

# Reduced order modelling of intracranial aneurysm flow using proper orthogonal decomposition and neural networks

Michael MacRaidl<sup>1,2,3</sup>  | Ali Sarrami-Foroushani<sup>1,4</sup> | Toni Lassila<sup>5</sup> | Alejandro F. Frangi<sup>1,3,4,6,7</sup> 

<sup>1</sup>Centre for Computational Imaging and Modelling in Medicine (CIMIM), University of Manchester, Manchester, UK

<sup>2</sup>EPSRC Centre for Doctoral Training in Fluid Dynamics, University of Leeds, Leeds, UK

<sup>3</sup>Department of Computer Science, School of Engineering, University of Manchester, Manchester, UK

<sup>4</sup>School of Health Science, University of Manchester, Manchester, UK

<sup>5</sup>School of Computing, University of Leeds, Leeds, UK

<sup>6</sup>Department of Cardiovascular Sciences, KU Leuven, Leuven, Belgium

<sup>7</sup>Department of Electrical Engineering (ESAT), KU Leuven, Leuven, Belgium

## Correspondence

Michael MacRaidl and Alejandro F. Frangi, Centre for Computational Imaging and Modelling in Medicine (CIMIM), University of Manchester, Manchester, UK.  
Email: [michael.macraidl@manchester.ac.uk](mailto:michael.macraidl@manchester.ac.uk) and [alejandro.frangi@manchester.ac.uk](mailto:alejandro.frangi@manchester.ac.uk)

## Funding information

Royal Academy of Engineering under the RAEng Chair in Emerging Technologies, Grant/Award Number: INSILEX CiET1919/19; ERC Advanced Grant—UKRI Frontier Research Guarantee, Grant/Award Number: INSILICO EP/Y030494/1; European Commission to @neurIST and the @neurIST Consortium, Grant/Award Number: FP6-2004-IST-4-027703; European Commission via InSilc, Grant/Award Number: H2020-SC1-2017-CNECT-2-777119; Engineering and Physical Sciences Research Council (EPSRC) Centre for Doctoral Training in Fluid Dynamics, Grant/Award Number: P/L01615X/1

## Abstract

Reduced order modelling (ROMs) methods, such as proper orthogonal decomposition (POD), systematically reduce the dimensionality of high-fidelity computational models and potentially achieve large gains in execution speed. Machine learning (ML) using neural networks has been used to overcome limitations of traditional ROM techniques when applied to nonlinear problems, which has led to the recent development of reduced order models augmented by machine learning (ML-ROMs). However, the performance of ML-ROMs is yet to be widely evaluated in realistic applications and questions remain regarding the optimal design of ML-ROMs. In this study, we investigate the application of a non-intrusive parametric ML-ROM to a nonlinear, time-dependent fluid dynamics problem in a complex 3D geometry. We construct the ML-ROM using POD for dimensionality reduction and neural networks for interpolation of the ROM coefficients. We compare three different network designs in terms of approximation accuracy and performance. We test our ML-ROM on a flow problem in intracranial aneurysms, where flow variability effects are important when evaluating rupture risk and simulating treatment outcomes. The best-performing network design in our comparison used a two-stage POD reduction, a technique rarely used in previous studies. The best-performing ROM achieved mean test accuracies of 98.6% and 97.6% in the parent vessel and the aneurysm, respectively, while providing speed-up factors of the order  $10^5$ .

This is an open access article under the terms of the [Creative Commons Attribution](https://creativecommons.org/licenses/by/4.0/) License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

© 2024 The Author(s). *International Journal for Numerical Methods in Biomedical Engineering* published by John Wiley & Sons Ltd.

## KEYWORDS

haemodynamics, machine learning, reduced order modelling

## 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) use low-order representations of high-order model solutions that preserve essential input–output behaviour at the cost of some model accuracy and are commonly used to accelerate expensive computational models.<sup>1,2</sup> 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.<sup>3–9</sup>

Construction of ROMs for nonlinear PDEs typically involves: (i) solving the full order model (FOM) using standard numerical techniques at different parametric configurations to generate solution data (*snapshots*); (ii) extracting a low-order representation of the parametric solution manifold using dimensionality reduction techniques such as proper orthogonal decomposition (POD), dynamic mode decomposition (DMD), or auto-encoders<sup>6,10,11</sup>; and (iii) applying projection or interpolation methods in the low-order space to build a ROM.<sup>12</sup> 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. ML and neural networks have improved classical ROM methods by providing more efficient dimensionality reduction techniques,<sup>6,7,13,14</sup> approximating algebraic operators in projection-based ROMs for nonlinear problems,<sup>15,16</sup> improving the stability and long-term prediction accuracy of ROMs,<sup>5</sup> and providing high-dimensional interpolation techniques in interpolation-based ROMs.<sup>3,4,17</sup>

In this work, we apply a ML-ROM 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.<sup>18</sup> While aneurysms are often asymptomatic, their rupture causes subarachnoid haemorrhage, an event with high rates of mortality, morbidity, and disability. They also contribute directly to increasing healthcare costs.<sup>19</sup> 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.<sup>20</sup> Different physiological conditions have also been shown to affect aneurysm treatment performance.<sup>21</sup> 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 application with which to establish ROM performance.

Various ROM techniques have been applied to haemodynamic problems, including approaches based on POD,<sup>8,15,22–26</sup> DMD,<sup>27,28</sup> and reduced basis methods.<sup>29–31</sup> The approach we take is to use POD for dimensionality reduction and fully-connected neural networks (FCNNs) for interpolation of the ROM coefficients. We opt for non-intrusive interpolation-based ROMs for their ability to solve parameterised time-dependent problems while providing larger computational speed-ups than their projection-based counterparts.<sup>3,4,17</sup> We investigate three different network designs to improve on previous work on this type of ROM, including a design where we utilise a double-POD for further dimensionality reduction.<sup>32</sup> To generate training data, we use a validated computational fluid dynamics (CFD) model of blood flow in a 3D aneurysm derived from 3D rotational angiography (3DRA) images. We parameterise physiological flow variation by introducing three inlet flow-rate waveform scale factors that control flow magnitude, pulsatility, and heart rate. We sample the three-dimensional parameter space uniformly and use CFD data from the samples to train each ROM. For the best-performing network design and hyperparameters, we present results on the accuracy of the ROM relative to the FOM and on the acceleration factor and data storage reduction provided by the ROM.

The paper is outlined as follows. Section 2 details the methodology used in the CFD model (Section 2.1) and POD-Interpolation ROM (Section 2.2). Section 3 presents results from the multi-stage hyperparameter optimisation (Sections 3.1.2 and 3.1.3), the network design comparison (Section 3.2), the best-performing model (Section 3.3), and the acceleration and data storage reduction quantification (Section 3.4). Sections 4 and 5 provide discussion and concluding remarks, respectively.

## 2 | METHODOLOGY

### 2.1 | Computational simulation of intracranial aneurysm blood flow

#### 2.1.1 | Geometry

Figure 1 shows the ML-ROM methodology used in this study. Figure 2 shows the 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 References 33,34. The aneurysm maximum diameter and aspect ratio are 6.1 mm and 0.72, respectively. 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.

#### 2.1.2 | Computational fluid dynamics model

The momentum equations for incompressible and Newtonian fluid, the Navier–Stokes equations, were used to describe blood flow:

$$\rho \left( \frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} \right) = -\nabla p + \mu \nabla^2 \mathbf{u}, \quad \nabla \cdot \mathbf{u} = 0 \quad (1)$$

where  $\mathbf{u}$  and  $p$  are velocity and pressure, respectively. Blood was assumed to be a Newtonian fluid with constant density  $\rho = 1066 \text{ kg m}^{-3}$  and viscosity  $\mu = 0.0035 \text{ Pa s}$ . 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.

