Physics of Fluids

Deep learning-based surrogate model for three-dimensional patient-specific computational fluid dynamics

Cite as: Phys. Fluids **34**, 081906 (2022); https://doi.org/10.1063/5.0101128
Submitted: 29 May 2022 • Accepted: 25 July 2022 • Accepted Manuscript Online: 01 August 2022 • Published Online: 22 August 2022

Pan Du, Xiaozhi Zhu and Jian-Xun Wang

Physics of Fluids ARTICLE scitation.org/journal/phf

Deep learning-based surrogate model for three-dimensional patient-specific computational fluid dynamics

Cite as: Phys. Fluids **34**, 081906 (2022); doi: 10.1063/5.0101128
Submitted: 29 May 2022 · Accepted: 25 July 2022 ·
Published Online: 22 August 2022

Pan Du,[1,2] Xiaozhi Zhu,[3] and Jian-Xun Wang[1,2,4,a)]

AFFILIATIONS

[1]Department of Aerospace and Mechanical Engineering, University of Notre Dame, Notre Dame, Indiana 46556, USA
[2]Lucy Family Institute for Data & Society, University of Notre Dame, Notre Dame, Indiana 46556, USA
[3]Applied and Computational Mathematics and Statistics, University of Notre Dame, Notre Dame, Indiana 46556, USA
[4]Center for Sustainable Energy (ND Energy), University of Notre Dame, Notre Dame, Indiana 46556, USA

[a)]**Author to whom correspondence should be addressed:** jwang33@nd.edu. **Tel.:** +1 574-631-5302

ABSTRACT

Optimization and uncertainty quantification have been playing an increasingly important role in computational hemodynamics. However, existing methods based on principled modeling and classic numerical techniques have faced significant challenges, particularly when it comes to complex three-dimensional (3D) patient-specific shapes in the real world. First, it is notoriously challenging to parameterize the input space of arbitrary complex 3D geometries. Second, the process often involves massive forward simulations, which are extremely computationally demanding or even infeasible. We propose a novel deep learning surrogate modeling solution to address these challenges and enable rapid hemodynamic predictions. Specifically, a statistical generative model for 3D patient-specific shapes is developed based on a small set of baseline patient-specific geometries. An unsupervised shape correspondence solution is used to enable geometric morphing and scalable shape synthesis statistically. Moreover, a simulation routine is developed for automatic data generation by automatic meshing, boundary setting, simulation, and post-processing. An efficient supervised learning solution is proposed to map the geometric inputs to the hemodynamics predictions in latent spaces. Numerical studies on aortic flows are conducted to demonstrate the effectiveness and merit of the proposed techniques.

I. INTRODUCTION

Cardiovascular disease (CVD) remains the first leading cause of death and morbidity in the United States, posing a major healthcare concern.[1] Although medical imaging techniques, for example, computed tomography (CT) and magnetic resonance imaging (MRI), have enabled the acquisition of exquisite anatomical information and revolutionized cardiovascular medicine,[2] they are usually not able to provide hemodynamic information (e.g., flow field and pressure losses) on their own, which is believed to be more important to CVD diagnosis and prognosis.[3] Computational models based on physical principles of cardiovascular systems, combined with medical imaging, enable the derivation of functional information inaccessible by medical images alone. In the past decade, image-based computational modeling has become a paradigm in cardiovascular research[4–6] and is pioneering new clinical applications.[7–9] However, the first-principle physics-based models are often computationally expensive since they involve solving a large-scale discrete partial differential equation (PDE) system using numerical techniques, for example, finite volume or finite element methods (FVM/FEM). Particularly, when studying dynamics of blood flows (i.e., hemodynamics), computational fluid dynamics (CFD) and/or fluid–structure interaction (FSI) models are needed, which are enormously time-consuming (requiring supercomputing clusters) and prone to numerical challenges (requiring significant domain expertise). These roadblocks have primarily limited image-based CFD modeling in clinical applications that require timely feedback for further therapeutic assessment and treatment planning. For example, in the case of Heartflow (a medical company for precision heart care), image data must be sent outside the hospital to be processed by in-house supercomputers and results sent back to the hospital. Likewise, the high computational cost prohibits many-query simulations for uncertainty quantification (UQ), parameter estimation, design optimization, etc., which are becoming increasingly

vital in advancing the utility of cardiovascular modeling to practical applications.[10–15]

