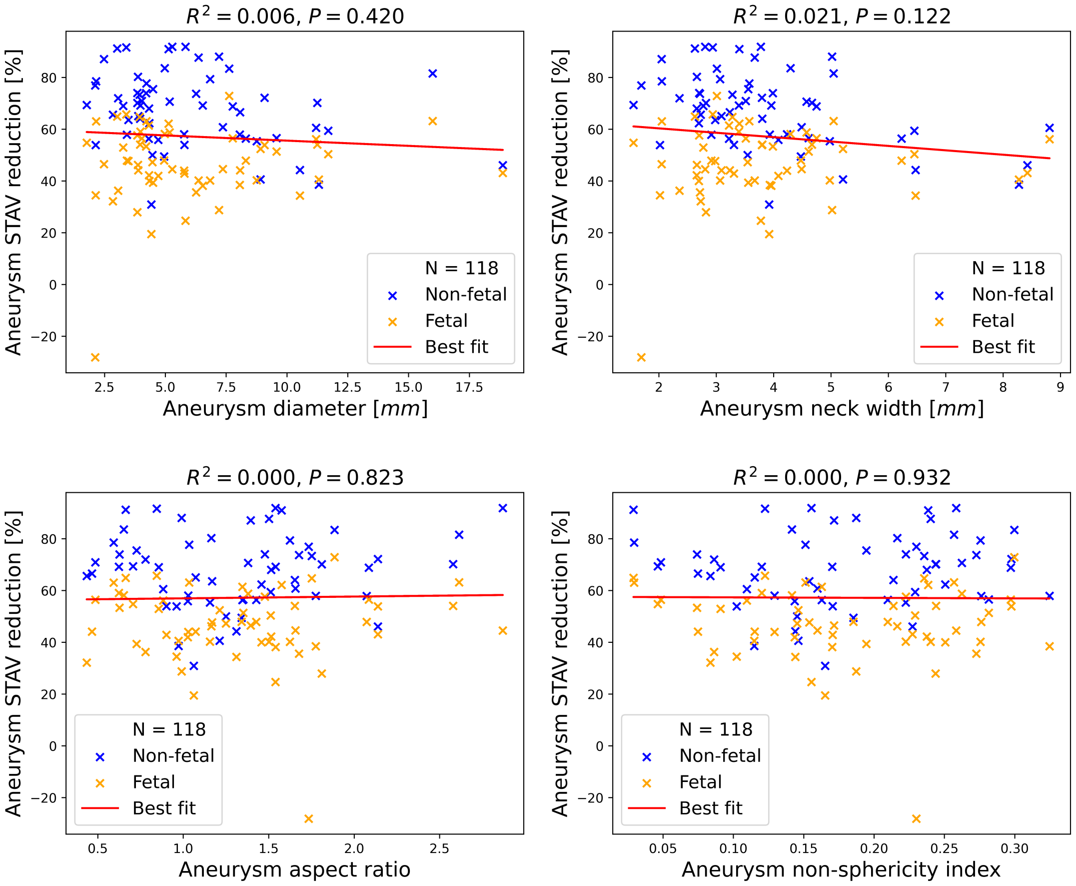


Figure A.3.1: Aneurysm STAV reduction vs. aneurysm morphological characteristics.

Aneurysm Neck MTAV Reduction

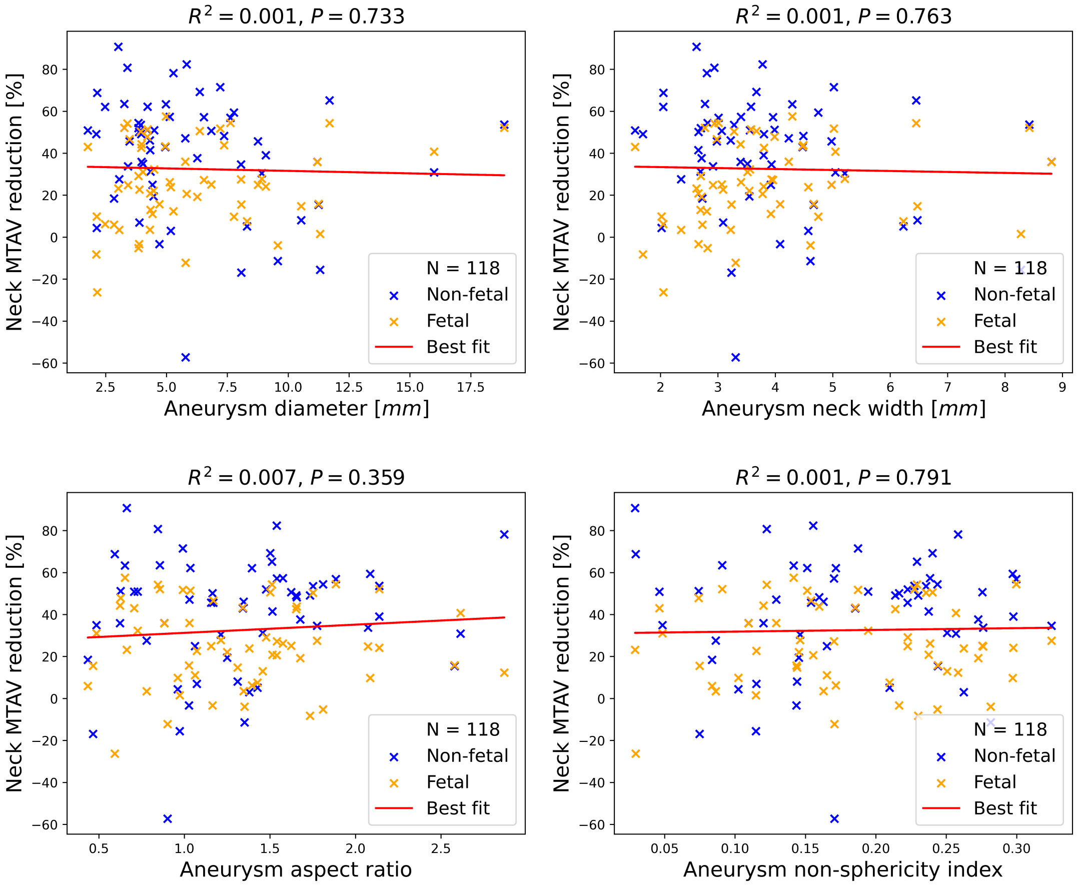


Figure A.3.2: Aneurysm neck MTAV reduction vs. aneurysm morphological characteristics.

Stent STAWSS vs. Aneurysm Characteristics

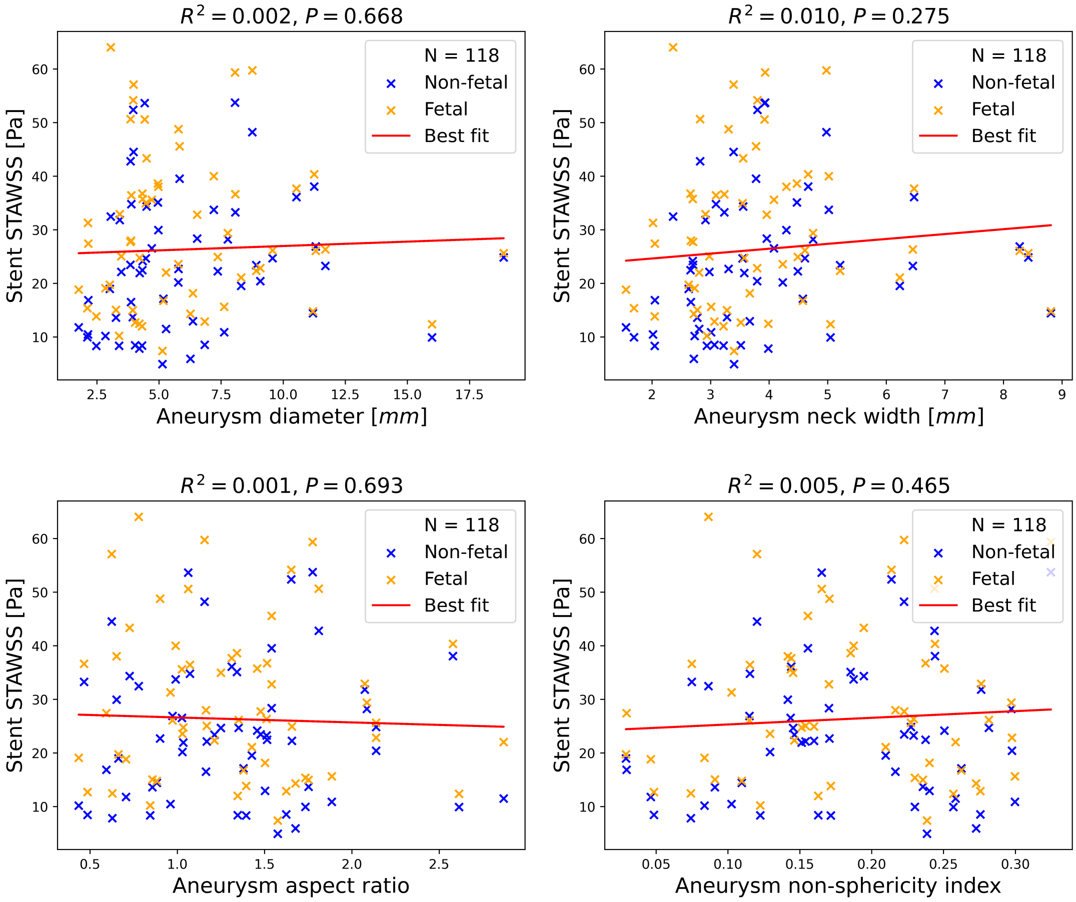


Figure A.3.3: Stent STAWSS vs. aneurysm morphological characteristics.

A.3.2 Flow Variables vs. PComA Size

The three flow variables of interest (aneurysm STAV reduction, aneurysm neck MTAV reduction, stent STAWSS) were plotted against the size of the PComA vessel. PComA size was characterised by its radius and by the relative radius between the PComA and ICA. A line of best fit was found between the morphological parameters and the pooled non-fetal and fetal flow variables. The R^2 and p values associated to the best fit line were calculated for each plot. The plots are shown in Figure A.3.4. The R^2 values are small and the p values are large, demonstrating that there is not a significant relationship between the three flow variables or the two measures of PComA size.

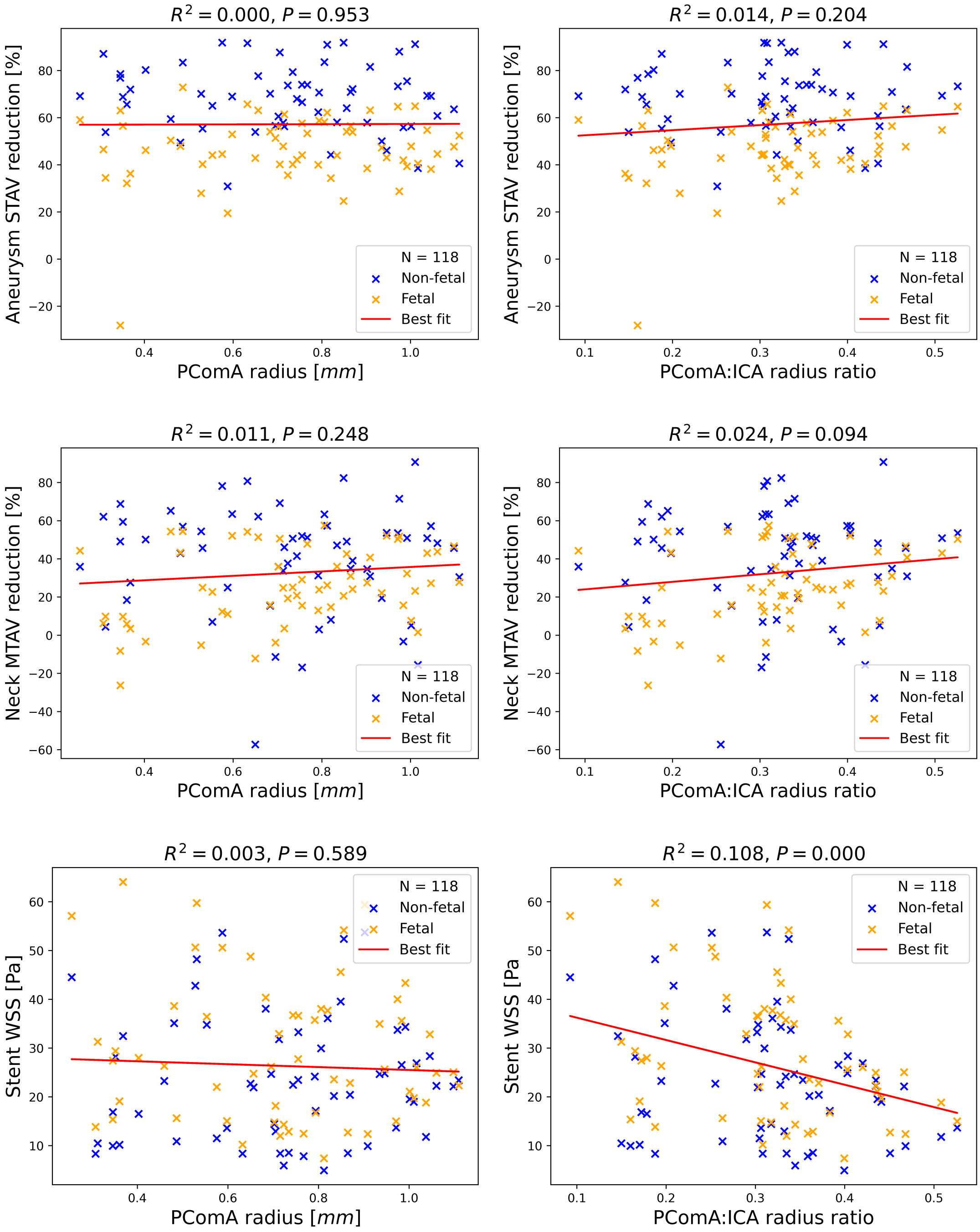


Figure A.3.4: Flow variables vs. PComA size.