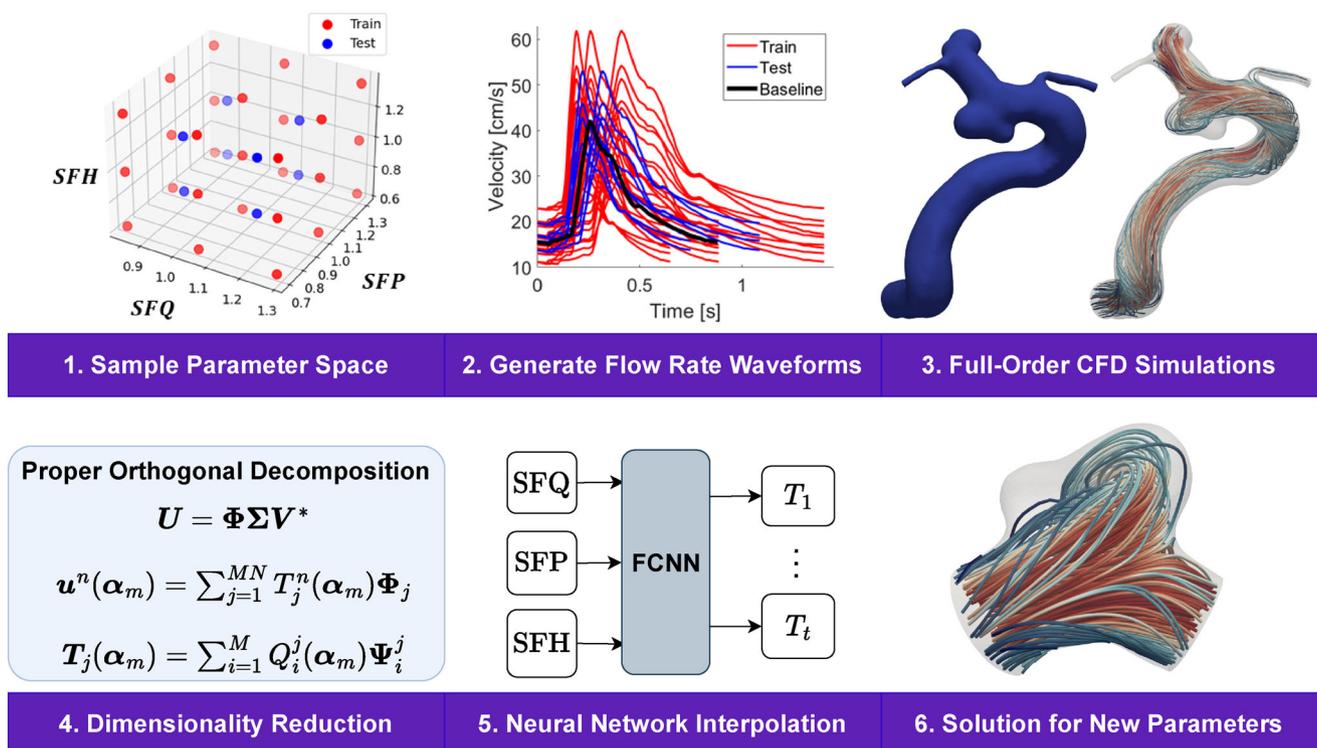
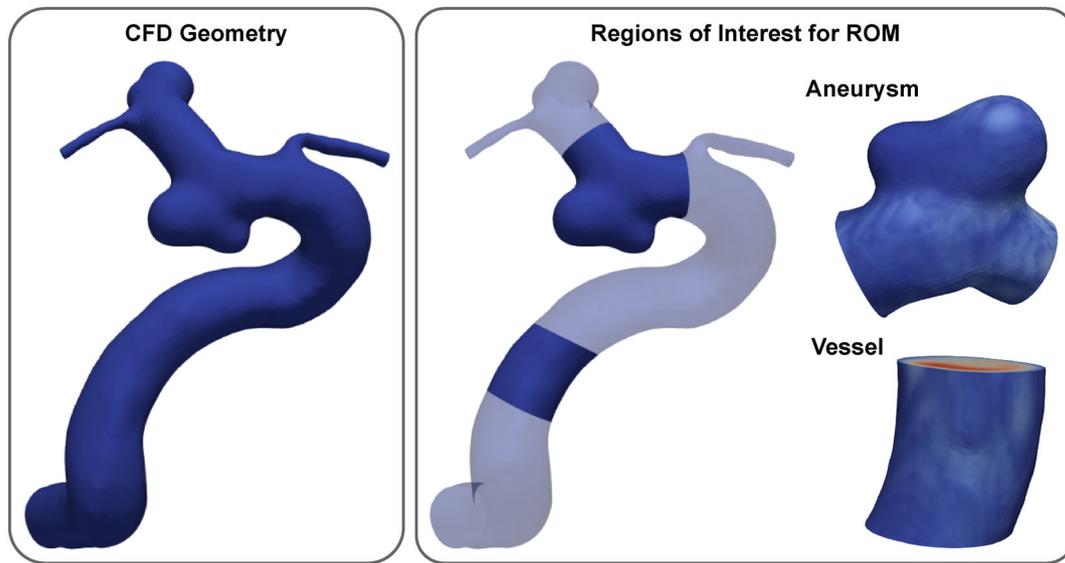


FIGURE 1 Our machine learning reduced order model (ML-ROM) methodology is based upon Proper Orthogonal Decomposition for dimensionality reduction and fully-connected neural networks for interpolation of the ROM coefficients.



**FIGURE 2** Internal carotid artery aneurysm geometry used in this study for computational fluid dynamics (CFD) simulations and clipped geometries used for the reduced order model (ROM).

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

### 2.1.3 | 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 (ICA) flow in 17 healthy young adults (age  $28 \pm 7$  years). Further details on the MGM are reported in References 20,35.

The FRW was parameterised in the form (following Geers et al.<sup>36</sup>)

$$Q(t) = aQ^0(ct) + b, \quad (2)$$

$$a = \underbrace{\frac{Q_{TA}}{Q_{TA}^0}}_{\text{SFQ}} \underbrace{\frac{PI}{PI^0}}_{\text{SFP}}, \quad b = Q_{TA} \left( 1 - \underbrace{\frac{PI}{PI^0}}_{\text{SFP}} \right), \quad c = \underbrace{\frac{HR}{HR^0}}_{\text{SFH}}, \quad (3)$$

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.,  $Q_{TA}^0$ ) represents the value of the given variable for the baseline waveform derived from the MGM. We defined three scale factors for flow magnitude (SFQ), pulsatility (SFP), and heart rate (SFH) through Equation (3), as

$$\text{SFQ} = \frac{Q_{TA}}{Q_{TA}^0}, \quad \text{SFP} = \frac{PI}{PI^0}, \quad \text{SFH} = \frac{HR}{HR^0}. \quad (4)$$

A range of configurations of SFQ, SFP, and SFH can be sampled within physiologically realistic bounds and then the FRW for each configuration is calculated through Equations (2) and (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.<sup>37</sup> measured left and right ICA flow rates in 17 young, healthy volunteers at rest. They found mean total (left + right) ICA flow of  $549 \text{ mL min}^{-1}$  with inter-patient standard deviation of  $\sigma_{inter} = 103 \text{ mL min}^{-1}$  and intra-patient standard deviation of  $\sigma_{intra} = 21 \text{ mL min}^{-1}$ . The minimum and maximum flow rates were found to be  $398 \text{ mL min}^{-1}$  and  $850 \text{ mL min}^{-1}$ , respectively. Normalising the minimum and maximum flow rates using the mean flow rate gives SFQ bounds of [0.72,1.54]. Using the intra-patient standard deviation and the mean flow rate gives bounds of [0.96,1.04]. Our ROM is only applicable to intra-patient flow rate variability, which is shown to be significantly smaller than the inter-patient flow rate bounds calculated previously. To induce additional variability into the snapshots, we selected bounds of [0.84,1.29], which is midway between the bounds for intra- and inter-patient variability.

### *Pulsatility index bounds*

PI is an easy to obtain (non-invasive) parameter that has been used to assess macrocirculation in highly prevalent medical conditions, such as hypertension, types 1 and 2 diabetes and thyroid disorders.<sup>38</sup> Schöning et al.<sup>39</sup> measured PI in 48 healthy adults and found a mean PI value of 1.08 and a standard deviation of 0.29 in the ICA. The standard deviation is  $\pm 31\%$  of the mean value, so we chose the upper and lower PI bounds to be  $\pm 31\%$  of the mean PI. Normalising by the mean PI value gives SFP bounds of [0.69,1.31].

### *Heart rate bounds*

Ford et al.<sup>37</sup> measured mean HR as  $68 \pm 8 \text{ bpm}$  (range 56–83) in 17 young, healthy volunteers at rest. Matsuo et al.<sup>40</sup> found that during moderate steady-state and incremental exercise for 12 young, healthy volunteers, mean HR did not exceed 120 bpm. We therefore select 56 as the minimum HR and 120 as the maximum, with 88 as the mean. Normalising the upper and lower HR bounds by the mean value gives SFH bounds of [0.63,1.36].

### *Sampling method*

We tested multiple sampling methods (uniform grid, Latin Hypercube, random) but found no significant difference in the results. We opted for a simple uniform grid sampling approach as a result. We used a  $3 \times 3 \times 3$  sampling grid with bounds for each parameter as stated previously. This generated a set of 27 simulations to be used as training data. We also generated a distinct set of eight parameter configurations from within the bounds of the parameter space to be used as a test set. These cases were selected to be a  $2 \times 2 \times 2$  grid with each parameter point spaced equidistantly from the nearest points in the  $3 \times 3 \times 3$  grid. Further details of the sampling method study can be found in Supplementary Section 2 in Data S1.

## 2.2 | POD-Interpolation ROM

This section describes the POD-NN method that we used in this work. Our approach and notation for POD is the same as used by Walton et al.,<sup>32</sup> and another description of this method can be found in Wang et al.<sup>12</sup> The benefits of this approach are: (i) it is non-intrusive and does not require accessing or changing the underlying CFD solver; (ii) it is relatively straightforward to implement and solve; (iii) it is faster to execute than projection-based ROMs; (iv) it can be applied to regions of interest instead of the full geometry; and (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.