As a more efficient alternative to the principled CFD/FSI models, reduced-order modeling (ROM) has been an area of intense investigation for years. One class of ROM is constructed by simplifying the physics of cardiovascular systems, such as lumped parameter (LP) models, one-dimensional (1D) models, or hybrid 3D/1D/LP models,[16–18] where the physical dimensions are reduced to different levels. For example, LP models typically treat the hemodynamic system as an electric circuit, where the relations among pressure drop, viscous effects, and flow velocity are modeled as simplified ordinary differential equations (ODEs). Although LP/1D models are very cheap, they only can predict global information (integral quantities) instead of local ones [e.g., spatiotemporal fields of velocity or wall shear stresses (WSS)], while the latter is more useful to advance cardiovascular research/health care. To preserve detailed hemodynamics information, the other class of ROM is built by projecting the full-order governing PDEs (e.g., Navier–Stokes equations) onto a reduced subspace spanned by a group of basis functions, such as proper orthogonal decomposition (POD) modes, known as POD–Galerkin projection, which has been developed for facilitating hemodynamics analysis in many 3D patient-specific configurations.[19–22] However, the projection-based ROM is often less stable in parametric settings and highly code-intrusive, posing great challenges to leveraging legacy CFD/FSI solvers.[23–25] With the increasing data availability and recent advances in machine learning (ML), there has been growing interest in developing non-intrusive data-driven approaches for surrogate modeling of cardiovascular systems. In general, data-driven surrogate models aim to learn the mapping between modeling inputs (e.g., geometry, inlet/outlet, material properties) and computed outputs (e.g., resolved flow field, pressure contour, wall shear stress distribution) based on full-fidelity CFD/FSI simulation data. A well-trained surrogate model can predict detailed hemodynamics information rapidly, supporting real-time or many-query applications. For example, collocation-based polynomial chaos expansion (PCE) has been used to build non-intrusive surrogates of 3D blood flow simulations for uncertainty quantification (UQ) tasks.[10,11] Gao *et al.*[14,15] developed a bi-fidelity (BF) surrogate model that leverages the efficiency and accuracy of the low- and high-fidelity (HF) CFD simulations, respectively. Deep neural network (DNN) has become a popular surrogate modeling approach, renowned for their universal functional approximation capability for high-dimensional system.[26,27] Most recently, DNN-based surrogate models have been developed for cardiovascular applications and shown great potential.[28–35]

However, ML-based surrogate modeling for complex patient-specific hemodynamics still faces significant challenges that existing techniques struggle to address simultaneously. First, the parameterization of the geometric space spanned by irregular 3D patient-specific shapes is very challenging. As a result, most existing ML-based surrogate hemodynamic models focus on inlet/outlet parameterization or studying idealized geometries, which can be easily parameterized with a few descriptors (e.g., radius), and only a few studies dealt with 3D patient-specific cases.[34,35] Second, a well-performed ML surrogate model often requires a large number of training samples. However, patient-specific geometries are commonly obtained by segmentation from medical images [e.g., CT and magnetic resonance angiography (MRA)], which are often cumbersome and time-consuming, leading to a prohibitive cost for data generation. To address the issue, it is necessary to effectively synthesize numerous new samples from a small set of patient-specific geometries obtained by image segmentation.

In this paper, we try to fill the gaps and develop an ML-based surrogate modeling framework for rapid predictions of comprehensive hemodynamics information in 3D patient-specific aortic geometries. In contrast to many ML-based surrogates that predict global hemodynamics information such as pressure loss, FFR, and mean WSS, this work focuses on predicting local blood flow information (e.g., velocity, pressure, and WSS fields), given irregular 3D patient-specific shapes. The contributions of the current paper are summarized as follows. First, we developed a method to parameterize 3D patient-specific aorta geometries based on stochastic shape modeling (SSM). Second, a generative model is proposed to synthesize a massive amount of 3D geometries from a small patient-specific dataset. Third, an automatic simulation data generation routine is built, enabling automatic meshing, prescribing boundary conditions, solving, and post-processing. Finally, an efficient ML-based surrogate is developed by learning the functional map from geometries to hemodynamics in latent spaces, and the performance of using DNN and bi-fidelity technique is compared. The rest of the paper is organized as follows: The proposed framework is introduced in Sec. II. Numerical results of the surrogate modeling are presented and discussed in Sec. III, and Sec. IV concludes the paper.