APPENDIX B

Chapter 5: Hyperparameter Studies

B.1 Normalisation and Activation Function Tests

Three strategies for normalisation and standardisation of the ROM coefficients were investigated: (i) global normalisation and standardisation, (ii) local normalisation and standardisation and (iii) no normalisation and standardisation. These strategies were tested for the T network only. The normalisation function was defined by:

$$X_{norm} = 2\left(\frac{X - X_{min}}{X_{max} - X_{min}}\right) - 1, \tag{B.1.1}$$

where X is the ROM coefficients. For the global normalisation, the minimum and maximum values are taken across the entire set of coefficients for all simulations, timesteps and POD modes. For the local normalisation, the above equation is applied to the coefficients for each POD mode separately, with the minimum and maximum values taken across all coefficients for the given mode. Standardisation is applied by using:

$$X_{stan} = \frac{(X - \mu_X)}{\sigma_X}, \tag{B.1.2}$$

where μ and σ are the mean and standard deviation, respectively. Various activation functions were also tested, namely: (i) ReLU on all layers, (ii) Tanh on all layers, (iii) Leaky ReLU on all layers, and (iv) LeakyReLU on hidden layers and Tanh on final layer.

These various data processing and activation function strategies were tested using a T network test network consisting of two hidden layers with ten and 100 neurons in each layer. The first 68 POD modes were retained in the model, corresponding to 99.999% of the energy in the system. All networks were trained until either 100 epochs or until the loss plateaued. A network was trained for every combination of the above data processing and activation function choices and the mean test error was calculated across eight test cases.

From Table B.1.1, it is evident that global normalisation and standardisation significantly outperforms the other choices. The best-performing activation function is Tanh, but using LeakyReLU and Tanh in combination only performed marginally worse. Given this, additional preliminary tests were performed using both activation function combinations in order to determine which is best. In this instance, each activation function choice was tested with a series of network sizes and number of modes retained in the model. Results for this are presented in Tables B.1.2 and B.1.3. For the vessel model, Table B.1.2 shows that the Tanh activation function achieves a minimum error

Table B.1.1: Mean test error for T networks trained with different normalisation and standardisation schemes (global, local, none) and different activation function choices (ReLU, Tanh, LeakyReLU, LeakyReLU plus Tanh final layer). The test cases is for the aneurysm geometry. The model uses a network with two hidden layers (ten and 100 neurons in each layer) and 68 POD modes retained.

	ReLU	Tanh	Leaky ReLU	Leaky ReLU and Tanh
Global	32.3%	22.5%	30.4%	25.1%
Local	102.7%	104%	102.8%	104.2%
None	99.9%	97.6%	99.3%	97.0%

of 4.78% vs. 3.71% for Tanh and LeakyReLU. For the aneurysm model, Table B.1.3 shows that the minimum error for Tanh is 4.04% vs. 4.11% for Tanh and Leaky ReLU. It also appears that the minimum error with Tanh activation grows to larger values than for Tanh and Leaky ReLU. Overall, the two activation functions produce similar results, but Tanh and Leaky ReLU seems to perform slightly better in this scenario. For this reason, Leaky ReLU was chosen for hidden layers and Tanh for the output layer in subsequent testing.

Table B.1.2: Preliminary hyperparameter study results for vessel model. Mean and minimum test errors are taken across all hyperparameter combinations.

Hyperparameter	Value	Mean Error [%]	Minimum Error [%]
T Network with Tanh (Vessel)			
Layers	[10]	22.1	5.58
	[10, 100]	9.65	4.78
	[10, 100, 1000]	11.9	5.83
	[1000, 1000, 1000]	12.1	6.32
	[10, 10, 10, 10, 10, 10]	22.9	8.68
Number of POD modes	1	11.1	10.7
	2	6.05	4.78
	6	8.59	4.55
	12	14.8	10.6
	24	32.9	15.7
T Network with Leaky ReLU + Tanh (Vessel)			
Layers	[10]	5.69	3.71
	[10, 100]	7.42	4.27
	[10, 100, 1000]	7.72	5.27
	[1000, 1000, 1000]	8.60	3.86
	[10, 10, 10, 10, 10, 10]	13.0	8.16
Number of POD modes	1	10.8	10.7
	2	5.37	4.17
	6	6.31	3.71
	12	7.83	4.31
	24	11.4	5.52

Table B.1.3: Preliminary hyperparameter study results for aneurysm model. Mean and minimum test errors are taken across all hyperparameter combinations.

Hyperparameter	Value	Mean Error [%]	Minimum Error [%]
T Network with Tanh (Aneurysm)			
Layers	[10]	47.2	4.04
	[10, 100]	16.9	6.09
	[10, 100, 1000]	26.0	6.52
	[1000, 1000, 1000]	24.6	5.54
	[10, 10, 10, 10, 10, 10]	45.4	14.1
Number of POD modes	1	13.7	13.5
	6	6.95	4.04
	24	15.7	14.4
	68	24.4	17.9
	143	87.9	28.2
T Network with Leaky ReLU + Tanh (Aneurysm)			
Layers	[10]	9.00	4.11
	[10, 100]	24.5	5.25
	[10, 100, 1000]	14.1	5.65
	[1000, 1000, 1000]	15.8	6.28
	[10, 10, 10, 10, 10, 10]	16.8	13.5
Number of POD modes	1	13.5	13.4
	6	7.23	4.11
	24	13.3	5.22
	68	19.6	8.88
	143	26.5	13.4

B.2 Hyperparameter Study 1

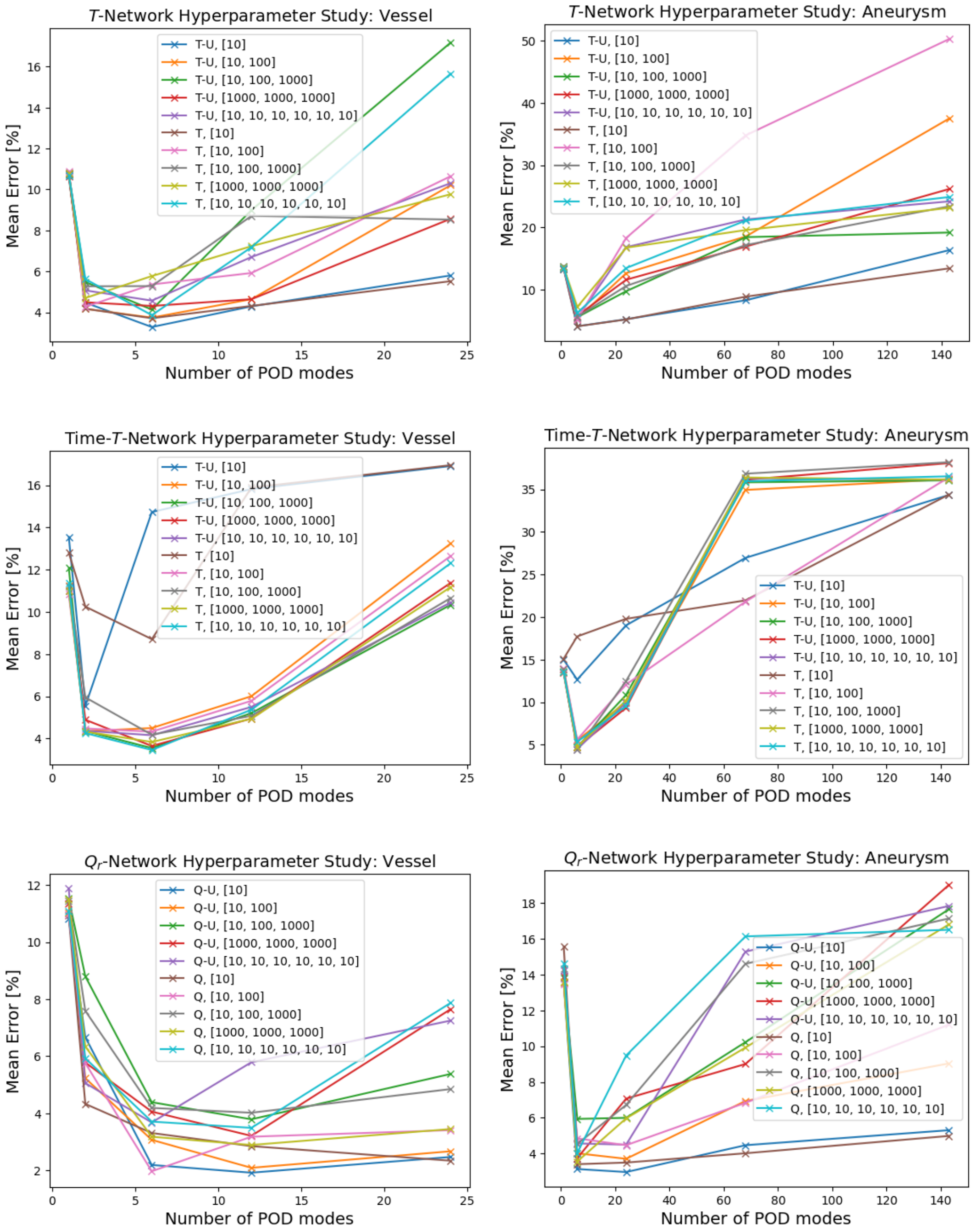


Figure B.2.1: Mean test error against number of POD modes for various network sizes and loss formulations.

Table B.2.1: First hyperparameter study results for vessel model. Mean and minimum test errors are taken across all hyperparameter combinations.

Hyperparameter	Value	Mean Error [%]	Minimum Error [%]
T Network			
Loss Construction	T-U	7.99	3.28
	T	8.34	3.71
Layers	[10]	5.70	3.28
	[10, 100]	7.06	3.74
	[10, 100, 1000]	8.51	4.13
	[1000, 1000, 1000]	8.05	3.86
	[10, 10, 10, 10, 10, 10]	12.6	8.16
Number of POD modes	1	10.8	10.6
	2	5.34	4.17
	6	5.96	3.28
	12	7.56	4.29
	24	11.2	5.52
Time-T Network			
Loss Construction	T-U	11.1	3.55
	T	10.0	3.46
Layers	[10]	13.1	5.56
	[10, 100]	7.74	4.33
	[10, 100, 1000]	7.26	3.55
	[1000, 1000, 1000]	7.22	3.46
	[10, 10, 10, 10, 10, 10]	21.0	4.32
Number of POD modes	1	11.5	10.8
	2	5.15	4.26
	6	7.90	3.46
	12	12.0	4.94
	24	16.3	10.3
Q_r Network			
Loss Construction	Q-U	7.07	1.92
	Q	6.87	1.97
Layers	[10]	4.78	1.92
	[10, 100]	4.94	1.97
	[10, 100, 1000]	6.60	3.79
	[1000, 1000, 1000]	6.58	3.49
	[10, 10, 10, 10, 10, 10]	13.0	7.63
Number of POD modes	1	12.2	10.8
	2	6.45	4.33
	6	5.05	1.97
	12	5.01	1.92
	24	6.20	2.34

Table B.2.2: First hyperparameter study results for aneurysm model. Mean and minimum test errors are taken across all hyperparameter combinations.

Hyperparameter	Value	Mean Error [%]	Minimum Error [%]
T Network			
Loss Construction	T-U	14.7	4.12
	T	16.0	4.11
Layers	[10]	9.24	4.11
	[10, 100]	21.0	5.25
	[10, 100, 1000]	13.7	5.58
	[1000, 1000, 1000]	16.0	5.43
	[10, 10, 10, 10, 10, 10]	16.9	13.5
Number of POD modes	1	13.6	13.4
	6	7.0	4.11
	24	12.6	5.22
	68	18.3	8.30
	143	25.4	13.4
Time-T Network			
Loss Construction	T-U	22.4	4.49
	T	21.2	4.45
Layers	[10]	21.7	12.7
	[10, 100]	19.0	5.51
	[10, 100, 1000]	20.7	4.45
	[1000, 1000, 1000]	20.1	4.49
	[10, 10, 10, 10, 10, 10]	28.8	7.11
Number of POD modes	1	14.0	13.5
	6	9.41	4.45
	24	16.3	9.34
	68	33.0	21.8
	143	36.2	34.3
Q_r Network			
Loss Construction	Q-U	10.1	2.97
	Q	10.4	3.41
Layers	[10]	6.09	2.97
	[10, 100]	7.81	3.71
	[10, 100, 1000]	11.1	4.54
	[1000, 1000, 1000]	11.7	4.04
	[10, 10, 10, 10, 10, 10]	14.5	14.0
Number of POD modes	1	14.2	13.5
	6	5.84	3.14
	24	6.90	2.97
	68	10.6	4.02
	143	13.7	4.99

B.3 Hyperparameter Study 2

Table B.3.1: Second hyperparameter study results for vessel model. Mean and minimum test errors are taken across all hyperparameter combinations. The lowest minimum error (i.e. the best-performing model) is in bold.

Hyperparameter	Value	Mean Error [%]	Minimum Error [%]
T Network			
Layers	[10]	3.82	3.08
	[10, 100]	4.23	**2.94**
Number of POD modes	2	4.32	4.17
	4	3.01	**2.94**
	6	3.51	3.28
	8	4.47	3.77
	10	4.37	4.02
	12	4.47	4.29
Time-T Network			
Layers	[10, 100, 1000]	4.64	3.27
	[1000, 1000, 1000]	4.49	**3.20**
	[1000]	8.94	5.15
Number of POD modes	2	5.14	4.26
	4	3.87	**3.20**
	6	5.79	3.46
	8	6.53	4.35
	10	7.33	4.72
	12	7.46	5.08
Q_r Network			
Layers	[10]	3.48	**1.30**
	[10, 100]	4.09	1.77
Number of POD modes	2	5.75	3.97
	4	3.53	1.86
	6	3.27	1.65
	8	3.13	1.39
	10	3.52	1.54
	12	3.39	**1.30**
	16	3.91	2.15
Number of second POD modes	1	7.30	6.89
	2	3.19	1.75
	3	2.89	1.39
	4	2.77	1.54
	5	3.35	**1.30**
	7	3.28	1.95
	10	3.72	1.86

Table B.3.2: Second hyperparameter study results for aneurysm model. Mean and minimum test errors are taken across all hyperparameter combinations. The lowest minimum error (i.e. the best-performing model) is in bold.

Hyperparameter	Value	Mean Error [%]	Minimum Error [%]
T Network			
Layers	[10]	4.12	**3.47**
	[10, 100]	7.42	4.73
Number of POD modes	4	4.37	4.01
	6	4.74	4.12
	8	4.48	3.58
	10	5.29	**3.47**
	12	4.69	3.83
	18	7.84	4.48
	24	8.96	5.25
Time-T Network			
Layers	[10, 100, 1000]	6.81	**4.45**
	[1000, 1000, 1000]	6.56	5.28
	[1000]	11.6	7.37
Number of POD modes	4	5.94	5.18
	6	6.02	**4.45**
	8	7.04	5.24
	12	8.41	5.86
	18	9.83	7.52
	24	12.68	9.56
Q_r Network			
Layers	[10]	4.43	**2.66**
	[10, 100]	5.93	3.28
Number of POD modes	1	4.66	2.88
	16	4.59	**2.66**
	20	5.02	2.99
	24	5.15	2.97
	28	5.98	3.16
	32	5.70	3.38
Number of second POD modes	1	8.37	8.17
	2	4.06	3.43
	3	3.64	**2.66**
	4	4.30	3.28
	5	4.29	2.99
	7	5.57	2.88
	10	6.05	3.49

Bibliography

[1] Ali Sarrami Foroushani. *In-silico clinical trials for assessment of intracranial flow diverters*. PhD thesis, University of Sheffield, 2018.