### 2.2.1 | Proper orthogonal decomposition

We used the full-order CFD model described in Section 2.1 to generate velocity fields for a range of parameter configurations. We used  $M=27$  parameter configurations, with  $\alpha_1, \dots, \alpha_M$  denoting the configuration vectors and

$\alpha_m = (\text{SFQ}_m, \text{SFP}_m, \text{SFH}_m)$ . Each time-varying solution of the FOM for one parameter configuration is taken as a snapshot. Each snapshot then becomes a  $D \times N$  matrix, where  $D$  is the number of mesh nodes and  $N$  is the number of timesteps. After running the FOM simulations for all configurations, we generate a snapshot matrix  $\mathbf{U}$  of the form:

$$\mathbf{U} = \begin{bmatrix} u_1^1(\alpha_1) & \dots & u_1^N(\alpha_1) & \dots & u_1^N(\alpha_M) \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ u_D^1(\alpha_1) & \dots & u_D^N(\alpha_1) & \dots & u_D^N(\alpha_M) \end{bmatrix}. \quad (5)$$

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

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

$$\mathbf{U}_{D \times MN} = \mathbf{\Phi}_{D \times MN} \times \mathbf{\Sigma}_{MN \times MN} \times \mathbf{V}^*_{MN \times MN}. \quad (6)$$

The dimensions of each matrix are included in Equation (6), with  $MN = M \times N$ .  $\mathbf{\Phi}$  are the left singular vectors of  $\mathbf{U}$ ,  $\mathbf{\Sigma}$  are the singular values and  $\mathbf{V}^*$  are the right singular vectors. The columns of  $\mathbf{\Phi}$ , denoted  $\Phi_j$  with  $j = 1, \dots, MN$ , are the mutually orthogonal POD modes.  $\mathbf{\Sigma}$  is a diagonal matrix with each of the singular values, denoted  $\sigma_j$ , indicating the energetic contribution of each POD mode.

Each column of  $\mathbf{U}$  can be reconstructed using the left singular vectors as

$$\mathbf{u}^n(\alpha_m) = \sum_{j=1}^{MN} T_j^n(\alpha_m) \Phi_j, \quad (7)$$

where  $T^n(\alpha_m)$  are a set of parameter-dependent coefficients that can be considered as coordinates in the basis given by  $\Phi$ .<sup>32</sup> Each of the simulated parameter configurations will have a corresponding matrix  $T^n(\alpha_m)$  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.

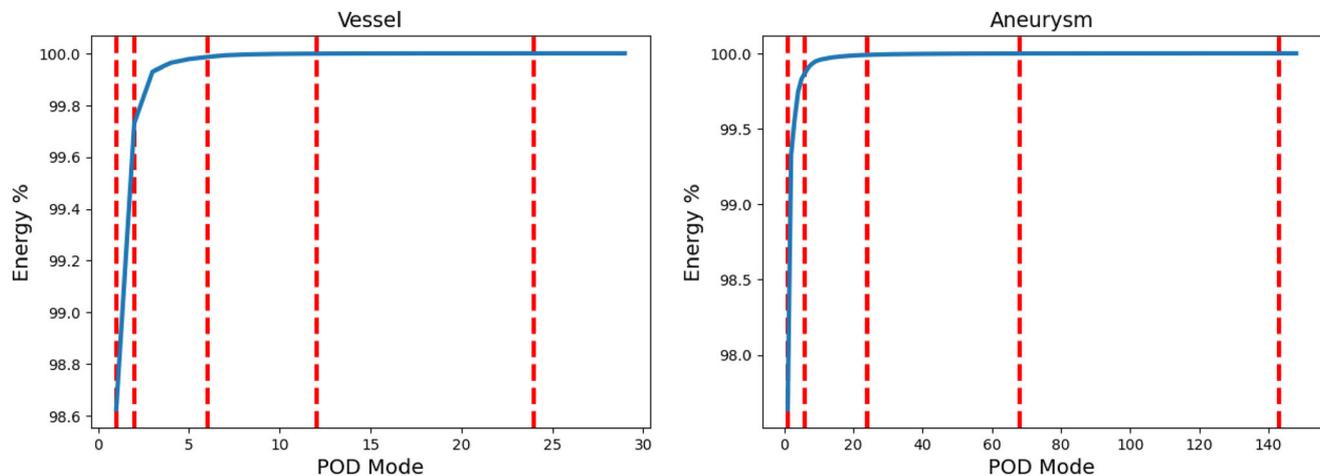
The system order can be reduced by considering the energy contained within each POD mode and then retaining  $r$  nodes up to a certain energy threshold. The sum of the squares of all of the singular values in  $\mathbf{\Sigma}$ , denoted by  $\sum_{j=1}^{M \times N} \sigma_j^2$  in Equation (8), gives 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  $E_{\text{frac}}$  to be retained and then finding  $r$  such that Equation (8) is satisfied.

$$E_{\text{frac}} \leq \frac{\sum_{j=1}^r \sigma_j^2}{\sum_{j=1}^{M \times N} \sigma_j^2}. \quad (8)$$

The snapshot matrix can then be truncated as follows:

$$\mathbf{U} \approx [\Phi_1 \dots \Phi_r] \begin{bmatrix} \sigma_1 & & & \\ & \ddots & & \\ & & \sigma_r & \\ & & & \ddots \end{bmatrix} \begin{bmatrix} \mathbf{v}_1^* \\ \vdots \\ \mathbf{v}_r^* \end{bmatrix}. \quad (9)$$

Figure 3 demonstrates the vast truncation that can be achieved in the vessel and aneurysm models.



**FIGURE 3** 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.

## 2.2.2 | Low-dimensional representation

Now that the order of the system has been reduced, Equation (7) can be modified to

$$\mathbf{u}^n(\boldsymbol{\alpha}_m) = \sum_{j=1}^r T_j^n(\boldsymbol{\alpha}_m) \Phi_j, \quad (10)$$

with the sum up to  $j=r$  rather than  $j=MN$ . Before interpolating between the  $T_j^n$  coefficients, we calculate them by exploiting the orthogonality of the POD basis vectors (Equation 11).

$$\Phi_i^T \Phi_j = \begin{cases} 1, & \text{if } i=j \\ 0, & \text{otherwise.} \end{cases} \quad (11)$$

For each parameter configuration in turn, we take an inner product of the transpose of each  $\Phi$  mode with both sides of Equation (10). To exploit orthogonality, each  $\Phi$  mode is looked at in turn. Taking only the first  $\Phi$  mode ( $j=1$ ), Equation (10) is now simply

$$\mathbf{u}^n(\boldsymbol{\alpha}_m) = T_1^n(\boldsymbol{\alpha}_m) \Phi_1. \quad (12)$$

Multiplying both sides of (12) by the transpose of  $\Phi_1$  gives

$$\Phi_1^T \mathbf{u}^n(\boldsymbol{\alpha}_m) = \Phi_1^T T_1^n(\boldsymbol{\alpha}_m) \Phi_1, \quad (13)$$

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

$$\Phi_1^T \mathbf{u}^n(\boldsymbol{\alpha}_m) = T_1^n(\boldsymbol{\alpha}_m) \underbrace{\Phi_1^T \Phi_1}_{=1} \quad (14)$$

We are left with an equation for  $T_1^n$ :

$$\mathbf{T}_1^n(\boldsymbol{\alpha}_m) = \underset{1 \times N}{\Phi_1^T} \underset{1 \times D}{\mathbf{u}^n} \underset{D \times N}{(\boldsymbol{\alpha}_m)}. \quad (15)$$

Each coefficient in  $T_1^n$  gives the weighting of the first  $\Phi$  mode at timestep  $n$ . A vector of  $T_1^n$  coefficients can be found for each parameter configuration for the first  $\Phi$  mode. These coefficients are arranged into a second snapshot matrix  $\mathbf{S}_j$ , as by Walton et al.<sup>32</sup>:

$$\mathbf{S}_j = \begin{bmatrix} T_j^1(\boldsymbol{\alpha}_1) & \dots & T_j^1(\boldsymbol{\alpha}_M) \\ \vdots & \dots & \vdots \\ T_j^N(\boldsymbol{\alpha}_1) & \dots & T_j^N(\boldsymbol{\alpha}_M) \end{bmatrix}. \quad (16)$$

Each column of  $\mathbf{S}_j$  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  $\Phi_j$ . A distinct  $\mathbf{S}_j$  is found for each  $\Phi$  mode. The goal is now to interpolate between these  $\mathbf{S}_j$  matrices to find the representation for a new set of parameters denoted  $\boldsymbol{\alpha}_k$ .

Performing a second SVD on  $\mathbf{S}_j$  gives a secondary set of POD modes, denoted  $\Psi_i^j$ . 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, each column of  $\mathbf{S}_j$  can be reconstructed using

$$\mathbf{T}_j(\boldsymbol{\alpha}_m) = \sum_{i=1}^{r_Q} Q_i^j(\boldsymbol{\alpha}_m) \Psi_i^j, \quad (17)$$

where  $r_Q$  is the number of secondary POD modes retained following truncation. The purpose of this secondary SVD is to split  $\mathbf{T}_j(\boldsymbol{\alpha}_m)$  into parameter-dependent terms,  $Q_i^j(\boldsymbol{\alpha}_m)$ , and parameter-independent terms,  $\Psi_i^j$ . This means that we only need to interpolate between the  $Q_i^j$  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

$$Q_i^j(\boldsymbol{\alpha}_m) = \underset{M \times M}{\Psi_i^{jT}} \underset{M \times N}{\mathbf{S}_j} \underset{N \times M}{(\boldsymbol{\alpha}_m)}. \quad (18)$$