II. METHODOLOGY

The pipeline of the proposed ML-based surrogate modeling framework is summarized by Fig. 1:

(i) a small set of 3D aorta geometries are reconstructed from anatomical images acquired via CTA/MRI scans on real patients; (ii) the geometric space spanned by the group of real patient-specific geometries is parameterized based on statistical shape modeling (SSM), where the variation of aorta surfaces can be described mathematically;

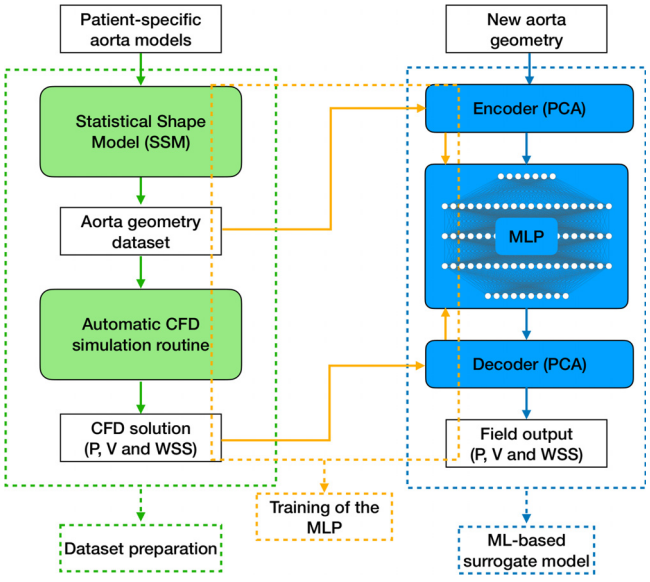

FIG. 1. The schematic of the proposed ML-based surrogate modeling of patient-specific aortic flows, with several components, including dataset generation (green box), network training (orange box), and DNN inference/prediction (blue box).

(iii) a large number of virtual 3D aorta shapes are synthesized within the latent geometric space; (iv) a python-based simulation routine is developed for automating meshing, equation solving, and post-processing of aortic CFD simulations on the generated ensemble of 3D synthetic shapes, and a large dataset of hemodynamics (e.g., pressure, velocity, and WSS fields) are obtained; (v) The 3D vascular shape and hemodynamic fields of interest (FOI) of each aorta sample are encoded into latent spaces using principal component analysis (PCA) or proper orthogonal decomposition (POD), and then, a DNN is constructed to learn the non-linear mapping within the latent space. Once the network is sufficiently trained, the ML surrogate model will be able to predict the fully resolved 3D hemodynamic flow information within a split second, given a new aorta shape.

A. Patient-specific aorta geometry parameterization

1. Aorta dataset preprocessing

Eight 3D aorta geometries are reconstructed from MRI scans of patients with coarctation of the aorta (COA) conditions by the German Heart Institute Berlin.[36] The geometric characteristic of this illness is that there is a narrowing near the aortic arch such as "A1" and "A2" in Fig. 3. Prior to the parameterization of the variation of aorta shapes, one crucial step is to build correspondence among them, which is very challenging since the deformation across different patients is non-isometric, especially aortas with a complex topological structure (e.g., many branches). To reduce complexity, this work is focused on 3D aortic geometries with homeomorphism. In particular, several branches of the original thoracic aorta geometries, including left subclavian artery (LSA), left carotid artery (LCA), right subclavian artery (RSA), and right carotid artery (RCA), are trimmed off, simplifying their structure to a single channel. Later, a point cloud of 4000 vertices is sampled from each aorta using Voronoi clustering implemented via the pyacvd module in python, which is loosely based on approximated centroidal Voronoi diagrams (ACVD).[37] To regularize the position and the orientation of the aorta samples in the 3D Cartesian coordinate system, rigid iterative closest point (ICP) algorithm[38] is adopted to align every aorta geometry to a template selected from the aorta dataset.