[2] Lorenzo Rinaldo, Waleed Brinjikji, Harry Cloft, Giuseppe Lanzino, L Fernando Gonzalez, Peter Kan, and Leonardo Rangel Castilla. Effect of fetal posterior circulation on efficacy of flow diversion for treatment of posterior communicating artery aneurysms: a multi-institutional study. *World neurosurgery*, 127:e1232–e1236, 2019.

[3] JPKH Alastruey, KH Parker, J Peiró, SM Byrd, and SJ Sherwin. Modelling the circle of willis to assess the effects of anatomical variations and occlusions on cerebral flows. *Journal of biomechanics*, 40(8):1794–1805, 2007.

[4] Laleh Zarrinkoob, Khalid Ambarki, Anders Wåhlin, Richard Birgander, Anders Eklund, and Jan Malm. Blood flow distribution in cerebral arteries. *Journal of Cerebral Blood Flow & Metabolism*, 35(4):648–654, 2015.

[5] AF Frangi, T Denison, P Myles, J Ordish, P Brown, R Turpin, M Kipping, M Palmer, D Flynn, P Afshari, C Lane, M de Cunha, M Horner, S Levine, T Marchal, R Bryan, G Tunbridge, J Pink, S Macpherson, S Niederer, R Shipley, E Dall'Ara, T Maeder, and M Thompson. *Unlocking the power of computational modelling and simulation across the product lifecycle in life sciences: A UK Landscape Report*. InSilicoUK Pro-Innovation Regulations Network, September 2023. doi: 10.5281/zenodo.7723230. URL `https://doi.org/10.5281/zenodo.7723230`. InnovateUK KTN and the Royal Academy of Engineering Chair in Emerging Technology (INSILEX CiET1819/19) funded this report. We also acknow-

ledge contributions from many members of the InSilicoUK Innovation Network (www.insilicouk.org).

[6] Linda Martin, Melissa Hutchens, Conrad Hawkins, and Alaina Radnov. How much do clinical trials cost. *Nat Rev Drug Discov*, 16(6):381–382, 2017.

[7] Chi Heem Wong, Kien Wei Siah, and Andrew W Lo. Estimation of clinical trial success rates and related parameters. *Biostatistics*, 20(2):273–286, 2019.

[8] Josh Makower, Aabed Meer, and Lyn Denend. Fda impact on us medical technology innovation: a survey of over 200 medical technology companies. *Arlington (Virginia): National Venture Capital Association*, 2010.

[9] Jeanne Lenzer et al. Medical device industry: international investigation exposes lax regulation, 2018.

[10] Sustainable Trials Study Group et al. Towards sustainable clinical trials. *Bmj*, 334(7595):671–673, 2007.

[11] Marco Viceconti, Adriano Henney, and Edwin Morley-Fletcher. In silico clinical trials: how computer simulation will transform the biomedical industry. *International Journal of Clinical Trials*, 3(2):37–46, 2016.

[12] Marco Viceconti, Miguel A Juárez, Cristina Curreli, Marzio Pennisi, Giulia Russo, and Francesco Pappalardo. Credibility of in silico trial technologiesâ€”a theoretical framing. *IEEE Journal of Biomedical and Health Informatics*, 24(1):4–13, 2019.

[13] M. Viceconti, A. Henney, and E. Morley-Fletcher. In silico clinical trials: how computer simulation will transform the biomedical industry. *Int. J. Clinical Trials*, 3(2):37–46, 2016.

[14] Marco De Vivo, Matteo Masetti, Giovanni Bottegoni, and Andrea Cavalli. Role of molecular dynamics and related methods in drug discovery. *Journal of medicinal chemistry*, 59(9):4035–4061, 2016.

[15] Xiaomei Zhuang and Chuang Lu. Pbpk modeling and simulation in drug research and development. *Acta Pharmaceutica Sinica B*, 6(5):430–440, 2016.

[16] Lalo Magni, Davide Martino Raimondo, Chiara Dalla Man, Giuseppe De Nicolao, B Kovatchev, and Claudio Cobelli. Model predictive control of glucose concentration in type i diabetic patients: An in silico trial. *Biomedical Signal Processing and Control*, 4(4):338–346, 2009.

[17] Praneeta R Konduri, Henk A Marquering, Ed E Van Bavel, Alfons Hoekstra, Charles BLM Majoie, Insist Investigators, et al. In-silico trials for treatment of acute ischemic stroke. *Frontiers in Neurology*, 11:1062, 2020.

[18] Claire Miller, Raymond M Padmos, Max van der Kolk, Tamás I Józsa, Noor Samuels, Yidan Xue, Stephen J Payne, and Alfons G Hoekstra. In silico trials for treatment of acute ischemic stroke: Design and implementation. *Computers in biology and medicine*, 137:104802, 2021.

[19] Peter Ifeoluwa Adegbola, Olumide Samuel Fadahunsi, Aanuoluwa Eunice Adegbola, and Banjo Semire. In silico studies of potency and safety assessment of selected trial drugs for the treatment of covid-19. *In silico pharmacology*, 9(1): 1–12, 2021.

[20] Ali Sarrami-Foroushani, Toni Lassila, Michael MacRaild, Joshua Asquith, Kit CB Roes, James V Byrne, and Alejandro F Frangi. In-silico trial of intracranial flow diverters replicates and expands insights from conventional clinical trials. *Nature communications*, 12(1):1–12, 2021.

[21] Ali Sarrami-Foroushani, Toni Lassila, Ali Gooya, Arjan J Geers, and Alejandro F Frangi. Uncertainty quantification of wall shear stress in intracranial aneurysms using a data-driven statistical model of systemic blood flow variability. *Journal of biomechanics*, 49(16):3815–3823, 2016.

[22] Ali Sarrami-Foroushani, Toni Lassila, and Alejandro F Frangi. Virtual endovascular treatment of intracranial aneurysms: models and uncertainty. *Wiley Interdisciplinary Reviews: Systems Biology and Medicine*, 9(4):e1385, 2017.

[23] Ali Sarrami-Foroushani, Toni Lassila, Seyed Mostafa Hejazi, Sanjoy Nagaraja, Andrew Bacon, and Alejandro F Frangi. A computational model for prediction of clot platelet content in flow-diverted intracranial aneurysms. *Journal of Biomechanics*, 91:7–13, 2019.

[24] Kathleen Withers, Grace Carolan-Rees, and Megan Dale. Pipelineâ„¢ embolization device for the treatment of complex intracranial aneurysms. *Applied health economics and health policy*, 11(1):5–13, 2013.

[25] Charles Vega, Jeremiah V Kwoon, and Sean D Lavine. Intracranial aneurysms: current evidence and clinical practice. *American family physician*, 66(4):601, 2002.

[26] Gabriel JE Rinkel, Mamuka Djibuti, Ale Algra, and J Van Gijn. Prevalence and risk of rupture of intracranial aneurysms: a systematic review. *Stroke*, 29(1): 251–256, 1998.

[27] International Study of Unruptured Intracranial Aneurysms Investigators D. Wiebers et. al. Unruptured intracranial aneurysmsâ€"risk of rupture and risks of surgical intervention. *New England Journal of Medicine*, 339(24):1725–1733, 1998.

[28] H Meng, VM Tutino, J Xiang, and A Siddiqui. High wss or low wss? complex interactions of hemodynamics with intracranial aneurysm initiation, growth, and rupture: toward a unifying hypothesis. *American Journal of Neuroradiology*, 35 (7):1254–1262, 2014.

[29] E Sander Connolly Jr, Alejandro A Rabinstein, J Ricardo Carhuapoma, Colin P Derdeyn, Jacques Dion, Randall T Higashida, Brian L Hoh, Catherine J Kirkness, Andrew M Naidech, Christopher S Ogilvy, et al. Guidelines for the management of aneurysmal subarachnoid hemorrhage: a guideline for healthcare professionals from the american heart association/american stroke association. *Stroke*, 43(6): 1711–1737, 2012.

[30] Oliver Rivero-Arias, Alastair Gray, and Jane Wolstenholme. Burden of disease and costs of aneurysmal subarachnoid haemorrhage (asah) in the united kingdom. *Cost effectiveness and resource allocation*, 8:1–12, 2010.

[31] Alexander Keedy. An overview of intracranial aneurysms. *McGill Journal of Medicine: MJM*, 9(2):141, 2006.

[32] A Chien, M Xu, H Yokota, F Scalzo, E Morimoto, and Noriko Salamon. Nonsphericity index and size ratio identify morphologic differences between growing

and stable aneurysms in a longitudinal study of 93 cases. *American Journal of Neuroradiology*, 39(3):500–506, 2018.

[33] Rodney A Gabriel, Helen Kim, Stephen Sidney, Charles E McCulloch, Vineeta Singh, S Claiborne Johnston, Nerissa U Ko, Achal S Achrol, Jonathan G Zaroff, and William L Young. Ten-year detection rate of brain arteriovenous malformations in a large, multiethnic, defined population, 2010.

[34] Hiroshi Ujiie, Kazuei Sato, Hideaki Onda, Akihiro Oikawa, Mizuo Kagawa, Kintomo Takakura, and Naotoshi Kobayashi. Clinical analysis of incidentally discovered unruptured aneurysms. *Stroke*, 24(12):1850–1856, 1993.

[35] Juan R Cebral and Marcelo Raschi. Suggested connections between risk factors of intracranial aneurysms: a review. *Annals of biomedical engineering*, 41(7): 1366–1383, 2013.

[36] Nika V Polevaya, M Yashar S Kalani, Gary K Steinberg, and CK Victor. The transition from hunterian ligation to intracranial aneurysm clips: a historical perspective. *Neurosurgical focus*, 20(6):1–7, 2006.

[37] Yuichi Murayama, Yih Lin Nien, Gary Duckwiler, Y Pierre Gobin, Reza Jahan, John Frazee, Neil Martin, and Fernando Viñuela. Guglielmi detachable coil embolization of cerebral aneurysms: 11 years' experience. *Journal of neurosurgery*, 98(5):959–966, 2003.

[38] Y Kawanabe, A Sadato, W Taki, and N Hashimoto. Endovascular occlusion of intracranial aneurysms with guglielmi detachable coils: correlation between coil packing density and coil compaction. *Acta neurochirurgica*, 143(5), 2001.

[39] Eva H Brilstra, Gabriel JE Rinkel, Yolanda van der Graaf, Willem Jan J van Rooij, and Ale Algra. Treatment of intracranial aneurysms by embolization with coils: a systematic review. *Stroke*, 30(2):470–476, 1999.

[40] A Fernandez Zubillaga, Guido Guglielmi, Fernando Vinuela, and Gary R Duckwiler. Endovascular occlusion of intracranial aneurysms with electrically detachable coils: correlation of aneurysm neck size and treatment results. *American Journal of Neuroradiology*, 15(5):815–820, 1994.

[41] Gary Rajah, Sandra Narayanan, and Leonardo Rangel-Castilla. Update on flow diverters for the endovascular management of cerebral aneurysms. *Neurosurgical focus*, 42(6):E2, 2017.

[42] Timothy Phillips, Peter Mitchell, Richard Dowling, Bernard Yan, et al. Endovascular treatment of intracranial aneurysms with new generation flow diverting stents. early experience in an australian centre. 2010.

[43] James V Byrne and István Szikora. Flow diverters in the management of intracranial aneurysms: a review. *EJMINT Original Article*, 1225000057:29, 2012.

[44] Waleed Brinjikji, Mohammad H Murad, Giuseppe Lanzino, Harry J Cloft, and David F Kallmes. Endovascular treatment of intracranial aneurysms with flow diverters: a meta-analysis. *Stroke*, 44(2):442–447, 2013.

[45] Kyle M Fargen and Brian L Hoh. Flow diversion technologies in evolution: a review of the first two generations of flow diversion devices. *World neurosurgery*, 84(2):254–256, 2015.

[46] Ignacio Arrese, Rosario Sarabia, Rebeca Pintado, and Miguel Delgado-Rodriguez. Flow-diverter devices for intracranial aneurysms: systematic review and meta-analysis. *Neurosurgery*, 73(2):193–200, 2013.

[47] Brent King, Sasha Vaziri, Amit Singla, Kyle M Fargen, and J Mocco. Clinical and angiographic outcomes after stent-assisted coiling of cerebral aneurysms with enterprise and neuroform stents: a comparative analysis of the literature. *Journal of neurointerventional surgery*, 7(12):905–909, 2015.

[48] Nohra el Chalouhi, Pascal M Jabbour, Stavropoula I Tjoumakaris, Robert M Starke, Aaron S Dumont, Haisong Liu, Robert Rosenwasser, Sedeek El Moursi, and L Fernando Gonzalez. Treatment of large and giant intracranial aneurysms: cost comparison of flow diversion and traditional embolization strategies. *World neurosurgery*, 82(5):696–701, 2014.

[49] US Food and Drug Administration. General issues: Meeting to discuss the evaluation of safety and effectiveness of endovascular medical devices intended to treat intracranial aneurysms. Technical report, US Food and Drug Administration, 2011.

[50] Purvee D Patel, Nohra Chalouhi, Elias Atallah, Stavropoula Tjoumakaris, David Hasan, Hekmat Zarzour, Robert Rosenwasser, and Pascal Jabbour. Off-label uses of the pipeline embolization device: a review of the literature. *Neurosurgical focus*, 42(6):E4, 2017.

[51] J-m Liu, Y Zhou, Y Li, T Li, B Leng, P Zhang, G Liang, Q Huang, P-f Yang, H Shi, et al. Parent artery reconstruction for large or giant cerebral aneurysms using the tubridge flow diverter: a multicenter, randomized, controlled clinical trial (parat). *American Journal of Neuroradiology*, 39(5):807–816, 2018.

[52] Jean Raymond, Jean-Christophe Gentric, Tim E Darsaut, Daniela Iancu, Miguel Chagnon, Alain Weill, and Daniel Roy. Flow diversion in the treatment of aneurysms: a randomized care trial and registry. *Journal of neurosurgery*, 127(3): 454–462, 2016.