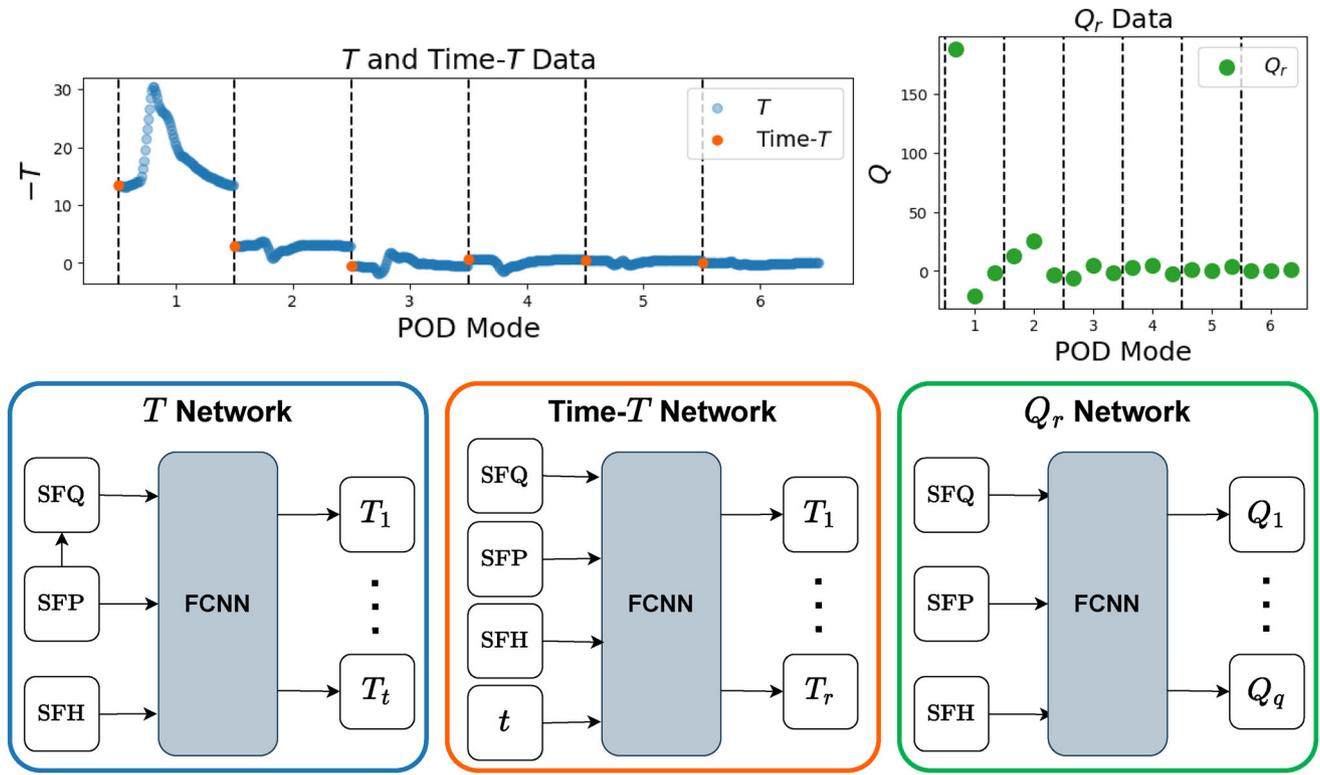
Each column of  $Q_i^j$  gives the weighting coefficient for the corresponding  $\Psi_i^j$  mode. We then interpolate the  $Q_i^j$  values to find  $Q$  values for a new parameter configuration.

### 2.2.3 | Neural network interpolation

Neural networks excel at approximating nonlinear functions with high-dimensional input–output relations. In our case, the neural network inputs are the scale factors (and potentially time) and the outputs are the corresponding ROM coefficients (either  $T$  or  $Q$  values). We constructed three different neural networks for interpolating the ROM coefficients, as shown in Figure 4.

The number of neurons in the output layer of each network in Figure 4 is different. The  $T$  network has  $r \times N$  outputs and the Time- $T$  network has  $r$  outputs, where  $r$  is the number of truncated POD modes and  $N$  is the number of timesteps. The  $Q_r$  network has  $r \times r_Q$  outputs, where  $r_Q$  is the number of modes retained in the second POD truncation. We performed an in-depth hyperparameter study for the network parameters, details and results for which can be found in Sections 3.1.2 and 3.1.3.

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  $\mathbf{T}$  values can then be found by multiplying by the stored  $\Psi$  modes:



**FIGURE 4** 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 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.

$$\mathbf{T}_j(\boldsymbol{\alpha}_k) = \sum_{i=1}^{r_Q} Q_i^j(\boldsymbol{\alpha}_k) \Psi_i^j. \quad (19)$$

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

$$\mathbf{u}^n(\boldsymbol{\alpha}_k) = \sum_{j=1}^r T_j^n(\boldsymbol{\alpha}_k) \Phi_j. \quad (20)$$

This model allows us to quickly evaluate 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 our ML-ROM, we present Algorithms 1 and 2 for the offline and online stages of the ROM, respectively.

### 3 | RESULTS

The results section is outlined as follows. First, we present the hyperparameter studies performed on each network design shown in Figure 4. Second, for the best-performing network design, we present results quantifying the different sources of error in the ROM, namely the errors due to POD truncation and network generalisation. Finally, for the best-performing sampling regime and network design, we demonstrate the accuracy, acceleration and data storage reduction the POD-NN ROM offers relative to the high-fidelity CFD model.

**Algorithm 1 POD-NN Offline Phase**

- 1: Use FOM to simulate a number of parameter configurations  $\alpha_m$ .
- 2: Construct snapshot matrix  $\mathbf{U}$ .
- 3: Economic SVD of  $\mathbf{U} = \Phi \Sigma \mathbf{V}^*$  to generate POD modes  $\Phi_j$ .
- 4: Retain first  $r$  POD modes based upon energy criteria:  $E_{\text{frac}} \leq \frac{\sum_{j=1}^r \sigma_j^2}{\sum_{j=1}^M \sigma_j^2}$ .
- 5: Take inner product of snapshot matrix and each POD mode to generate time coefficients:  $\mathbf{u}^n(\alpha_m) = \sum_{j=1}^r T_j^n(\alpha_m) \Phi_j$ .
- 6: Construct time coefficient matrix  $\mathbf{S}_j$  for each POD mode.
- 7: SVD of  $\mathbf{S}_j$  to generate  $Q_i^j(\alpha_m)$  and  $\Psi_i^j$ .
- 8: Save the retained  $\Phi_r$  and  $\Psi_i^j$  modes and low-dimensional representations  $Q_i^j(\alpha_m)$  of each full order simulation.
- 9: 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  $\alpha_k$ , evaluate  $T_j^n(\alpha_k)$  or  $Q_i^j(\alpha_k)$  using the trained FCNNs.
- 2: For  $Q_r$  network, calculate  $T$  coefficients for new parameter configuration:  $T_j^n(\alpha_k) = \sum_{i=1}^M Q_i^j(\alpha_k) \Psi_i^j$ .
- 3: Calculate solution for new parameter configuration:  $\mathbf{u}^n(\alpha_k) = \sum_{j=1}^r T_j^n(\alpha_k) \Phi_j$ .

### 3.1 | Hyperparameter studies

#### 3.1.1 | 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. We performed preliminary tests 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 Supplementary Section 1 in Data S1. From these preliminary tests, we found good network performance 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, we used Adam optimisation with a learning rate of  $10^{-3}$ <sup>41</sup> and trained until either 1000 epochs or until the loss plateaued such that the mean loss over the 10 most recent epochs is within  $10^{-5}$  of the mean loss over the 10 preceding epochs. We found that 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 we used a batch size of 10 in this case.

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

#### 3.1.2 | Hyperparameter study: Stage 1

In the first hyperparameter study, we varied the parameters through the values specified in Table 1. Various hidden layer sizes were tested, including a small network with one hidden layer and 10 neurons, a wide network with three hidden layers and 1000 neurons per layer, and a deep network with six hidden layers and 10 neurons per layer. For

TABLE 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 (vessel)	1, 2, 6, 12, 24
No. of POD modes (aneurysm)	1, 6, 24, 68, 143

each network, we tested single-component and multi-component loss functions. The single component loss was simply the prediction error between the ROM coefficients ( $T_r$ ,  $T_r$ , or  $Q_r$ ) and their ground truth values. 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 10 timesteps, so for this network the multi-component loss only calculated the solution field error for these 10 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, 2, 6, 12, 24} and {1, 6, 24, 68, 143} POD modes for the vessel and aneurysm models, respectively. The cumulative energy stored in the POD modes is shown in Figure 3. Note that in the first hyperparameter study, we retained three second POD modes for all  $Q_r$  networks. This parameter is investigated in more detail in the second hyperparameter study.

From the first hyperparameter study results (Figure S1 and Tables S4 and S5), we draw the following conclusions: (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 did produce a smaller minimum error in the  $Q_r$  network. (iii) Accuracy is greatest for approximately 99.9%–99.99% energy retention.

### 3.1.3 | Hyperparameter study: Stage 2

Based on the conclusions from the first hyperparameter study, we restricted the hyperparameter ranges in the second study to those presented in Table 2. For the  $Q_r$  network, we now investigate an additional hyperparameter for the number of second POD modes retained.

Following the second hyperparameter study, results for which can be seen in Tables S6 and S7, we identified the optimal network configurations outlined in Table 3.

## 3.2 | 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. 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. Snapshot sampling error typically affects ROM accuracy, but we did not find our ROM performance to be sensitive to the choice of sampling algorithm (see Supplementary Section 2 in Data S1 for details).