[53] Bongjae Chung and Juan Raul Cebral. Cfd for evaluation and treatment planning of aneurysms: review of proposed clinical uses and their challenges. *Annals of biomedical engineering*, 43(1):122–138, 2015.

[54] Sethuraman Sankaran, David Lesage, Rhea Tombropoulos, Nan Xiao, Hyun Jin Kim, David Spain, Michiel Schaap, and Charles A Taylor. Physics driven real-time blood flow simulations. *Computer Methods in Applied Mechanics and Engineering*, 364:112963, 2020.

[55] Timothy J Gundert, Alison L Marsden, Weiguang Yang, and John F LaDisa Jr. Optimization of cardiovascular stent design using computational fluid dynamics. 2012.

[56] Katharine H Fraser, M Ertan Taskin, Bartley P Griffith, and Zhongjun J Wu. The use of computational fluid dynamics in the development of ventricular assist devices. *Medical engineering & physics*, 33(3):263–280, 2011.

[57] Ming-Chen Hsu, David Kamensky, Fei Xu, Josef Kiendl, Chenglong Wang, Michael CH Wu, Joshua Mineroff, Alessandro Reali, Yuri Bazilevs, and Michael S Sacks. Dynamic and fluid–structure interaction simulations of bioprosthetic heart valves using parametric design with t-splines and fung-type material models. *Computational mechanics*, 55:1211–1225, 2015.

[58] David N Ku et al. Blood flow in arteries. *Annual review of fluid mechanics*, 29 (1):399–434, 1997.

[59] Alfio Quarteroni, Massimiliano Tuveri, and Alessandro Veneziani. Computational vascular fluid dynamics: problems, models and methods. *Computing and Visualization in Science*, 2(4):163–197, 2000.

[60] Lazaros Papamanolis, Hyun Jin Kim, Clara Jaquet, Matthew Sinclair, Michiel Schaap, Ibrahim Danad, Pepijn van Diemen, Paul Knaapen, Laurent Najman, Hugues Talbot, et al. Myocardial perfusion simulation for coronary artery disease: a coupled patient-specific multiscale model. *Annals of biomedical engineering*, 49: 1432–1447, 2021.

[61] Paolo Crosetto, Philippe Reymond, Simone Deparis, Dimitrios Kontaxakis, Nikolaos Stergiopulos, and Alfio Quarteroni. Fluid–structure interaction simulation of aortic blood flow. *Computers & Fluids*, 43(1):46–57, 2011.

[62] Seyed Saeid Khalafvand, Fei Xu, Jos Westenberg, Frank Gijsen, and Sasa Kenjeres. Intraventricular blood flow with a fully dynamic mitral valve model. *Computers in biology and medicine*, 104:197–204, 2019.

[63] Ivan Fumagalli, Rebecca Polidori, Francesca Renzi, Laura Fusini, Alfio Quarteroni, Gianluca Pontone, and Christian Vergara. Fluid-structure interaction analysis of transcatheter aortic valve implantation. *International Journal for Numerical Methods in Biomedical Engineering*, page e3704, 2023.

[64] Mirko Bonfanti, Stavroula Balabani, John P Greenwood, Sapna Puppala, Shervanthi Homer-Vanniasinkam, and Vanessa Díaz-Zuccarini. Computational tools for clinical support: a multi-scale compliant model for haemodynamic simulations in an aortic dissection based on multi-modal imaging data. *Journal of The Royal Society Interface*, 14(136):20170632, 2017.

[65] Soumen Jana. Endothelialization of cardiovascular devices. *Acta biomaterialia*, 99:53–71, 2019.

[66] Meongkeun Ju, Swe Soe Ye, Bumseok Namgung, Seungkwan Cho, Hong Tong Low, Hwa Liang Leo, and Sangho Kim. A review of numerical methods for red

blood cell flow simulation. *Computer methods in biomechanics and biomedical engineering*, 18(2):130–140, 2015.

[67] Amirhossein Arzani and Shawn C Shadden. Characterization of the transport topology in patient-specific abdominal aortic aneurysm models. *Physics of Fluids*, 24(8), 2012.

[68] Jurgen AHR Claassen, Dick HJ Thijssen, Ronney B Panerai, and Frank M Faraci. Regulation of cerebral blood flow in humans: physiology and clinical implications of autoregulation. *Physiological reviews*, 101(4):1487–1559, 2021.

[69] AV Alexandrov. Current and future recanalization strategies for acute ischemic stroke. *Journal of internal medicine*, 267(2):209–219, 2010.

[70] Elizabeth A Logsdon, Stacey D Finley, Aleksander S Popel, and Feilim Mac Gabhann. A systems biology view of blood vessel growth and remodelling. *Journal of cellular and molecular medicine*, 18(8):1491–1508, 2014.

[71] David M Wootton and David N Ku. Fluid mechanics of vascular systems, diseases, and thrombosis. *Annual review of biomedical engineering*, 1(1):299–329, 1999.

[72] Amirhossein Arzani. Accounting for residence-time in blood rheology models: do we really need non-newtonian blood flow modelling in large arteries? *Journal of The Royal Society Interface*, 15(146):20180486, 2018.

[73] Kartik Jain. Transition to turbulence in an oscillatory flow through stenosis. *Biomechanics and modeling in mechanobiology*, 19(1):113–131, 2020.

[74] Alireza Gholipour, Mergen H Ghayesh, and Anthony Zander. Nonlinear biomechanics of bifurcated atherosclerotic coronary arteries. *International Journal of Engineering Science*, 133:60–83, 2018.

[75] Ryo Torii, Marie Oshima, Toshio Kobayashi, Kiyoshi Takagi, and Tayfun E Tezduyar. Fluid–structure interaction modeling of blood flow and cerebral aneurysm: significance of artery and aneurysm shapes. *Computer Methods in Applied Mechanics and Engineering*, 198(45-46):3613–3621, 2009.

[76] Emily L Manchester, Selene Pirola, Mohammad Yousuf Salmasi, Declan P Oâ€™Regan, Thanos Athanasiou, and Xiao Yun Xu. Analysis of turbulence

effects in a patient-specific aorta with aortic valve stenosis. *Cardiovascular engineering and technology*, 12:438–453, 2021.

[77] Fernando Mut, Danny Ruijters, Drazenko Babic, Carlos Bleise, Pedro Lylyk, and Juan R Cebral. Effects of changing physiologic conditions on the in vivo quantification of hemodynamic variables in cerebral aneurysms treated with flow diverting devices. *International Journal for Numerical Methods in Biomedical Engineering*, 30(1):135–142, 2014.

[78] Martina Bukač, Sunčica Čanić, Josip Tambača, and Yifan Wang. Fluid–structure interaction between pulsatile blood flow and a curved stented coronary artery on a beating heart: A four stent computational study. *Computer Methods in Applied Mechanics and Engineering*, 350:679–700, 2019.

[79] Xiangkun Liu, Wen Zhang, Ping Ye, Qiyi Luo, and Zhaohua Chang. Fluid-structure interaction analysis on the influence of the aortic valve stent leaflet structure in hemodynamics. *Frontiers in Physiology*, page 910, 2022.

[80] Tomás Bodnár, Adélia Sequeira, and M Prosi. On the shear-thinning and viscoelastic effects of blood flow under various flow rates. *Applied Mathematics and Computation*, 217(11):5055–5067, 2011.

[81] Jianping Xiang, Sabareesh K Natarajan, Markus Tremmel, Ding Ma, J Mocco, L Nelson Hopkins, Adnan H Siddiqui, Elad I Levy, and Hui Meng. Hemodynamic–morphologic discriminants for intracranial aneurysm rupture. *Stroke*, 42(1):144–152, 2011.

[82] Loic Boussel, Vitaliy Rayz, Charles McCulloch, Alastair Martin, Gabriel Acevedo-Bolton, Michael Lawton, Randall Higashida, Wade S Smith, William L Young, and David Saloner. Aneurysm growth occurs at region of low wall shear stress: patient-specific correlation of hemodynamics and growth in a longitudinal study. *Stroke*, 39(11):2997–3002, 2008.

[83] Matthew D Ford, Hristo N Nikolov, Jaques S Milner, Stephen P Lownie, Edwin M DeMont, Wojciech Kalata, Francis Loth, David W Holdsworth, and David A Steinman. Piv-measured versus cfd-predicted flow dynamics in anatomically realistic cerebral aneurysm models. *Journal of biomechanical engineering*, 130(2), 2008.

[84] MA Castro, Christopher M Putman, and JR Cebral. Computational fluid dynamics modeling of intracranial aneurysms: effects of parent artery segmentation on intra-aneurysmal hemodynamics. *American Journal of Neuroradiology*, 27(8): 1703–1709, 2006.

[85] Juan R Cebral, Marcelo A Castro, James E Burgess, Richard S Pergolizzi, Michael J Sheridan, and Christopher M Putman. Characterization of cerebral aneurysms for assessing risk of rupture by using patient-specific computational hemodynamics models. *American Journal of Neuroradiology*, 26(10):2550–2559, 2005.

[86] Sujan Dhar, Markus Tremmel, J Mocco, Minsuok Kim, Junichi Yamamoto, Adnan H Siddiqui, L Nelson Hopkins, and Hui Meng. Morphology parameters for intracranial aneurysm rupture risk assessment. *Neurosurgery*, 63(2):185, 2008.

[87] Aristotelis P Mitsos, Nikolaos MP Kakalis, Yiannis P Ventikos, and James V Byrne. Haemodynamic simulation of aneurysm coiling in an anatomically accurate computational fluid dynamics model. *Neuroradiology*, 50:341–347, 2008.

[88] Hernán G Morales, Minsuok Kim, EE Vivas, M-C Villa-Uriol, Ignacio Larrabide, T Sola, Leopoldo Guimaraens, and AF Frangi. How do coil configuration and packing density influence intra-aneurysmal hemodynamics? *American Journal of Neuroradiology*, 32(10):1935–1941, 2011.

[89] Annarita Bernardini, Ignacio Larrabide, Hernán G Morales, Giancarlo Pennati, Lorenza Petrini, Salvatore Cito, and Alejandro F Frangi. Influence of different computational approaches for stent deployment on cerebral aneurysm haemodynamics. *Interface focus*, 1(3):338–348, 2011.

[90] Yasaman Aghli, Mojtaba Dayyani, Behzad Golparvar, Humain Baharvahdat, Raphael Blanc, Michel Piotin, and Hamid Niazmand. Image-based computational hemodynamic analysis of an anterior communicating aneurysm treated with the woven endobridge device. *Interdisciplinary Neurosurgery*, 25:101251, 2021.

[91] T. Lassila, A. Sarrami-Foroushani, S.M. Hejazi, and A.F. Frangi. Population-specific modelling of between/within-subject flow variability in the carotid arteries of the elderly. *Int. J. Numer. Method. Biomed. Eng.*, page e3271, 2019.

[92] Matthias Heil and Andrew L Hazel. Fluid-structure interaction in internal physiological flows. *Annual review of fluid mechanics*, 43:141–162, 2011.

[93] Leopold Grinberg, Alexander Yakhot, and George Em Karniadakis. Analyzing transient turbulence in a stenosed carotid artery by proper orthogonal decomposition. *Annals of Biomedical Engineering*, 37(11):2200–2217, 2009.

[94] Tufail Patankar, Jeremy Madigan, Jonathan Downer, Hemant Sonwalkar, Peter Cowley, and Francesco Iori. How precise is presize neurovascular? accuracy evaluation of flow diverter deployed-length prediction. *Journal of Neurosurgery*, 137(4):1072–1080, 2022.

[95] Samuel Voß, Philipp Berg, Gabor Janiga, Martin Skalej, and Oliver Beuing. Variability of intra-aneurysmal hemodynamics caused by stent-induced vessel deformation. *Current Directions in Biomedical Engineering*, 3(2):305–308, 2017.

[96] Peter Benner, Serkan Gugercin, and Karen Willcox. A survey of projection-based model reduction methods for parametric dynamical systems. *SIAM review*, 57 (4):483–531, 2015.

[97] Alfio Quarteroni, Gianluigi Rozza, et al. *Reduced order methods for modeling and computational reduction*, volume 9. Springer, 2014.

[98] Sivaguru S Ravindran. A reduced-order approach for optimal control of fluids using proper orthogonal decomposition. *International journal for numerical methods in fluids*, 34(5):425–448, 2000.

[99] Toni Lassila, Andrea Manzoni, Alfio Quarteroni, and Gianluigi Rozza. Model order reduction in fluid dynamics: challenges and perspectives. *Reduced Order Methods for modeling and computational reduction*, pages 235–273, 2014.

[100] Stefano Buoso, Andrea Manzoni, Hatem Alkadhi, André Plass, Alfio Quarteroni, and Vartan Kurtcuoglu. Reduced-order modeling of blood flow for noninvasive functional evaluation of coronary artery disease. *Biomechanics and Modeling in Mechanobiology*, pages 1–15, 2019.

[101] Matteo Salvador, Luca Dede, and Andrea Manzoni. Non intrusive reduced order modeling of parametrized pdes by kernel pod and neural networks. *Computers & Mathematics with Applications*, 104:1–13, 2021.

[102] Gilles Tissot, Laurent Cordier, Nicolas Benard, and Bernd R Noack. Model reduction using dynamic mode decomposition. *Comptes Rendus Mécanique*, 342 (6-7):410–416, 2014.

[103] Peter J Schmid. Dynamic mode decomposition and its variants. *Annual Review of Fluid Mechanics*, 54:225–254, 2022.

[104] Romit Maulik, Bethany Lusch, and Prasanna Balaprakash. Reduced-order modeling of advection-dominated systems with recurrent neural networks and convolutional autoencoders. *Physics of Fluids*, 33(3):037106, 2021.

[105] Pranshu Pant, Ruchit Doshi, Pranav Bahl, and Amir Barati Farimani. Deep learning for reduced order modelling and efficient temporal evolution of fluid simulations. *Physics of Fluids*, 33(10):107101, 2021.

[106] Maziar Raissi, Paris Perdikaris, and George E Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational physics*, 378:686–707, 2019.

[107] Lucian Itu, Saikiran Rapaka, Tiziano Passerini, Bogdan Georgescu, Chris Schwemmer, Max Schoebinger, Thomas Flohr, Puneet Sharma, and Dorin Comaniciu. A machine-learning approach for computation of fractional flow reserve from coronary computed tomography. *Journal of applied physiology*, 121 (1):42–52, 2016.

[108] Gaoyang Li, Haoran Wang, Mingzi Zhang, Simon Tupin, Aike Qiao, Youjun Liu, Makoto Ohta, and Hitomi Anzai. Prediction of 3d cardiovascular hemodynamics before and after coronary artery bypass surgery via deep learning. *Communications biology*, 4(1):99, 2021.