### 3.2.1 | POD truncation and network generalisation errors

For the optimal network designs presented in Table 3, we evaluated the model performance on the test and training sets of data for vessel and aneurysm regions. We are able to quantify the contributions of the POD truncation and network

TABLE 2 Second hyperparameter study details.

Hyperparameter	Values
<i>T</i> network	
Hidden layers	[10], [10, 100]
Loss construction	Multi-component
Number of POD modes (vessel)	4, 8, 10
Number of POD modes (aneurysm)	4, 8, 10, 12, 18
Time- <i>T</i> network	
Hidden layers	[10, 100, 1000], [1000, 1000, 1000], [1000]
Loss construction	Single-component
Number of POD modes (vessel)	2, 4, 6, 8, 10, 12
Number of POD modes (aneurysm)	4, 6, 8, 12, 18
<i>Q<sub>r</sub></i> network	
Hidden layers	[10], [10, 100]
Loss construction	Multi-component
Number of POD modes (vessel)	4, 8, 10, 12, 16
Number of POD modes (aneurysm)	12, 16, 20, 28, 32
Number of second POD modes	1, 2, 3, 4, 5, 7, 10

TABLE 3 Optimal hyperparameters for each network design in each region of interest.

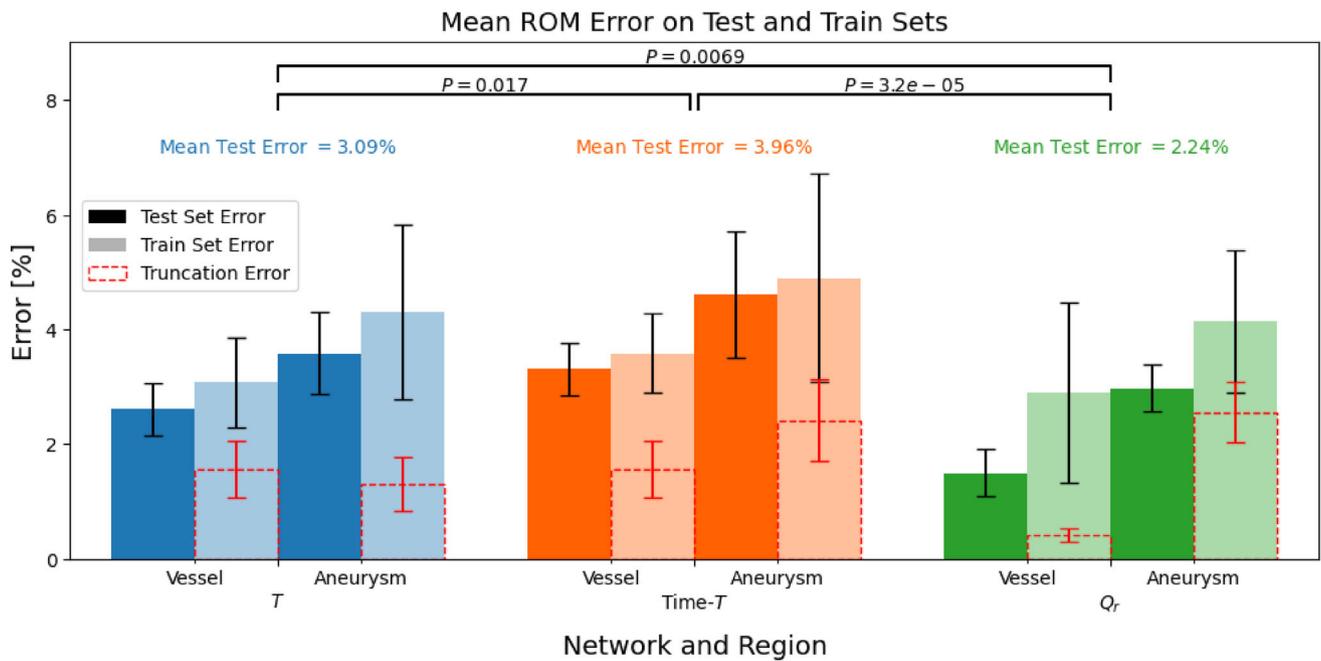
Model	Network	Network size	Loss construction	No. of POD modes	No. of second POD modes
Vessel	<i>T</i>	[10, 100]	Multi-component	4	NA
	Time- <i>T</i>	[1000, 1000, 1000]	Single-component	4	NA
	<i>Q<sub>r</sub></i>	[10]	Multi-component	12	5
Aneurysm	<i>T</i>	[10]	Multi-component	10	NA
	Time- <i>T</i>	[10, 100, 1000]	Single-component	6	NA
	<i>Q<sub>r</sub></i>	[10]	Multi-component	16	3

generalisation errors 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. We then calculated the network generalisation error as the difference between the ROM error using the network predicted and ground truth *T* and *Q* coefficients.

Figure 5 shows the ROM errors for each network design for the testing and training data sets for each geometry. The *Q<sub>r</sub>* 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<sub>r</sub>* network gives the most accurate results, we trained networks with different values of POD truncation energy in order to see how the weighting between POD truncation and network generalisation error varies. We used the optimal network parameters (Table 3) 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<sub>r</sub>* network, we used the same number of second POD modes ( $r_Q = 3$ ) so that the variation in truncation error with additional POD modes would not be affected by this parameter for the different regions.

Figure 6 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



**FIGURE 5** Mean ROM testing errors on the test and train sets of 8 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. The mean test error values on the test set for each network and model are as follows:  $T$  (vessel) = 2.61%,  $T$  (aneurysm) = 3.58%, Time- $T$  (vessel) = 3.31%, Time- $T$  (aneurysm) = 4.61%  $Q_r$  (vessel) = 1.50%,  $Q_r$  (aneurysm) = 2.97%.  $p$  values are calculated using a  $T$ -test for the means of independent samples.

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.

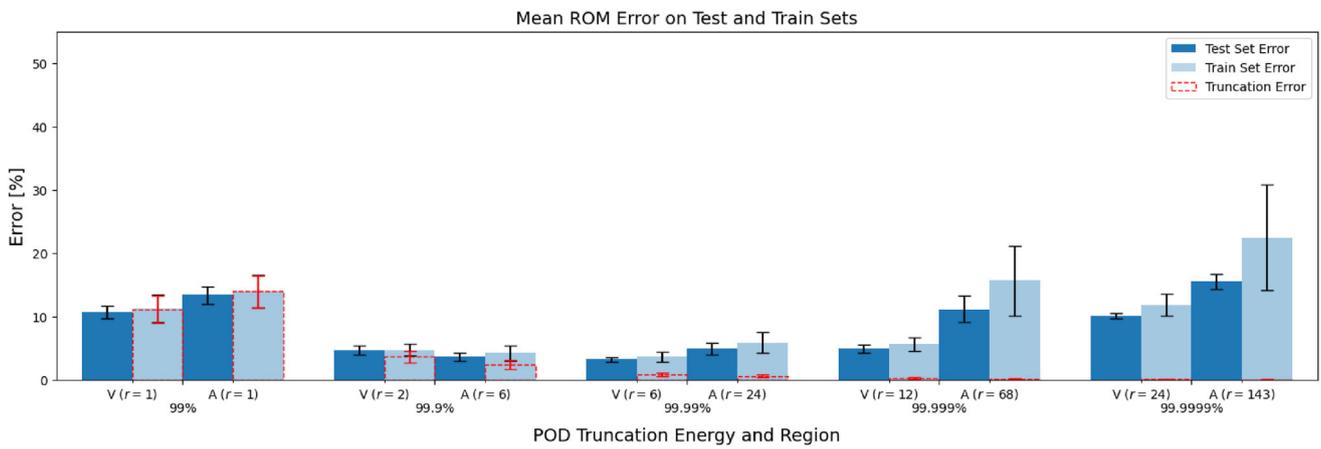
### 3.3 | Best-performing model

To further demonstrate the quantitative accuracy of the ROM predictions and show the qualitative agreement between ROM and FOM, we present visualisations and additional figures that were created using the optimal  $Q_r$  network, which gave the best-performing ROM. We only present these extra results for the aneurysm model as the flow is more complex in this case and for the sake of brevity. Figure 7 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:

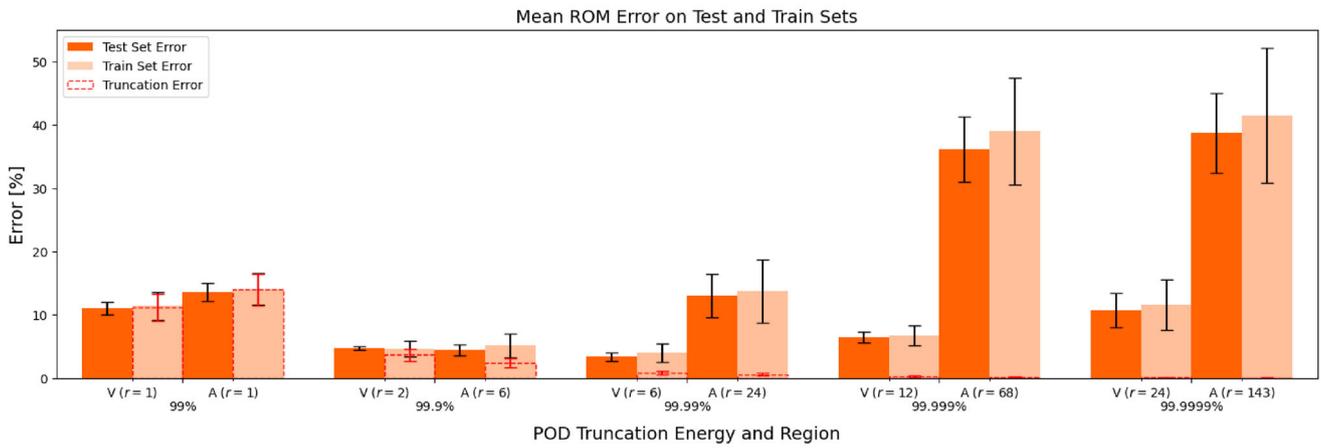
$$\text{Relative Error} = \frac{U_{ROM} - U_{FOM}}{\bar{U}_{FOM}} \times 100\%, \quad (21)$$

where  $\bar{U}_{FOM}$  is the space-and-time-averaged FOM velocity magnitude. The largest relative errors occur at systole ( $\sim \pm 30\%$ ).