[109] Gaoyang Li, Xiaorui Song, Haoran Wang, Siwei Liu, Jiayuan Ji, Yuting Guo, Aike Qiao, Youjun Liu, and Xuezheng Wang. Prediction of cerebral aneurysm hemodynamics with porous-medium models of flow-diverting stents via deep learning. *Frontiers in Physiology*, 12:733444, 2021.

[110] Lu Lu, Pengzhan Jin, Guofei Pang, Zhongqiang Zhang, and George Em Karniadakis. Learning nonlinear operators via deeponet based on the universal ap-

proximation theorem of operators. *Nature Machine Intelligence*, 3(3):218–229, 2021.

[111] Somdatta Goswami, Aniruddha Bora, Yue Yu, and George Em Karniadakis. Physics-informed deep neural operators networks. *arXiv preprint arXiv:2207.05748*, 2022.

[112] Amirhossein Arzani and Scott TM Dawson. Data-driven cardiovascular flow modelling: examples and opportunities. *Journal of the Royal Society Interface*, 18 (175):20200802, 2021.

[113] Amirhossein Arzani, Jian-Xun Wang, Michael S Sacks, and Shawn C Shadden. Machine learning for cardiovascular biomechanics modeling: challenges and beyond. *Annals of Biomedical Engineering*, 50(6):615–627, 2022.

[114] Elías Cueto and Francisco Chinesta. Real time simulation for computational surgery: a review. *Advanced Modeling and Simulation in Engineering Sciences*, 1(1):1–18, 2014.

[115] Yubing Shi, Patricia Lawford, and Rodney Hose. Review of zero-d and 1-d models of blood flow in the cardiovascular system. *Biomedical engineering online*, 10(1): 1–38, 2011.

[116] Vuk Milišić and Alfio Quarteroni. Analysis of lumped parameter models for blood flow simulations and their relation with 1d models. *ESAIM: Mathematical modelling and numerical analysis*, 38(4):613–632, 2004.

[117] Leopold Grinberg, Elizabeth Cheever, Tomer Anor, Joseph R Madsen, and GE Karniadakis. Modeling blood flow circulation in intracranial arterial networks: a comparative 3d/1d simulation study. *Annals of biomedical engineering*, 39(1):297–309, 2011.

[118] Jiun-Jr Wang, Aoife B O'Brien, Nigel G Shrive, Kim H Parker, and John V Tyberg. Time-domain representation of ventricular-arterial coupling as a windkessel and wave system. *American Journal of Physiology-Heart and Circulatory Physiology*, 284(4):H1358–H1368, 2003.

[119] Sebastian Acosta, Charles Puelz, Béatrice Rivière, Daniel J Penny, and Craig G Rusin. Numerical method of characteristics for one-dimensional blood flow. *Journal of computational physics*, 294:96–109, 2015.

[120] NP Smith, AJ Pullan, and Peter J Hunter. An anatomically based model of transient coronary blood flow in the heart. *SIAM Journal on Applied mathematics*, 62(3):990–1018, 2002.

[121] Frans N Van de Vosse and Nikos Stergiopulos. Pulse wave propagation in the arterial tree. *Annual Review of Fluid Mechanics*, 43(1):467–499, 2011.

[122] Sally Epstein, Marie Willemet, Phil J Chowienczyk, and Jordi Alastruey. Reducing the number of parameters in 1d arterial blood flow modeling: less is more for patient-specific simulations. *American Journal of Physiology-Heart and Circulatory Physiology*, 309(1):H222–H234, 2015.

[123] Paul D Morris, Andrew Narracott, Hendrik von Tengg-Kobligk, Daniel Alejandro Silva Soto, Sarah Hsiao, Angela Lungu, Paul Evans, Neil W Bressloff, Patricia V Lawford, D Rodney Hose, et al. Computational fluid dynamics modelling in cardiovascular medicine. *Heart*, 102(1):18–28, 2016.

[124] K Gashi, EMH Bosboom, and FN Van de Vosse. The influence of model order reduction on the computed fractional flow reserve using parameterized coronary geometries. *Journal of Biomechanics*, 82:313–323, 2019.

[125] Mehran Mirramezani and Shawn C Shadden. A distributed lumped parameter model of blood flow. *Annals of Biomedical Engineering*, 48(12):2870–2886, 2020.

[126] Philippe Reymond, Fabienne Perren, François Lazeyras, and Nikos Stergiopulos. Patient-specific mean pressure drop in the systemic arterial tree, a comparison between 1-d and 3-d models. *Journal of biomechanics*, 45(15):2499–2505, 2012.

[127] Pablo Javier Blanco, Carlos Alberto Bulant, Lucas O Müller, GD Talou, C Guedes Bezerra, PA Lemos, and Raúl Antonino Feijóo. Comparison of 1d and 3d models for the estimation of fractional flow reserve. *Scientific reports*, 8 (1):1–12, 2018.

[128] Nan Xiao, Jordi Alastruey, and C Alberto Figueroa. A systematic comparison between 1-d and 3-d hemodynamics in compliant arterial models. *International journal for numerical methods in biomedical engineering*, 30(2):204, 2014.

[129] George Papadakis and Jean Raspaud. Wave propagation in stenotic vessels; theoretical analysis and comparison between 3d and 1d fluid–structure-interaction models. *Journal of Fluids and Structures*, 88:352–366, 2019.

[130] Alena Jonášová, Ondřej Bublík, and Jan Vimmr. A comparative study of 1d and 3d hemodynamics in patient-specific hepatic portal vein networks. *Applied and Computational Mechanics*, 8(2), 2014.

[131] Isidor Kokalari, Theodhor Karaja, and Maria Guerrisi. Review on lumped parameter method for modeling the blood flow in systemic arteries. *Journal of Biomedical Science and Engineering*, 2013.

[132] Pablo J Blanco, MR Pivello, SA Urquiza, and RA Feijóo. On the potentialities of 3d–1d coupled models in hemodynamics simulations. *Journal of biomechanics*, 42(7):919–930, 2009.

[133] Tiziano Passerini, Mariarita de Luca, Luca Formaggia, Alfio Quarteroni, and Alessandro Veneziani. A 3d/1d geometrical multiscale model of cerebral vasculature. *Journal of Engineering Mathematics*, 64(4):319–330, 2009.

[134] Sethuraman Sankaran, Mahdi Esmaily Moghadam, Andrew M Kahn, Elaine E Tseng, Julius M Guccione, and Alison L Marsden. Patient-specific multiscale modeling of blood flow for coronary artery bypass graft surgery. *Annals of biomedical engineering*, 40(10):2228–2242, 2012.

[135] Abhay B Ramachandra, Andrew M Kahn, and Alison L Marsden. Patient-specific simulations reveal significant differences in mechanical stimuli in venous and arterial coronary grafts. *Journal of cardiovascular translational research*, 9(4):279–290, 2016.

[136] Sethuraman Sankaran, Hyun Jin Kim, Gilwoo Choi, and Charles A Taylor. Uncertainty quantification in coronary blood flow simulations: impact of geometry, boundary conditions and blood viscosity. *Journal of biomechanics*, 49(12):2540–2547, 2016.

[137] Jordi Alastruey, Nan Xiao, Henry Fok, Tobias Schaeffter, and C Alberto Figueroa. On the impact of modelling assumptions in multi-scale, subject-specific models of aortic haemodynamics. *Journal of The Royal Society Interface*, 13(119):20160073, 2016.

[138] Raymond M Padmos, Tamás I Józsa, Wahbi K El-Bouri, Praneeta R Konduri, Stephen J Payne, and Alfons G Hoekstra. Coupling one-dimensional arterial blood flow to three-dimensional tissue perfusion models for in silico trials of acute ischaemic stroke. *Interface focus*, 11(1):20190125, 2021.

[139] Michele Girfoglio, Francesco Ballarin, Giuseppe Infantino, Francesca Nicoló, Andrea Montalto, Gianluigi Rozza, Roberto Scrofani, Marina Comisso, and Francesco Musumeci. Non-intrusive podi-rom for patient-specific aortic blood flow in presence of a lvad device. *Medical Engineering & Physics*, 107:103849, 2022.

[140] Marvin Fritz, Tobias Köppl, John Tinsley Oden, Andreas Wagner, Barbara Wohlmuth, and Chengyue Wu. A 1d–0d–3d coupled model for simulating blood flow and transport processes in breast tissue. *International Journal for Numerical Methods in Biomedical Engineering*, 38(7):e3612, 2022.

[141] John Leask Lumley. The structure of inhomogeneous turbulent flows. *Atmospheric turbulence and radio wave propagation*, pages 166–178, 1967.

[142] Kazufumi Ito and Sivaguru S Ravindran. A reduced-order method for simulation and control of fluid flows. *Journal of computational physics*, 143(2):403–425, 1998.

[143] Giovanni Stabile, Saddam Hijazi, Andrea Mola, Stefano Lorenzi, and Gianluigi Rozza. Pod-galerkin reduced order methods for cfd using finite volume discretisation: vortex shedding around a circular cylinder. *arXiv preprint arXiv:1701.03424*, 2017.

[144] Giovanni Stabile and Gianluigi Rozza. Finite volume pod-galerkin stabilised reduced order methods for the parametrised incompressible navier–stokes equations. *Computers & Fluids*, 173:273–284, 2018.

[145] Martin Hess, Alessandro Alla, Annalisa Quaini, Gianluigi Rozza, and Max Gunzburger. A localized reduced-order modeling approach for pdes with bifurcating

solutions. *Computer Methods in Applied Mechanics and Engineering*, 351:379–403, 2019.

[146] S Walton, O Hassan, and K Morgan. Reduced order modelling for unsteady fluid flow using proper orthogonal decomposition and radial basis functions. *Applied Mathematical Modelling*, 37(20-21):8930–8945, 2013.

[147] Dunhui Xiao, Pan Yang, Fangxin Fang, Jiansheng Xiang, Chris C Pain, and Ionel M Navon. Non-intrusive reduced order modelling of fluid–structure interactions. *Computer Methods in Applied Mechanics and Engineering*, 303:35–54, 2016.

[148] Luca Bertagna and Alessandro Veneziani. A model reduction approach for the variational estimation of vascular compliance by solving an inverse fluid–structure interaction problem. *Inverse Problems*, 30(5):055006, 2014.

[149] Kristin McLeod, Alfonso Caiazzo, Miguel A Fernández, Tommaso Mansi, Irene E Vignon-Clementel, Maxime Sermesant, Xavier Pennec, Younes Boudjemline, and Jean-Frederic Gerbeau. Atlas-based reduced models of blood flows for fast patient-specific simulations. In *International Workshop on Statistical Atlases and Computational Models of the Heart*, pages 95–104. Springer, 2010.

[150] Romain Guibert, Kristin Mcleod, Alfonso Caiazzo, Tommaso Mansi, Miguel A Fernández, Maxime Sermesant, Xavier Pennec, Irene E Vignon-Clementel, Younes Boudjemline, and Jean-Frédéric Gerbeau. Group-wise construction of reduced models for understanding and characterization of pulmonary blood flows from medical images. *Medical image analysis*, 18(1):63–82, 2014.

[151] Francesco Ballarin, Elena Faggiano, Sonia Ippolito, Andrea Manzoni, Alfio Quarteroni, Gianluigi Rozza, and Roberto Scrofani. Fast simulations of patient-specific haemodynamics of coronary artery bypass grafts based on a pod–galerkin method and a vascular shape parametrization. *Journal of Computational Physics*, 315:609–628, 2016.

[152] Marco Tezzele, Francesco Ballarin, and Gianluigi Rozza. Combined parameter and model reduction of cardiovascular problems by means of active subspaces and pod-galerkin methods. In *Mathematical and numerical modeling of the cardiovascular system and applications*, pages 185–207. Springer, 2018.

[153] Clarence W Rowley, Tim Colonius, and Richard M Murray. Model reduction for compressible flows using pod and galerkin projection. *Physica D: Nonlinear Phenomena*, 189(1-2):115–129, 2004.

[154] María-Luisa Rapún and José M Vega. Reduced order models based on local pod plus galerkin projection. *Journal of Computational Physics*, 229(8):3046–3063, 2010.

[155] Michel Bergmann, C-H Bruneau, and Angelo Iollo. Enablers for robust pod models. *Journal of Computational Physics*, 228(2):516–538, 2009.

[156] Saifon Chaturantabut and Danny C Sorensen. Nonlinear model reduction via discrete empirical interpolation. *SIAM Journal on Scientific Computing*, 32(5): 2737–2764, 2010.

[157] Kanchan Kapoor, Balbir Singh, and Inder Jit Dewan. Variations in the configuration of the circle of willis. *Anatomical science international*, 83(2):96–106, 2008.

[158] Thuan Lieu, Charbel Farhat, and Michel Lesoinne. Reduced-order fluid/structure modeling of a complete aircraft configuration. *Computer methods in applied mechanics and engineering*, 195(41-43):5730–5742, 2006.

[159] I Kalashnikova, MF Barone, and MR Brake. A stable galerkin reduced order model for coupled fluid–structure interaction problems. *International Journal for Numerical Methods in Engineering*, 95(2):121–144, 2013.

[160] A Placzek, D-M Tran, and R Ohayon. A nonlinear pod-galerkin reduced-order model for compressible flows taking into account rigid body motions. *Computer methods in applied mechanics and engineering*, 200(49-52):3497–3514, 2011.

[161] Francesco Ballarin and Gianluigi Rozza. Pod–galerkin monolithic reduced order models for parametrized fluid-structure interaction problems. *International Journal for Numerical Methods in Fluids*, 82(12):1010–1034, 2016.

[162] Angelo Iollo, Stéphane Lanteri, and J-A Désidéri. Stability properties of pod–galerkin approximations for the compressible navier–stokes equations. *Theoretical and Computational Fluid Dynamics*, 13(6):377–396, 2000.

[163] Sirod Sirisup and George E Karniadakis. A spectral viscosity method for correcting the long-term behavior of pod models. *Journal of Computational Physics*, 194(1):92–116, 2004.

[164] Cristóbal Bertoglio, Alfonso Caiazzo, and Miguel A Fernández. Fractional-step schemes for the coupling of distributed and lumped models in hemodynamics. *SIAM Journal on Scientific Computing*, 35(3):B551–B575, 2013.

[165] Yi Wang, Bo Yu, Zhizhu Cao, Weizhong Zou, and Guojun Yu. A comparative study of pod interpolation and pod projection methods for fast and accurate prediction of heat transfer problems. *International Journal of Heat and Mass Transfer*, 55(17-18):4827–4836, 2012.

[166] Dunhui Xiao, Fangxin Fang, Christopher Pain, and Guangwei Hu. Non-intrusive reduced-order modelling of the navier–stokes equations based on rbf interpolation. *International Journal for Numerical Methods in Fluids*, 79(11):580–595, 2015.

[167] Arash Hajisharifi, Francesco Romanò, Michele Girfoglio, Andrea Beccari, Domenico Bonanni, and Gianluigi Rozza. A non-intrusive data-driven reduced order model for parametrized cfd-dem numerical simulations. *Journal of Computational Physics*, 491:112355, 2023.

[168] Dunhui Xiao, Fangxin Fang, Andrew G Buchan, Christopher C Pain, Ionel M Navon, and Ann Muggeridge. Non-intrusive reduced order modelling of the navier–stokes equations. *Computer Methods in Applied Mechanics and Engineering*, 293:522–541, 2015.

[169] Francesco Ballarin, Elena Faggiano, Andrea Manzoni, Alfio Quarteroni, Gianluigi Rozza, Sonia Ippolito, Carlo Antona, and Roberto Scrofani. Numerical modeling of hemodynamics scenarios of patient-specific coronary artery bypass grafts. *Biomechanics and modeling in mechanobiology*, 16(4):1373–1399, 2017.

[170] Suyue Han, Clemens M Schirmer, and Yahya Modarres-Sadeghi. A reduced-order model of a patient-specific cerebral aneurysm for rapid evaluation and treatment planning. *Journal of Biomechanics*, page 109653, 2020.

[171] Michele Girfoglio, L Scandurra, Francesco Ballarin, Giuseppe Infantino, Francesca Nicolo, Andrea Montalto, Gianluigi Rozza, Roberto Scrofani, Marina

Comisso, and Francesco Musumeci. Non-intrusive data-driven rom framework for hemodynamics problems. *Acta mechanica sinica*, 37(7):1183–1191, 2021.

[172] Zakia Zainib, Francesco Ballarin, Stephen Fremes, Piero Triverio, Laura Jiménez-Juan, and Gianluigi Rozza. Reduced order methods for parametric optimal flow control in coronary bypass grafts, toward patient-specific data assimilation. *International Journal for Numerical Methods in Biomedical Engineering*, 37(12): e3367, 2021.

[173] Peter J Schmid. Dynamic mode decomposition of numerical and experimental data. *Journal of fluid mechanics*, 656:5–28, 2010.

[174] Milad Habibi, Scott Dawson, and Amirhossein Arzani. Data-driven pulsatile blood flow physics with dynamic mode decomposition. *Fluids*, 5(3):111, 2020.

[175] J Nathan Kutz, Steven L Brunton, Bingni W Brunton, and Joshua L Proctor. *Dynamic mode decomposition: data-driven modeling of complex systems*. SIAM, 2016.

[176] Matthew O Williams, Ioannis G Kevrekidis, and Clarence W Rowley. A data–driven approximation of the koopman operator: Extending dynamic mode decomposition. *Journal of Nonlinear Science*, 25(6):1307–1346, 2015.

[177] Milad Habibi, Scott Dawson, and Amirhossein Arzani. Reduced order modeling of pulsatile blood flow: multistage dynamic mode decomposition with control. In *APS Division of Fluid Dynamics Meeting Abstracts*, pages Q29–009, 2019.

[178] Milad Habibi, Roshan M D'Souza, Scott TM Dawson, and Amirhossein Arzani. Integrating multi-fidelity blood flow data with reduced-order data assimilation. *Computers in Biology and Medicine*, 135:104566, 2021.

[179] Eduardo Rodríguez-López, Douglas W Carter, and Bharathram Ganapathisubramani. Dynamic mode decomposition-based reconstructions for fluid–structure interactions: An application to membrane wings. *Journal of Fluids and Structures*, 104:103315, 2021.

[180] Soledad Le Clainche and José M Vega. Higher order dynamic mode decomposition. *SIAM Journal on Applied Dynamical Systems*, 16(2):882–925, 2017.

[181] J Nathan Kutz, Xing Fu, and Steven L Brunton. Multiresolution dynamic mode decomposition. *SIAM Journal on Applied Dynamical Systems*, 15(2):713–735, 2016.

[182] Daniel Dylewsky, Molei Tao, and J Nathan Kutz. Dynamic mode decomposition for multiscale nonlinear physics. *Physical Review E*, 99(6):063311, 2019.

[183] Hannah Lu and Daniel M Tartakovsky. Lagrangian dynamic mode decomposition for construction of reduced-order models of advection-dominated phenomena. *Journal of Computational Physics*, 407:109229, 2020.

[184] Giuseppe Di Labbio and Lyes Kadem. Reduced-order modeling of left ventricular flow subject to aortic valve regurgitation. *Physics of Fluids*, 31(3):031901, 2019.

[185] George C Bourantas, Mehdi Ghommem, George C Kagadis, Konstantinos Katsanos, Vassilis C Loukopoulos, Vasilis N Burganos, and George C Nikiforidis. Real-time tumor ablation simulation based on the dynamic mode decomposition method. *Medical physics*, 41(5):053301, 2014.

[186] Víctor Beltrán, Soledad Le Clainche, and José M Vega. An adaptive data-driven reduced order model based on higher order dynamic mode decomposition. *Journal of Scientific Computing*, 92(1):12, 2022.

[187] Quincy A Huhn, Mauricio E Tano, Jean C Ragusa, and Youngsoo Choi. Parametric dynamic mode decomposition for reduced order modeling. *Journal of Computational Physics*, 475:111852, 2023.

[188] Gianluigi Rozza, Dinh Bao Phuong Huynh, and Anthony T Patera. Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations: application to transport and continuum mechanics. *Archives of Computational Methods in Engineering*, 15(3):229, 2008.

[189] Jan S Hesthaven, Gianluigi Rozza, Benjamin Stamm, et al. *Certified reduced basis methods for parametrized partial differential equations*, volume 590. Springer, 2016.

[190] Toni Lassila and Gianluigi Rozza. Parametric free-form shape design with pde models and reduced basis method. *Computer Methods in Applied Mechanics and Engineering*, 199(23-24):1583–1592, 2010.

[191] Karen Veroy, Christophe Prud'Homme, Dimitrios Rovas, and Anthony Patera. A posteriori error bounds for reduced-basis approximation of parametrized noncoercive and nonlinear elliptic partial differential equations. In *16th AIAA Computational Fluid Dynamics Conference*, page 3847, 2003.

[192] Andrea Manzoni, Alfio Quarteroni, and Gianluigi Rozza. Model reduction techniques for fast blood flow simulation in parametrized geometries. *International journal for numerical methods in biomedical engineering*, 28(6-7):604–625, 2012.

[193] Toni Lassila, Andrea Manzoni, Alfio Quarteroni, and Gianluigi Rozza. A reduced computational and geometrical framework for inverse problems in hemodynamics. *International journal for numerical methods in biomedical engineering*, 29(7):741–776, 2013.

[194] Claudia M Colciago and Simone Deparis. Reduced numerical approximation of reduced fluid-structure interaction problems with applications in hemodynamics. *Frontiers in Applied Mathematics and Statistics*, 4:18, 2018.

[195] Alfio Quarteroni and Gianluigi Rozza. Numerical solution of parametrized navier–stokes equations by reduced basis methods. *Numerical Methods for Partial Differential Equations: An International Journal*, 23(4):923–948, 2007.

[196] Simone Deparis and Gianluigi Rozza. Reduced basis method for multi-parameter-dependent steady navier–stokes equations: applications to natural convection in a cavity. *Journal of Computational Physics*, 228(12):4359–4378, 2009.

[197] Toni Lassila, Alfio Quarteroni, and Gianluigi Rozza. A reduced basis model with parametric coupling for fluid-structure interaction problems. *SIAM Journal on Scientific Computing*, 34(2):A1187–A1213, 2012.

[198] Francisco Chinesta, Pierre Ladeveze, and Elias Cueto. A short review on model order reduction based on proper generalized decomposition. *Archives of Computational Methods in Engineering*, 18(4):395, 2011.

[199] Francisco Chinesta, Roland Keunings, and Adrien Leygue. *The proper generalized decomposition for advanced numerical simulations: a primer*. Springer Science & Business Media, 2013.

[200] Antoine Dumon, Cyrille Allery, and Amine Ammar. Proper generalized decomposition method for incompressible navier–stokes equations with a spectral discretization. *Applied Mathematics and Computation*, 219(15):8145–8162, 2013.

[201] Pierre Ladeveze and Ludovic Chamoin. On the verification of model reduction methods based on the proper generalized decomposition. *Computer Methods in Applied Mechanics and Engineering*, 200(23-24):2032–2047, 2011.

[202] Francisco Chinesta, Amine Ammar, Adrien Leygue, and Roland Keunings. An overview of the proper generalized decomposition with applications in computational rheology. *Journal of Non-Newtonian Fluid Mechanics*, 166(11):578–592, 2011.

[203] Amine Ammar, Francisco Chinesta, Elías Cueto, and Manuel Doblaré. Proper generalized decomposition of time-multiscale models. *International Journal for Numerical Methods in Engineering*, 90(5):569–596, 2012.

[204] David Néron and Pierre Ladevèze. Proper generalized decomposition for multiscale and multiphysics problems. *Archives of Computational Methods in Engineering*, 17:351–372, 2010.

[205] Francisco Chinesta, Amine Ammar, and Elıas Cueto. Proper generalized decomposition of multiscale models. *International Journal for Numerical Methods in Engineering*, 83(8-9):1114–1132, 2010.

[206] Steven L Brunton and J Nathan Kutz. *Data-driven science and engineering: Machine learning, dynamical systems, and control.* Cambridge University Press, 2019.

[207] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[208] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5):359–366, 1989.

[209] Steven L Brunton, Bernd R Noack, and Petros Koumoutsakos. Machine learning for fluid mechanics. *Annual review of fluid mechanics*, 52:477–508, 2020.

[210] Athanasios Voulodimos, Nikolaos Doulamis, Anastasios Doulamis, Eftychios Protopapadakis, et al. Deep learning for computer vision: A brief review. *Computational intelligence and neuroscience*, 2018, 2018.

[211] Tom Young, Devamanyu Hazarika, Soujanya Poria, and Erik Cambria. Recent trends in deep learning based natural language processing. *ieee Computational intelligenCe magazine*, 13(3):55–75, 2018.

[212] Jan S Hesthaven and Stefano Ubbiali. Non-intrusive reduced order modeling of nonlinear problems using neural networks. *Journal of Computational Physics*, 363:55–78, 2018.

[213] Omer San, Romit Maulik, and Mansoor Ahmed. An artificial neural network framework for reduced order modeling of transient flows. *Communications in Nonlinear Science and Numerical Simulation*, 77:271–287, 2019.

[214] Stefania Fresca and Andrea Manzoni. Pod-dl-rom: enhancing deep learning-based reduced order models for nonlinear parametrized pdes by proper orthogonal decomposition. *Computer Methods in Applied Mechanics and Engineering*, 388: 114181, 2022.

[215] Qian Wang, Jan S Hesthaven, and Deep Ray. Non-intrusive reduced order modeling of unsteady flows using artificial neural networks with application to a combustion problem. *Journal of computational physics*, 384:289–307, 2019.

[216] Caterina Balzotti, Pierfrancesco Siena, Michele Girfoglio, Annalisa Quaini, and Gianluigi Rozza. A data-driven reduced order method for parametric optimal blood flow control: application to coronary bypass graft. *arXiv preprint arXiv:2206.15384*, 2022.

[217] Qian Wang, Nicolò Ripamonti, and Jan S Hesthaven. Recurrent neural network closure of parametric pod-galerkin reduced-order models based on the mori-zwanzig formalism. *Journal of Computational Physics*, 410:109402, 2020.

[218] Han Gao, Jian-Xun Wang, and Matthew J Zahr. Non-intrusive model reduction of large-scale, nonlinear dynamical systems using deep learning. *Physica D: Nonlinear Phenomena*, 412:132614, 2020.

[219] Stefania Fresca, Andrea Manzoni, Luca Dedè, and Alfio Quarteroni. Pod-enhanced deep learning-based reduced order models for the real-time simulation of cardiac electrophysiology in the left atrium. *Frontiers in physiology*, page 1431, 2021.

[220] Stefania Fresca and Andrea Manzoni. Real-time simulation of parameter-dependent fluid flows through deep learning-based reduced order models. *Fluids*, 6(7):259, 2021.

[221] Thomas Daniel, Fabien Casenave, Nissrine Akkari, and David Ryckelynck. Model order reduction assisted by deep neural networks (rom-net). *Advanced Modeling and Simulation in Engineering Sciences*, 7:1–27, 2020.

[222] David Amsallem, Matthew J Zahr, and Charbel Farhat. Nonlinear model order reduction based on local reduced-order bases. *International Journal for Numerical Methods in Engineering*, 92(10):891–916, 2012.

[223] Hunor Csala, Scott Dawson, and Amirhossein Arzani. Comparing different nonlinear dimensionality reduction techniques for data-driven unsteady fluid flow modeling. *Physics of Fluids*, 34(11), 2022.

[224] David R Rutkowski, Alejandro Roldán-Alzate, and Kevin M Johnson. Enhancement of cerebrovascular 4d flow mri velocity fields using machine learning and computational fluid dynamics simulation data. *Scientific reports*, 11(1):1–11, 2021.

[225] Liang Liang, Wenbin Mao, and Wei Sun. A feasibility study of deep learning for predicting hemodynamics of human thoracic aorta. *Journal of biomechanics*, 99: 109544, 2020.

[226] George Em Karniadakis, Ioannis G Kevrekidis, Lu Lu, Paris Perdikaris, Sifan Wang, and Liu Yang. Physics-informed machine learning. *Nature Reviews Physics*, 3(6):422–440, 2021.

[227] Shengze Cai, Zhiping Mao, Zhicheng Wang, Minglang Yin, and George Em Karniadakis. Physics-informed neural networks (pinns) for fluid mechanics: A review. *Acta Mechanica Sinica*, 37(12):1727–1738, 2021.

[228] Luning Sun, Han Gao, Shaowu Pan, and Jian-Xun Wang. Surrogate modeling for fluid flows based on physics-constrained deep learning without simulation data. *Computer Methods in Applied Mechanics and Engineering*, 361:112732, 2020.

[229] Christopher J Arthurs and Andrew P King. Active training of physics-informed neural networks to aggregate and interpolate parametric solutions to the navier-stokes equations. *Journal of Computational Physics*, 438:110364, 2021.