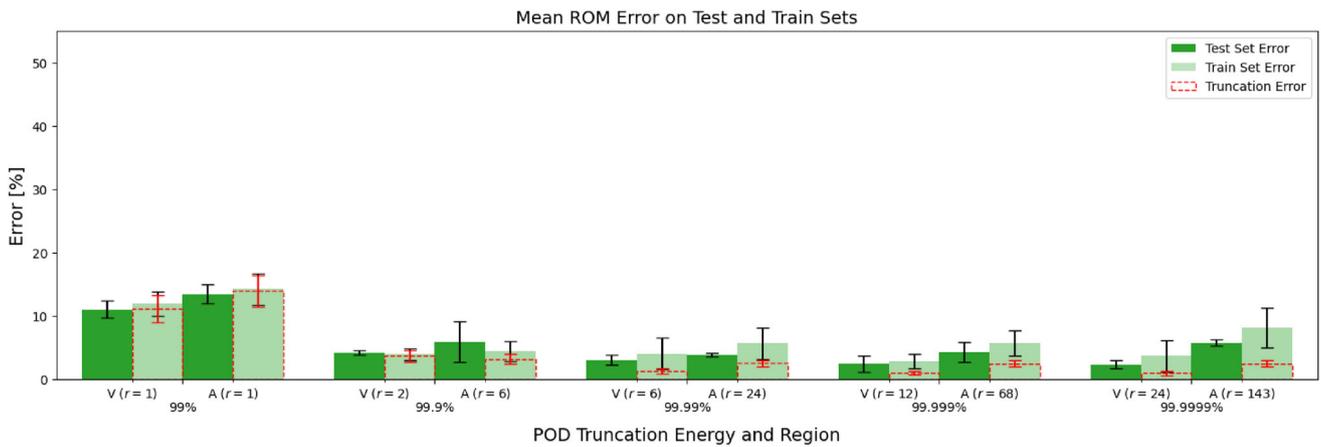
Figure 7 demonstrates the qualitative agreement between ROM and FOM at systole and diastole. Figure 8 shows Bland-Altman plots in the aneurysm region for systolic, diastolic, and time-averaged ROM and CFD velocity magnitude. Figure 8 also shows spatially-averaged velocity magnitude field waveforms for ROM and FOM, the standard deviation across the spatial points, and the absolute relative error between the two waveforms. Figure 8 demonstrates the agreement between the time-averaged and spatially-averaged fields. The largest discrepancy in velocity magnitude occurs in the post-systolic period of the cardiac cycle, where the ROM slightly smooths the gradients of the CFD waveform.



(A)  $T$  network.



(B) Time- $T$  network.



(C)  $Q_r$  network.

FIGURE 6 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.

### 3.4 | Acceleration and data storage reduction quantification

#### 3.4.1 | Acceleration

We quantified ROM speed 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 s, which

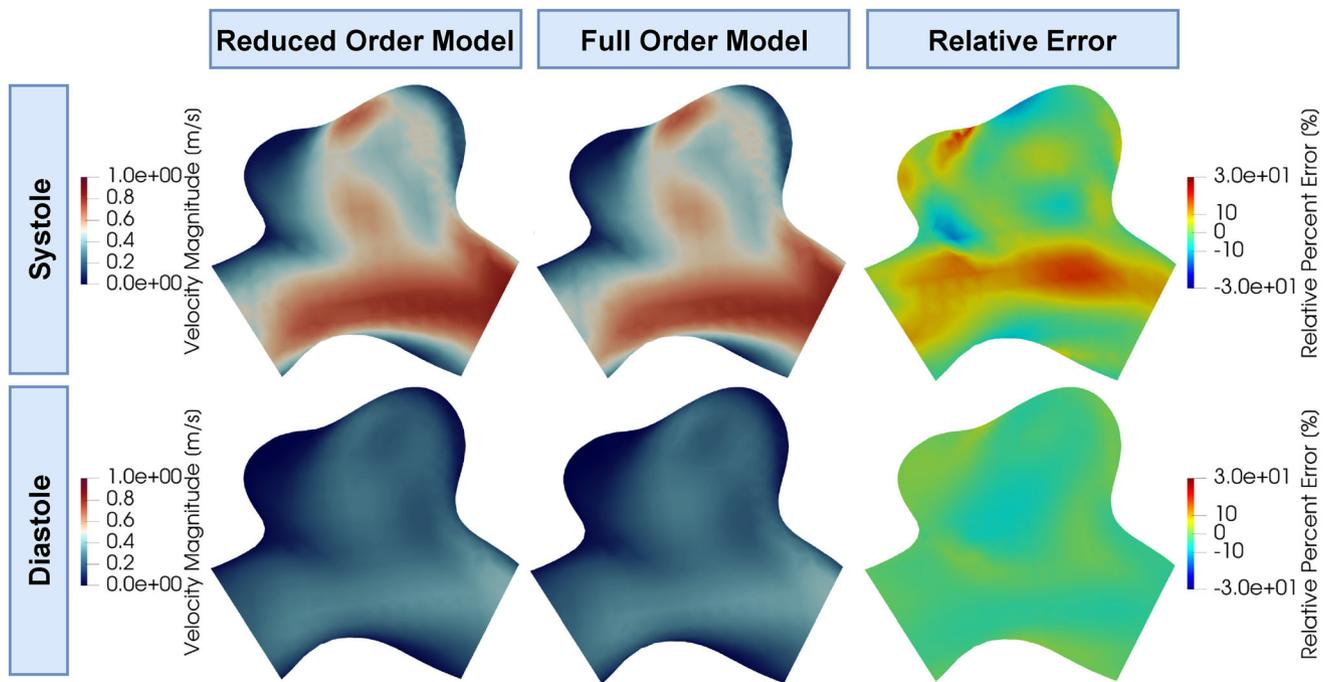


FIGURE 7 Visualisations of systolic and diastolic ROM and CFD velocity magnitude fields and the relative error between them in a 2D slice of the aneurysm geometry for a median error test case using the  $Q_r$  network.

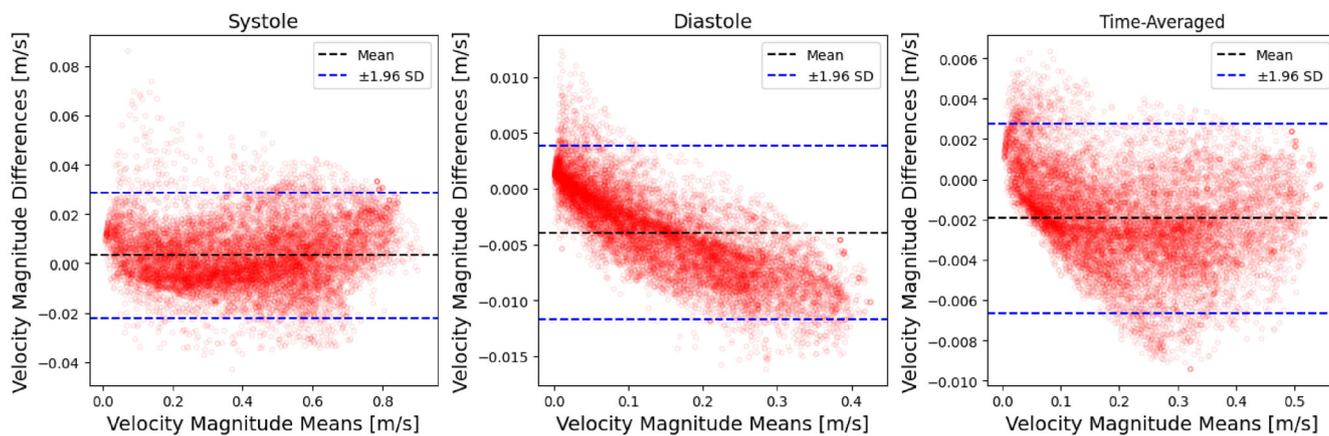
highlights that the POD-NN ROM methods can provide fast solution inference. To calculate acceleration, we compared the mean ROM run-time with the time it took to run the FOM locally and on a high-performance computing (HPC) cluster. We performed one evaluation of the FOM locally using one CPU. This simulation took 1 h 39 min and 32 s. We used 16 cores for the HPC simulations and the mean run-time across the eight test cases was 25 min and 51 s. We then calculated the acceleration of the various ROM designs for each geometry relative to these FOM run-times. The results are presented in Table 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.