[230] Oliver Hennigh, Susheela Narasimhan, Mohammad Amin Nabian, Akshay Subramaniam, Kaustubh Tangsali, Zhiwei Fang, Max Rietmann, Wonmin Byeon, and Sanjay Choudhry. Nvidia simnetâ„¢: An ai-accelerated multi-physics simulation framework. In *International Conference on Computational Science*, pages 447–461. Springer, 2021.

[231] Philipp Moser, Wolfgang Fenz, Stefan Thumfart, Isabell Ganitzer, and Michael Giretzlehner. Modeling of 3d blood flows with physics-informed neural networks: Comparison of network architectures. *Fluids*, 8(2):46, 2023.

[232] Georgios Kissas, Yibo Yang, Eileen Hwuang, Walter R Witschey, John A Detre, and Paris Perdikaris. Machine learning in cardiovascular flows modeling: Predicting arterial blood pressure from non-invasive 4d flow mri data using physics-informed neural networks. *Computer Methods in Applied Mechanics and Engineering*, 358:112623, 2020.

[233] Lu Lu, Raphael Pestourie, Wenjie Yao, Zhicheng Wang, Francesc Verdugo, and Steven G Johnson. Physics-informed neural networks with hard constraints for inverse design. *SIAM Journal on Scientific Computing*, 43(6):B1105–B1132, 2021.

[234] Xiaowei Jin, Shengze Cai, Hui Li, and George Em Karniadakis. Nsfnets (navier-stokes flow nets): Physics-informed neural networks for the incompressible navier-stokes equations. *Journal of Computational Physics*, 426:109951, 2021.

[235] Stefano Buoso, Thomas Joyce, and Sebastian Kozerke. Personalising left-ventricular biophysical models of the heart using parametric physics-informed neural networks. *Medical Image Analysis*, 71:102066, 2021.

[236] Francisco Sahli Costabal, Simone Pezzuto, and Paris Perdikaris. δ-pinns:

physics-informed neural networks on complex geometries. *arXiv preprint arXiv:2209.03984*, 2022.

[237] Filipe de Avila Belbute-Peres, Yi-fan Chen, and Fei Sha. Hyperpinn: Learning parameterized differential equations with physics-informed hypernetworks. In *The Symbiosis of Deep Learning and Differential Equations*, 2021.

[238] Shaan Desai, Marios Mattheakis, Hayden Joy, Pavlos Protopapas, and Stephen Roberts. One-shot transfer learning of physics-informed neural networks. *arXiv preprint arXiv:2110.11286*, 2021.

[239] Ameya D Jagtap, Kenji Kawaguchi, and George Em Karniadakis. Adaptive activation functions accelerate convergence in deep and physics-informed neural networks. *Journal of Computational Physics*, 404:109136, 2020.

[240] Lu Lu, Xuhui Meng, Zhiping Mao, and George Em Karniadakis. Deepxde: A deep learning library for solving differential equations. *SIAM Review*, 63(1):208–228, 2021.

[241] Chenxi Wu, Min Zhu, Qinyang Tan, Yadhu Kartha, and Lu Lu. A comprehensive study of non-adaptive and residual-based adaptive sampling for physics-informed neural networks. *Computer Methods in Applied Mechanics and Engineering*, 403: 115671, 2023.

[242] Han Gao, Luning Sun, and Jian-Xun Wang. Super-resolution and denoising of fluid flow using physics-informed convolutional neural networks without high-resolution labels. *Physics of Fluids*, 33(7):073603, 2021.

[243] Mojtaba F Fathi, Isaac Perez-Raya, Ahmadreza Baghaie, Philipp Berg, Gabor Janiga, Amirhossein Arzani, and Roshan M Dâ€™Souza. Super-resolution and denoising of 4d-flow mri using physics-informed deep neural nets. *Computer Methods and Programs in Biomedicine*, 197:105729, 2020.

[244] Amirhossein Arzani, Jian-Xun Wang, and Roshan M D'Souza. Uncovering near-wall blood flow from sparse data with physics-informed neural networks. *Physics of Fluids*, 33(7):071905, 2021.

[245] Mohammad Sarabian, Hessam Babaee, and Kaveh Laksari. Physics-informed neural networks for brain hemodynamic predictions using medical imaging. *IEEE Transactions on Medical Imaging*, 2022.

[246] Minglang Yin, Xiaoning Zheng, Jay D Humphrey, and George Em Karniadakis. Non-invasive inference of thrombus material properties with physics-informed neural networks. *Computer Methods in Applied Mechanics and Engineering*, 375: 113603, 2021.

[247] Fergus Shone, Nishant Ravikumar, Toni Lassila, Michael MacRaild, Yongxing Wang, Zeike A Taylor, Peter Jimack, Erica Dallâ€™Armellina, and Alejandro F Frangi. Deep physics-informed super-resolution of cardiac 4d-flow mri. In *International Conference on Information Processing in Medical Imaging*, pages 511–522. Springer, 2023.

[248] Maziar Raissi, Alireza Yazdani, and George Em Karniadakis. Hidden fluid mechanics: Learning velocity and pressure fields from flow visualizations. *Science*, 367 (6481):1026–1030, 2020.

[249] Shengze Cai, Zhicheng Wang, Sifan Wang, Paris Perdikaris, and George Em Karniadakis. Physics-informed neural networks for heat transfer problems. *Journal of Heat Transfer*, 143(6), 2021.

[250] Dehong Fang and Jifu Tan. Immersed boundary-physics informed machine learning approach for fluid–solid coupling. *Ocean Engineering*, 263:112360, 2022.

[251] Jungeun Kim, Kookjin Lee, Dongeun Lee, Sheo Yon Jhin, and Noseong Park. Dpm: a novel training method for physics-informed neural networks in extrapolation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 8146–8154, 2021.

[252] Ameya D Jagtap, Ehsan Kharazmi, and George Em Karniadakis. Conservative physics-informed neural networks on discrete domains for conservation laws: Applications to forward and inverse problems. *Computer Methods in Applied Mechanics and Engineering*, 365:113028, 2020.

[253] Ameya D Jagtap and George E Karniadakis. Extended physics-informed neural networks (xpinns): A generalized space-time domain decomposition based deep

learning framework for nonlinear partial differential equations. In *AAAI Spring Symposium: MLPS*, pages 2002–2041, 2021.

[254] Khemraj Shukla, Ameya D Jagtap, and George Em Karniadakis. Parallel physics-informed neural networks via domain decomposition. *Journal of Computational Physics*, 447:110683, 2021.

[255] Xuhui Meng, Zhen Li, Dongkun Zhang, and George Em Karniadakis. Ppinn: Parareal physics-informed neural network for time-dependent pdes. *Computer Methods in Applied Mechanics and Engineering*, 370:113250, 2020.

[256] Shengze Cai, Zhicheng Wang, Lu Lu, Tamer A Zaki, and George Em Karniadakis. Deepm&mnet: Inferring the electroconvection multiphysics fields based on operator approximation by neural networks. *Journal of Computational Physics*, 436:110296, 2021.

[257] Zhiping Mao, Lu Lu, Olaf Marxen, Tamer A Zaki, and George Em Karniadakis. Deepm&mnet for hypersonics: Predicting the coupled flow and finite-rate chemistry behind a normal shock using neural-network approximation of operators. *Journal of computational physics*, 447:110698, 2021.

[258] Xabier Morales, Jordi Mill, Kristine A Juhl, Andy Olivares, Guillermo Jimenez-Perez, Rasmus R Paulsen, and Oscar Camara. Deep learning surrogate of computational fluid dynamics for thrombus formation risk in the left atrial appendage. In *Statistical Atlases and Computational Models of the Heart. Multi-Sequence CMR Segmentation, CRT-EPiggy and LV Full Quantification Challenges: 10th International Workshop, STACOM 2019, Held in Conjunction with MICCAI 2019, Shenzhen, China, October 13, 2019, Revised Selected Papers 10*, pages 157–166. Springer, 2020.

[259] Edward Ferdian, Avan Suinesiaputra, David J Dubowitz, Debbie Zhao, Alan Wang, Brett Cowan, and Alistair A Young. 4dflownet: super-resolution 4d flow mri using deep learning and computational fluid dynamics. *Frontiers in Physics*, 8:138, 2020.

[260] Ramtin Gharleghi, Arcot Sowmya, and Susann Beier. Transient wall shear stress estimation in coronary bifurcations using convolutional neural networks. *Computer Methods and Programs in Biomedicine*, 225:107013, 2022.

[261] Minglang Yin, Ehsan Ban, Bruno V Rego, Enrui Zhang, Cristina Cavinato, Jay D Humphrey, and George Em Karniadakis. Simulating progressive intramural damage leading to aortic dissection using deeponet: an operator–regression neural network. *Journal of the Royal Society Interface*, 19(187):20210670, 2022.

[262] Liang Liang, Minliang Liu, Caitlin Martin, John A Elefteriades, and Wei Sun. A machine learning approach to investigate the relationship between shape features and numerically predicted risk of ascending aortic aneurysm. *Biomechanics and modeling in mechanobiology*, 16:1519–1533, 2017.

[263] Liang Liang, Minliang Liu, Caitlin Martin, and Wei Sun. A machine learning approach as a surrogate of finite element analysis–based inverse method to estimate the zero-pressure geometry of human thoracic aorta. *International journal for numerical methods in biomedical engineering*, 34(8):e3103, 2018.

[264] Connor Shorten and Taghi M Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of big data*, 6(1):1–48, 2019.

[265] Pau Romero, Miguel Lozano, Francisco Martínez-Gil, Dolors Serra, Rafael Sebastián, Pablo Lamata, and Ignacio García-Fernández. Clinically-driven virtual patient cohorts generation: An application to aorta. *Frontiers in Physiology*, page 1375, 2021.

[266] Haoran Dou, Seppo Virtanen, Nishant Ravikumar, and Alejandro F Frangi. A generative shape compositional framework: Towards representative populations of virtual heart chimaeras. *arXiv preprint arXiv:2210.01607*, 2022.

[267] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017.

[268] Ali Kashefi and Tapan Mukerji. Physics-informed pointnet: A deep learning solver for steady-state incompressible flows and thermal fields on multiple sets of irregular geometries. *Journal of Computational Physics*, 468:111510, 2022.

[269] Tianping Chen and Hong Chen. Universal approximation to nonlinear operators

by neural networks with arbitrary activation functions and its application to dynamical systems. *IEEE Transactions on Neural Networks*, 6(4):911–917, 1995.

[270] Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Neural operator: Graph kernel network for partial differential equations. *arXiv preprint arXiv:2003.03485*, 2020.

[271] Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. *arXiv preprint arXiv:2010.08895*, 2020.

[272] Sifan Wang, Hanwen Wang, and Paris Perdikaris. Learning the solution operator of parametric partial differential equations with physics-informed deeponets. *Science advances*, 7(40):eabi8605, 2021.

[273] Minglang Yin, Enrui Zhang, Yue Yu, and George Em Karniadakis. Interfacing finite elements with deep neural operators for fast multiscale modeling of mechanics problems. *Computer Methods in Applied Mechanics and Engineering*, 402: 115027, 2022.

[274] Sifan Wang and Paris Perdikaris. Long-time integration of parametric evolution equations with physics-informed deeponets. *Journal of Computational Physics*, 475:111855, 2023.

[275] Amirhossein Arzani, Lingxiao Yuan, Pania Newell, and Bei Wang. Interpreting and generalizing deep learning in physics-based problems with functional linear models. *arXiv preprint arXiv:2307.04569*, 2023.

[276] Samuel Lanthaler, Siddhartha Mishra, and George E Karniadakis. Error estimates for deeponets: A deep learning framework in infinite dimensions. *Transactions of Mathematics and Its Applications*, 6(1):tnac001, 2022.

[277] Marco Viceconti, Claudio Cobelli, Tarek Haddad, Adam Himes, Boris Kovatchev, and Mark Palmer. In silico assessment of biomedical products: the conundrum of rare but not so rare events in two case studies. *Proceedings of the Institution*

of Mechanical Engineers, Part H: Journal of Engineering in Medicine, 231(5): 455–466, 2017.

[278] Ilana Buffenstein, Bree Kaneakua, Emily Taylor, Masako Matsunaga, So Yung Choi, Enrique Carrazana, Jason Viereck, Kore Kai Liow, and Arash Ghaffari-Rafi. Demographic recruitment bias of adults in united states randomized clinical trials by disease categories between 2008 to 2019: a systematic review and meta-analysis. *Scientific Reports*, 13(1):42, 2023.

[279] Badih Daou, Edison P Valle-Giler, Nohra Chalouhi, Robert M Starke, Stavropoula Tjoumakaris, David Hasan, Robert H Rosenwasser, Ryan Hebert, and Pascal Jabbour. Patency of the posterior communicating artery following treatment with the pipeline embolization device. *Journal of neurosurgery*, 126(2):564–569, 2017.

[280] Anna Luisa Kühn, Guilherme Dabus, Peter Kan, Ajay K Wakhloo, and Ajit S Puri. Flow-diverter stents for endovascular management of non-fetal posterior communicating artery aneurysmsâ€"analysis on aneurysm occlusion, vessel patency, and patient outcome. *Interventional Neuroradiology*, 24(4):363–374, 2018.

[281] Alejandro Enriquez-Marulanda, Krishnan Ravindran, Mohamed M Salem, Luis C Ascanio, Peter Kan, Visish M Srinivasan, Christoph J Griessenauer, Clemens M Schirmer, Abhi Jain, Justin M Moore, et al. Evaluation of radiological features of the posterior communicating artery and their impact on efficacy of saccular aneurysm treatment with the pipeline embolization device: a case series study. *World neurosurgery*, 125:e998–e1007, 2019.

[282] A Fleur Van Raamt, Willem PTM Mali, Peter Jan Van Laar, and Yolanda Van Der Graaf. The fetal variant of the circle of willis and its influence on the cerebral collateral circulation. *Cerebrovascular diseases*, 22(4):217–224, 2006.

[283] Kiarash Golshani, Andrew Ferrell, Ali Zomorodi, Tony P Smith, and Gavin W Britz. A review of the management of posterior communicating artery aneurysms in the modern era. *Surgical neurology international*, 1, 2010.

[284] Peeraphong Thiarawat, Behnam Rezai Jahromi, Danil A Kozyrev, Patcharin Intarakhao, Mario K Teo, Joham Choque-Velasquez, Mika Niemelä, and Juha

Hernesniemi. Are fetal-type posterior cerebral arteries associated with an increased risk of posterior communicating artery aneurysms? *Neurosurgery*, 84(6): 1306–1312, 2019.

[285] Amir Shaban, Karen C Albright, Amelia K Boehme, Sheryl Martin-Schild, et al. Circle of willis variants: fetal pca. *Stroke research and treatment*, 2013, 2013.

[286] Mario Zanaty, Nohra Chalouhi, Robert M Starke, Pascal Jabbour, Katherine O Ryken, Ketan R Bulsara, and David Hasan. Failure of the pipeline embolization device in posterior communicating artery aneurysms associated with a fetal posterior cerebral artery. *Case reports in vascular medicine*, 2016, 2016.

[287] Adam N Wallace, Yasha Kayan, Matthew J Austin, Josser E Delgado Almandoz, Mudassar Kamran, DeWitte T Cross III, Christopher J Moran, Joshua W Osbun, and Akash P Kansagra. Pipeline embolization of posterior communicating artery aneurysms associated with a fetal origin posterior cerebral artery. *Clinical neurology and neurosurgery*, 160:83–87, 2017.

[288] T. Becske, D.F. Kallmes, I. Saatci, C.G. McDougall, I. Szikora, G. Lanzino, C.J. Moran, H.H. Woo, D.K. Lopes, A.L. Berez, et al. Pipeline for uncoilable or failed aneurysms: results from a multicenter clinical trial. *Radiology*, 267(3):858–868, 2013.

[289] T. Becske, M.B. Potts, M. Shapiro, D.F. K., W. Brinjikji, I. Saatci, C.G. McDougall, I. Szikora, G. Lanzino, C.J. Moran, et al. Pipeline for uncoilable or failed aneurysms: 3-year follow-up results. *J. Neurosurg.*, 127(1):81–88, 2016.

[290] T. Becske, W. Brinjikji, M.B. Potts, D.F. Kallmes, M. Shapiro, C.J. Moran, E.I. Levy, C.G. McDougall, I. Szikora, G. Lanzino, et al. Long-term clinical and angiographic outcomes following pipeline embolization device treatment of complex internal carotid artery aneurysms: five-year results of the pipeline for uncoilable or failed aneurysms trial. *Neurosurgery*, 80(1):40–48, 2017.

[291] R.A. Hanel, D.F. Kallmes, D.K. Lopes, P.K. Nelson, A. Siddiqui, P. Jabbour, V.M. Pereira, I. Szikora, O.O. Zaidat, C. Bettegowda, et al. Prospective study on embolization of intracranial aneurysms with the pipeline device: the PREMIER study 1 year results. *J. Neurointervent. Surg.*, 12(1):62–66, 2020.

[292] D.F. Kallmes, W. Brinjikji, E. Boccardi, E. Ciceri, O. Diaz, R. Tawk, H. Woo, P. Jabbour, F. Albuquerque, R. Chapot, et al. Aneurysm study of pipeline in an observational registry (aspire). *Intervent. Neurol.*, 5(1-2):89–99, 2016.

[293] David F Kallmes, R Hanel, D Lopes, E Boccardi, Alain Bonafé, S Cekirge, D Fiorella, P Jabbour, E Levy, C McDougall, et al. International retrospective study of the pipeline embolization device: a multicenter aneurysm treatment study. *American Journal of Neuroradiology*, 36(1):108–115, 2015.

[294] M.C. Villa-Uriol, G. Berti, D.R. Hose, A. Marzo, A. Chiarini, J. Penrose, J. Pozo, J.G. Schmidt, P. Singh, R. Lycett, et al. @neurist complex information processing toolchain for the integrated management of cerebral aneurysms. *Interface Focus*, 1(3):308–319, 2011.

[295] Norman Juchler, Sabine Schilling, Philippe Bijlenga, Vartan Kurtcuoglu, and Sven Hirsch. Shape trumps size: image-based morphological analysis reveals that the 3d shape discriminates intracranial aneurysm disease status better than aneurysm size. *Frontiers in Neurology*, 13:809391, 2022.

[296] I. Larrabide, M. Kim, L. Augsburger, M.C. Villa-Uriol, D. Rüfenacht, and A.F. Frangi. Fast virtual deployment of self-expandable stents: method and in vitro evaluation for intracranial aneurysmal stenting. *Medical Image Anal.*, 16(3):721–730, 2012.

[297] Department of Health and Social Care. *Equity in Medical Devices: Independent Review.* Department of Health and Social Care, 2024. URL `https://assets.publishing.service.gov.uk/media/65e89e9e62ff48001a87b2d8/equity-in-medical-devices-independent-review-report-web-accessible.pdf`.

[298] MC Villa-Uriol, G Berti, DR Hose, A Marzo, A Chiarini, J Penrose, J Pozo, JG Schmidt, P Singh, R Lycett, et al. @ neurist complex information processing toolchain for the integrated management of cerebral aneurysms. *Interface Focus*, 1(3):308–319, 2011.

[299] Fengming Lin, Yan Xia, Shuang Song, Nishant Ravikumar, and Alejandro F Frangi. High-throughput 3dra segmentation of brain vasculature and aneurysms using deep learning. *Computer Methods and Programs in Biomedicine*, 230:107355, 2023.

[300] Rafik Ouared, Ignacio Larrabide, Olivier Brina, Pierre Bouillot, Gorislav Erceg, Hasan Yilmaz, Karl-Olof Lovblad, and Vitor Mendes Pereira. Computational fluid dynamics analysis of flow reduction induced by flow-diverting stents in intracranial aneurysms: a patient-unspecific hemodynamics change perspective. *Journal of NeuroInterventional Surgery*, 8(12):1288–1293, 2016.

[301] B. Chung, F. Mut, R. Kadirvel, R. Lingineni, D.F. Kallmes, and J.R. Cebral. Hemodynamic analysis of fast and slow aneurysm occlusions by flow diversion in rabbits. *J. Neurointerv. Surg.*, 7(12):931–935, 2015.

[302] F. Mut, M. Raschi, E. Scrivano, C. Bleise, J. Chudyk, R. Ceratto, P. Lylyk, and J.R. Cebral. Association between hemodynamic conditions and occlusion times after flow diversion in cerebral aneurysms. *J. Neurointerv. Surg.*, 7(4):286–290, 2015.

[303] Z Kulcsár, E Houdart, A Bonafé, G Parker, J Millar, AJP Goddard, S Renowden, G Gal, B Turowski, K Mitchell, et al. Intra-aneurysmal thrombosis as a possible cause of delayed aneurysm rupture after flow-diversion treatment. *American Journal of Neuroradiology*, 32(1):20–25, 2011.

[304] K Ravindran, MM Salem, AY Alturki, AJ Thomas, CS Ogilvy, and JM Moore. Endothelialization following flow diversion for intracranial aneurysms: a systematic review. *American Journal of Neuroradiology*, 40(2):295–301, 2019.

[305] Philipp Berg, Christina Iosif, Sebastien Ponsonnard, Catherine Yardin, Gábor Janiga, and Charbel Mounayer. Endothelialization of over-and undersized flow-diverter stents at covered vessel side branches: an in vivo and in silico study. *Journal of biomechanics*, 49(1):4–12, 2016.

[306] Nicolas Benard, Damien Coisne, Erwan Donal, and Robert Perrault. Experimental study of laminar blood flow through an artery treated by a stent implantation: characterisation of intra-stent wall shear stress. *Journal of biomechanics*, 36(7):991–998, 2003.

[307] Sunil Appanaboyina, Fernando Mut, Rainald Löhner, Christopher Putman, and Juan Cebral. Simulation of intracranial aneurysm stenting: techniques and challenges. *Computer Methods in Applied Mechanics and Engineering*, 198(45-46): 3567–3582, 2009.

[308] Matthew D Ford, Noam Alperin, Sung Hoon Lee, David W Holdsworth, and David A Steinman. Characterization of volumetric flow rate waveforms in the normal internal carotid and vertebral arteries. *Physiological measurement*, 26(4): 477, 2005.

[309] Juan R Cebral, Marcelo Adrián Castro, Sunil Appanaboyina, Christopher M Putman, Daniel Millan, and Alejandro F Frangi. Efficient pipeline for image-based patient-specific analysis of cerebral aneurysm hemodynamics: technique and sensitivity. *IEEE transactions on medical imaging*, 24(4):457–467, 2005.

[310] Luca Antiga, Marina Piccinelli, Lorenzo Botti, Bogdan Ene-Iordache, Andrea Remuzzi, and David A Steinman. An image-based modeling framework for patient-specific computational hemodynamics. *Medical & biological engineering & computing*, 46(11):1097–1112, 2008.

[311] Anthony Gruber, Max Gunzburger, Lili Ju, and Zhu Wang. A comparison of neural network architectures for data-driven reduced-order modeling. *Computer Methods in Applied Mechanics and Engineering*, 393:114764, 2022.

[312] Hrvoje Bogunović, José María Pozo, María Cruz Villa-Uriol, Charles BLM Majoie, Rene van den Berg, Hugo AF Gratama van Andel, Juan M Macho, Jordi Blasco, Luis San Román, and Alejandro F Frangi. Automated segmentation of cerebral vasculature with aneurysms in 3dra and tof-mra using geodesic active regions: An evaluation study. *Medical physics*, 38(1):210–222, 2011.

[313] Toni Lassila, Ali Sarrami-Foroushani, SeyedMostafa Hejazi, and Alejandro F Frangi. Population-specific modelling of between/within-subject flow variability in the carotid arteries of the elderly. *International journal for numerical methods in biomedical engineering*, 36(1):e3271, 2020.

[314] AJ Geers, I Larrabide, HG Morales, and Alejandro F Frangi. Approximating hemodynamics of cerebral aneurysms with steady flow simulations. *Journal of biomechanics*, 47(1):178–185, 2014.

[315] Melanie Wielicka, Jolanta Neubauer-Geryk, Grzegorz Kozera, and Leszek Bieniaszewski. Clinical application of pulsatility index. *Medical Research Journal*, 5 (3):201–210, 2020.

[316] M Schöning, Jochen Walter, and Peter Scheel. Estimation of cerebral blood flow through color duplex sonography of the carotid and vertebral arteries in healthy adults. *Stroke*, 25(1):17–22, 1994.

[317] Takashi Matsuo, Shin-ichi Watanabe, Mutsumi Sorimachi, Mizuta Kanda, Yuji Ohta, and Tatsuhisa Takahashi. Blood flow velocity waveforms in the middle cerebral artery at rest and during exercise. *Artificial Life and Robotics*, 16(3): 398–402, 2011.

[318] Jian Yu, Chao Yan, and Mengwu Guo. Non-intrusive reduced-order modeling for fluid problems: A brief review. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 233(16):5896–5912, 2019.

[319] Claire E Heaney, Xiangqi Liu, Hanna Go, Zef Wolffs, Pablo Salinas, Ionel M Navon, and Christopher C Pain. Extending the capabilities of data-driven reduced-order models to make predictions for unseen scenarios: applied to flow around buildings. *Frontiers in Physics*, page 563, 2022.

[320] Claire E Heaney, Zef Wolffs, Jón Atli Tómasson, Lyes Kahouadji, Pablo Salinas, André Nicolle, Ionel M Navon, Omar K Matar, Narakorn Srinil, and Christopher C Pain. An ai-based non-intrusive reduced-order model for extended domains applied to multiphase flow in pipes. *Physics of Fluids*, 34(5):055111, 2022.

[321] Alessa Hering, Lasse Hansen, Tony CW Mok, Albert CS Chung, Hanna Siebert, Stephanie Häger, Annkristin Lange, Sven Kuckertz, Stefan Heldmann, Wei Shao, et al. Learn2reg: comprehensive multi-task medical image registration challenge, dataset and evaluation in the era of deep learning. *IEEE Transactions on Medical Imaging*, 2022.

[322] Kimberley M Timmins, Irene C van der Schaaf, Edwin Bennink, Ynte M Ruigrok, Xingle An, Michael Baumgartner, Pascal Bourdon, Riccardo De Feo, Tommaso Di Noto, Florian Dubost, et al. Comparing methods of detecting and segmenting unruptured intracranial aneurysms on tof-mras: The adam challenge. *Neuroimage*, 238:118216, 2021.

[323] YD Cho, WJ Lee, KM Kim, H-S Kang, JE Kim, and Moon Hee Han. Stent-assisted coil embolization of posterior communicating artery aneurysms. *American Journal of Neuroradiology*, 34(11):2171–2176, 2013.

[324] S Asnafi, A Rouchaud, L Pierot, W Brinjikji, MH Murad, and DF Kallmes. Efficacy and safety of the woven endobridge (web) device for the treatment of intracranial aneurysms: a systematic review and meta-analysis. *American Journal of Neuroradiology*, 37(12):2287–2292, 2016.

[325] P Bhogal, I Lylyk, J Chudyk, N Perez, C Bleise, and P Lylyk. The contourâ€”early human experience of a novel aneurysm occlusion device. *Clinical neuroradiology*, 31:147–154, 2021.

[326] Marco Viceconti, Francesco Pappalardo, Blanca Rodriguez, Marc Horner, Jeff Bischoff, and Flora Musuamba Tshinanu. In silico trials: Verification, validation and uncertainty quantification of predictive models used in the regulatory evaluation of biomedical products. *Methods*, 185:120–127, 2021.

[327] Helena E Riggs and Charles Rupp. Variation in form of circle of willis: the relation of the variations to collateral circulation: anatomic analysis. *Archives of Neurology*, 8(1):8–14, 1963.

[328] A El Khamlichi, M Azouzi, F Bellakhdar, A Ouhcein, and A Lahlaidi. Anatomic configuration of the circle of willis in the adult studied by injection technics. apropos of 100 brains. *Neuro-chirurgie*, 31(4):287–293, 1985.

[329] CM Fisher. Anatomical variations. *Vasc Dis*, 2:99–105, 1965.

[330] Guy Lazorthes, André Gouazé, Jean Jacques Santini, and Georges Salamon. Le cercle artériel du cerveau (circulus arteriosus cerebri). *Anatomia Clinica*, 1:241–257, 1979.

[331] Behzad Eftekhar, Majid Dadmehr, Saeed Ansari, Mohammad Ghodsi, Bashir Nazparvar, and Ebrahim Ketabchi. Are the distributions of variations of circle of willis different in different populations?–results of an anatomical study and review of literature. *BMC neurology*, 6(1):1–9, 2006.

[332] K Ranil D De Silva, Rukmal Silva, Dhammika Amaratunga, WSL Gunasekera, and Rohan W Jayesekera. Types of the cerebral arterial circle (circle of willis) in a sri lankan population. *BMC neurology*, 11(1):1–8, 2011.