### 3.4.2 | Data storage reduction

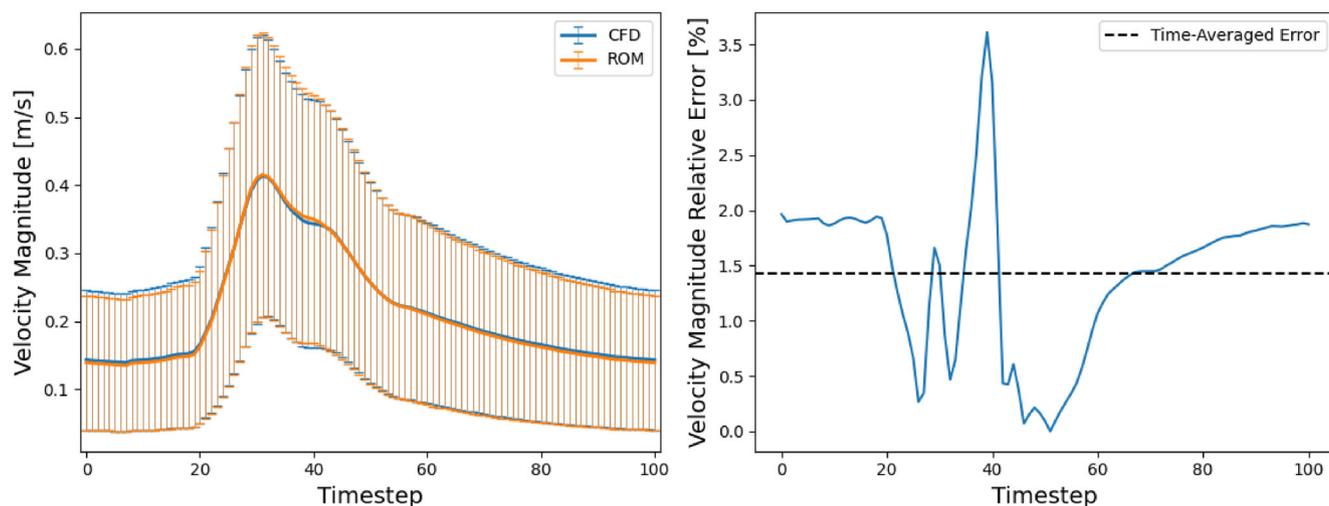
Retaining all POD modes in the ROM equates to performing direct interpolation between the high-order solution snapshots. When considering data storage requirements, a comparison can therefore be made between the amount of data that must be stored in order to construct a POD-Interpolation ROM and the 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. The amount of data stored by the reduced order models compared to that required for DSI is approximately 120 times less for the aneurysm model. This reduction factor would be further increased if the number of POD modes was decreased (e.g., the reduction factor for 5 POD modes would be roughly 380).

## 4 | DISCUSSION

We proposed a non-intrusive parametric reduced order model (ROM) that uses POD for dimensionality reduction and neural networks (NN) for interpolation of the ROM coefficients. We applied our POD-NN ROM to intracranial aneurysm fluid dynamics with physiological variability characterised by inflow parameters for flow magnitude (SFQ),



(A) Bland-Altman plots for systolic, diastolic and time-averaged velocity magnitude. Differences are CFD velocity magnitude fields subtracted from the ROM velocity magnitude fields.



(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 percentage error (equation 21).

FIGURE 8 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.

TABLE 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 (s)	Acceleration factor* (relative to 1 CPU)	Acceleration factor* (relative to 16 CPUs)
Vessel	$T$	0.012	$5.1 \times 10^5$	$1.3 \times 10^5$
	Time- $T$	0.031	$1.9 \times 10^5$	$5.0 \times 10^4$
	$Q_r$	0.022	$2.7 \times 10^5$	$7.1 \times 10^4$
Aneurysm	$T$	0.043	$1.4 \times 10^5$	$3.6 \times 10^4$
	Time- $T$	0.055	$1.1 \times 10^5$	$2.8 \times 10^4$
	$Q_r$	0.062	$9.7 \times 10^4$	$2.5 \times 10^4$

\*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.

TABLE 5 Number of variables involved in POD-Interpolation and direct solution interpolation.

Method	Total size of stored data	Aneurysm model <sup>a</sup>
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

Note:  $D$ , 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.

<sup>a</sup>For the aneurysm model,  $D = 6334$ ,  $N = 101$ , and  $M = 27$ . For the calculation, we chose  $r = 16$  and  $r_Q = 3$ , which are the optimal parameters for the  $Q_r$  network.

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.<sup>4,8,9</sup> We investigated this approach (referred to as the *Time- $T$  network*) and two other approaches, 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 ML.<sup>32</sup>

Our 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 6 shows that the 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 our results demonstrated that the  $Q_r$  network performs best, a more extensive and 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 ROM accuracy and the resources put into searching for the optimal network. If aiming to streamline the network optimisation process, then our results demonstrate that using a two-stage POD reduction can provide accurate results with a more straightforward network optimisation procedure. Given that simplifying the ROM data appears to improve the network robustness, it may be that using more advanced ML-based dimensionality reduction tools such as autoencoders could further simplify the training data and further improve ROM performance or simplify training.<sup>42</sup>

We found that each ROM we investigated provides a speed-up of order  $10^5$  relative to the full-order CFD simulation performed on one CPU and of order  $10^4$  relative to the CFD simulation on 16 CPUs. 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 a vascular flow context, the 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.

Our ROM could be useful in scenarios such as inverse modelling (e.g., to identify boundary conditions that produce outputs matching experimental data<sup>8</sup>), 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). The ROM could also be suitable for real-time flow monitoring in various applications due to its speed. In its current design, however, our ROM is not suitable for problems where capturing geometric variability is essential. Two examples of this in a vascular flow context are real-time surgical feedback where the geometry changes dynamically and 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 as additional network inputs in POD-NN ROMs but only for relatively simple geometries.<sup>3</sup> Another possible approach to overcome this challenge is to use domain decomposition ROMs that partition an unseen geometry into sub-geometries that are similar to the geometries for which snapshots were previously calculated.<sup>43,44</sup> This approach has been applied to flow over urban landscapes and pipe flow problems, but could potentially be applied to vascular flow problems too. In the wider ROM field, higher dimensional parameterisations have been used for vascular flow applications, but these are often limited in their generality.<sup>15</sup> Constructing simulation acceleration methods that can accurately evaluate flow solutions in previously unseen geometries remains a key challenge, particularly in vascular flow modelling problems where geometries can vary greatly between patients and in pathological vessels.<sup>45</sup>

## 5 | CONCLUSION

We have successfully developed and applied a POD-NN ROM to the physiological variation in blood flow in intracranial aneurysms. We found that a two-stage POD reduction improves model performance compared with the widely used one-stage POD approach. Future work into POD-NN ROMs could include: (i) adding geometric input parameters into the model; (ii) introducing smart sampling techniques to handle higher dimensional input parameter spaces; (iii) including a device in the vascular model and determining whether the ROM is suitable for the multi-scale nature of such problems; and (iv) investigating the use of ML-based dimensionality reduction algorithms in place of POD.

### AUTHOR CONTRIBUTIONS

**Michael MacRaid:** Conceptualisation; investigation; methodology; software; writing—original draft; writing—review and editing. **Ali Sarrami-Foroushani:** Conceptualisation; investigation; methodology; supervision; writing—review and editing. **Toni Lassila:** Conceptualisation; investigation; methodology; supervision; writing—review and editing. **Alejandro Frangi:** Conceptualisation; supervision; writing—review and editing.

### ACKNOWLEDGEMENTS

A.F.F. acknowledges support from the Royal Academy of Engineering under the RAEng Chair in Emerging Technologies (INSILEX CiET1919/19) and ERC Advanced Grant—UKRI Frontier Research Guarantee (INSILICO EP/Y030494/1) schemes. A.F.F. acknowledges seminal funding from the European Commission to @neurIST (FP6-2004-IST-4-027703) and the @neurIST Consortium. A.S-F., T.L., and A.F.F. were funded by the European Commission via InSilc (H2020-SC1-2017-CNECT-2-777119). M.M. was supported by the Engineering and Physical Sciences Research Council (EPSRC) Centre for Doctoral Training in Fluid Dynamics (P/L01615X/1). The information contained herein reflects only the authors' view, and none of the funders or the European Commission are responsible for any use that may be made of it.

### CONFLICT OF INTEREST STATEMENT

The authors declare no conflict of interest.

### ORCID

Michael MacRaid  <https://orcid.org/0009-0001-3307-743X>

Alejandro F. Frangi  <https://orcid.org/0000-0002-2675-528X>

### REFERENCES

1. Benner P, Gugercin S, Willcox K. A survey of projection-based model reduction methods for parametric dynamical systems. *SIAM Rev.* 2015;57(4):483-531.
2. Quarteroni A, Rozza G. *Reduced Order Methods for Modeling and Computational Reduction*. Vol 9. Springer; 2014.

3. Hesthaven JS, Ubbiali S. Non-intrusive reduced order modeling of nonlinear problems using neural networks. *J Comput Phys*. 2018;363:55-78.
4. Wang Q, Hesthaven JS, Ray D. Non-intrusive reduced order modeling of unsteady flows using artificial neural networks with application to a combustion problem. *J Comput Phys*. 2019;384:289-307.
5. Wang Q, Ripamonti N, Hesthaven JS. Recurrent neural network closure of parametric pod-galerkin reduced-order models based on the mori-zwanzig formalism. *J Comput Phys*. 2020;410:109402.
6. Maulik R, Lusch B, Balaprakash P. Reduced-order modeling of advection-dominated systems with recurrent neural networks and convolutional autoencoders. *Phys Fluids*. 2021;33(3):037106.
7. Pant P, Doshi R, Bahl P, Farimani AB. Deep learning for reduced order modelling and efficient temporal evolution of fluid simulations. *Phys Fluids*. 2021;33(10):107101.
8. Balzotti C, Siena P, Girfoglio M, Quaini A, Rozza G. A data-driven reduced order method for parametric optimal blood flow control: application to coronary bypass graft. *arXiv*. 2022.
9. Hajisharifi A, Romanò F, Girfoglio M, Beccari A, Bonanni D, Rozza G. A non-intrusive data-driven reduced order model for parametrized cfd-dem numerical simulations. *J Comput Phys*. 2023;491:112355.
10. Lassila T, Manzoni A, Quarteroni A, Rozza G. Model order reduction in fluid dynamics: challenges and perspectives. *Reduced Order Methods for Modeling and Computational Reduction*. Springer; 2014:235-273.
11. Schmid PJ. Dynamic mode decomposition and its variants. *Annu Rev Fluid Mech*. 2022;54:225-254.
12. Wang Y, Bo Y, Cao Z, Zou W, Guojun Y. A comparative study of pod interpolation and pod projection methods for fast and accurate prediction of heat transfer problems. *Int J Heat Mass Transf*. 2012;55(17-18):4827-4836.
13. Fresca S, Manzoni A. Pod-dl-rom: enhancing deep learning-based reduced order models for nonlinear parametrized pdes by proper orthogonal decomposition. *Comput Methods Appl Mech Eng*. 2022;388:114181.
14. Gruber A, Gunzburger M, Lili J, Wang Z. A comparison of neural network architectures for data-driven reduced-order modeling. *Comput Methods Appl Mech Eng*. 2022;393:114764.
15. Buoso S, Manzoni A, Alkadhi H, Plass A, Quarteroni A, Kurtcuoglu V. Reduced-order modeling of blood flow for noninvasive functional evaluation of coronary artery disease. *Biomech Model Mechanobiol*. 2019;18:1-15.
16. Gao H, Wang J-X, Zahr MJ. Non-intrusive model reduction of large-scale, nonlinear dynamical systems using deep learning. *Phys D*. 2020;412:132614.
17. San O, Maulik R, Ahmed M. An artificial neural network framework for reduced order modeling of transient flows. *Commun Nonlinear Sci Numer Simul*. 2019;77:271-287.
18. Rinkel GJE, Djibuti M, Algra A, Van Gijn J. Prevalence and risk of rupture of intracranial aneurysms: a systematic review. *Stroke*. 1998;29(1):251-256.
19. Meng H, Tutino VM, Xiang J, Siddiqui AJAJON. High wss or low wss? Complex interactions of hemodynamics with intracranial aneurysm initiation, growth, and rupture: toward a unifying hypothesis. *Am J Neuroradiol*. 2014;35(7):1254-1262.
20. Sarrami-Foroushani A, Lassila T, Gooya A, Geers AJ, Frangi AF. Uncertainty quantification of wall shear stress in intracranial aneurysms using a data-driven statistical model of systemic blood flow variability. *J Biomech*. 2016;49(16):3815-3823.
21. Sarrami-Foroushani A, Lassila T, MacRaid M, et al. In-silico trial of intracranial flow diverters replicates and expands insights from conventional clinical trials. *Nat Commun*. 2021;12(1):3861.
22. Grinberg L, Yakhot A, Karniadakis GE. Analyzing transient turbulence in a stenosed carotid artery by proper orthogonal decomposition. *Ann Biomed Eng*. 2009;37(11):2200-2217.
23. Guibert R, Mcleod K, Caiazzo A, et al. Group-wise construction of reduced models for understanding and characterization of pulmonary blood flows from medical images. *Med Image Anal*. 2014;18(1):63-82.
24. Bertagna L, Veneziani A. A model reduction approach for the variational estimation of vascular compliance by solving an inverse fluid-structure interaction problem. *Inverse Problems*. 2014;30(5):055006.
25. Ballarin F, Faggiano E, Ippolito S, et al. Fast simulations of patient-specific haemodynamics of coronary artery bypass grafts based on a pod-galerkin method and a vascular shape parametrization. *J Comput Phys*. 2016;315:609-628.
26. Tezzele M, Ballarin F, Rozza G. Combined parameter and model reduction of cardiovascular problems by means of active subspaces and pod-galerkin methods. *Mathematical and Numerical Modeling of the Cardiovascular System and Applications*. Springer; 2018:185-207.
27. Habibi M, Dawson S, Arzani A. Data-driven pulsatile blood flow physics with dynamic mode decomposition. *Fluids*. 2020;5(3):111.
28. Di Labbio G, Kadem L. Reduced-order modeling of left ventricular flow subject to aortic valve regurgitation. *Phys Fluids*. 2019;31(3):031901.
29. Manzoni A, Quarteroni A, Rozza G. Model reduction techniques for fast blood flow simulation in parametrized geometries. *Int J Numer Methods Biomed Eng*. 2012;28(6-7):604-625.
30. Lassila T, Manzoni A, Quarteroni A, Rozza G. A reduced computational and geometrical framework for inverse problems in hemodynamics. *Int J Numer Methods Biomed Eng*. 2013;29(7):741-776.
31. Colciago CM, Deparis S. Reduced numerical approximation of reduced fluid-structure interaction problems with applications in hemodynamics. *Front Appl Math Stat*. 2018;4:18.
32. Walton S, Hassan O, Morgan K. Reduced order modelling for unsteady fluid flow using proper orthogonal decomposition and radial basis functions. *Appl Math Model*. 2013;37(20-21):8930-8945.
33. Bogunović H, Pozo JM, Villa-Uriol MC, et al. Automated segmentation of cerebral vasculature with aneurysms in 3DRA and TOF-MRA using geodesic active regions: an evaluation study. *Med Phys*. 2011;38(1):210-222.

34. Villa-Uriol MC, Berti G, Hose DR, et al. @ neurist complex information processing toolchain for the integrated management of cerebral aneurysms. *Interface Focus*. 2011;1(3):308-319.
35. Lassila T, Sarrami-Foroushani A, Hejazi SM, Frangi AF. Population-specific modelling of between/within-subject flow variability in the carotid arteries of the elderly. *Int J Numer Methods Biomed Eng*. 2020;36(1):e3271.
36. Geers AJ, Larrabide I, Morales HG, Frangi AF. Approximating hemodynamics of cerebral aneurysms with steady flow simulations. *J Biomech*. 2014;47(1):178-185.
37. Ford MD, Alperin N, Lee SH, Holdsworth DW, Steinman DA. Characterization of volumetric flow rate waveforms in the normal internal carotid and vertebral arteries. *Physiol Meas*. 2005;26(4):477.
38. Wielicka M, Neubauer-Geryk J, Kozera G, Bieniaszewski L. Clinical application of pulsatility index. *Med Res J*. 2020;5(3):201-210.
39. Schöning M, Walter J, Scheel P. Estimation of cerebral blood flow through color duplex sonography of the carotid and vertebral arteries in healthy adults. *Stroke*. 1994;25(1):17-22.
40. Matsuo T, Watanabe S-i, Sorimachi M, Kanda M, Ohta Y, Takahashi T. Blood flow velocity waveforms in the middle cerebral artery at rest and during exercise. *Artif Life Robot*. 2011;16(3):398-402.
41. Kingma DP, Ba J. Adam: a method for stochastic optimization. *arXiv*. 2014.
42. Csala H, Dawson S, Arzani A. Comparing different nonlinear dimensionality reduction techniques for data-driven unsteady fluid flow modeling. *Phys Fluids*. 2022;34(11). <https://pubs.aip.org/aip/pof/article-abstract/34/11/117119/2848789/Comparing-different-nonlinear-dimensionality?redirectedFrom=fulltext>
43. Heaney CE, Liu X, Go H, et al. Extending the capabilities of data-driven reduced-order models to make predictions for unseen scenarios: applied to flow around buildings. *Front Phys Ther*. 2022;10:563.
44. Heaney CE, Wolffs Z, Tómasson JA, et al. An AI-based non-intrusive reduced-order model for extended domains applied to multiphase flow in pipes. *Phys Fluids*. 2022;34(5):055111.
45. MacRaidl M, Sarrami-Foroushani A, Lassila T, Frangi AF. Accelerated simulation methodologies for computational vascular flow modelling. *J R Soc Interface*. 2024;21(211):20230565.

## SUPPORTING INFORMATION

Additional supporting information can be found online in the Supporting Information section at the end of this article.

**How to cite this article:** MacRaidl M, Sarrami-Foroushani A, Lassila T, Frangi AF. Reduced order modelling of intracranial aneurysm flow using proper orthogonal decomposition and neural networks. *Int J Numer Meth Biomed Engng*. 2024;40(10):e3848. doi:[10.1002/cnm.3848](https://doi.org/10.1002/cnm.3848)