2. Statistical shape modeling and virtual aorta synthesis

With the aligned real aorta geometries, we establish an SSM framework to parameterize the complex 3D patient-specific geometric input space and synthesize enormous virtual aorta geometries. The procedure is conducted in three phases: (1) building correspondence; (2) projection to latent space; (3) synthesizing virtual aortas.

a. Building correspondence. In order to synthesize new aorta shapes, it is necessary to build correspondence between different aorta geometries. In particular, suppose we have several patient-specific aorta geometries. We first sample 4000 points on each aorta surface using a random sampling algorithm. If one directly linearly interpolates between a random pair of point clouds, the new geometry surface will not be smooth or even have cracks, indicating a failure of the synthesization. As a result, we sampled 4000 points on one template geometry and deform the point cloud to every rest aorta point cloud. Since we are using those point clouds as the surface vertices of the aorta mesh, it is necessary that the deformation function preserves the relative position of the points in the point cloud and hence maintain the correctness of the mesh. The process of building correspondence can be formulated as follow:

Let $\{A^i\}_{i=1,\dots,n}$ denote the set of n original patient-specific aorta shapes, and each shape sample $A^i \in \mathbb{R}^{3\times p}$ is represented by a point cloud of surface vertices $A^i = \{\mathbf{a}_j^i\}_{j=1}^p$, where p is the total number of vertices. Given a template geometry $A^S \in \{A^i\}_{i=1,\dots,n}$ selected from the set $\{A^i\}_{i=1,\dots,n}$, we aim to build correspondence by deforming the template A^S to all the other aorta shapes $A^T \in \{A^i\}_{i\neq s}$, which are referred to as source geometry and target geometries, respectively. The diffeomorphism function $A^T = \Phi(A^S) : \mathbb{R}^{3\times p} \to \mathbb{R}^{3\times p}$, describing the deformation from the template A^S to each of the rest aorta samples $\{A_i\}_{i\neq S}$, can be solved based on the controlled-point-based large deformation diffeomorphic metric mapping (LDDMM) algorithm.[39] In LDDMM, the deformation is described as a "current" of surface, which is the flux of a 3D "velocity" vector field over the template $\{\mathbf{a}_i^S\}_{i=1}^p$,

$$\boldsymbol{v}(\mathbf{a}_i) = \sum_{j=1}^{r} K(\mathbf{a}_i, \mathbf{q}_j)\cdot \boldsymbol{\mu}_j, \tag{1}$$

where $\{\mathbf{q}_j\}_1^r$ and $\{\boldsymbol{\mu}_k\}_1^r$ represent a set of r "control" points and corresponding "momentum" vectors, which are defined on the source geometry A^s; K is a Gaussian kernel $K(\mathbf{a},\mathbf{q}) = \exp(-||\mathbf{a}-\mathbf{q}||_{L_2}/\sigma^2)$ with σ controlling the typical width of the deformation. The deformation is performed along the pseudo time axis τ, and the deformed point cloud locations at τ are denoted by $A^s(\tau)$, where $A^s(t=0)$ represents the initial state (i.e., the point cloud $\{a_j^s\}_{j=1}^p$ of template A^s). The point deformations are governed by the following ODEs:

$$\dot{A}^s(\tau) = \mathbf{v}(A^s(\tau), \tau), \tag{2}$$

which can be solved using the second-order Runge–Kutta method, and meanwhile, the control points and momentum vectors are updated based on the Hamiltonian dynamics

$$\begin{cases} \dot{\mathbf{q}}(\tau) = \mathbf{K}(\mathbf{q}(\tau),\mathbf{q}(\tau))\boldsymbol{\mu}(\tau), & \text{(3a)} \\ \dot{\boldsymbol{\mu}}(\tau) = -\frac{1}{2}\nabla_{\mathbf{q}}\mathbf{K}(\mathbf{q}(\tau),\mathbf{q}(\tau))\boldsymbol{\mu}(\tau)^{\top}\boldsymbol{\mu}, & \text{(3b)} \end{cases}$$

where $\mathbf{K}$ represents the kernel matrix and $\{\mathbf{K}[\mathbf{q}(\tau),\mathbf{q}(\tau)]\}_{i,j} = K[\mathbf{q}_i(\tau), \mathbf{q}_j(\tau)]$. We force the deformed template $\tilde{A}^S_{\tau^e}$ at the final step (i.e., $\tau = \tau^e$) to match the target geometry A^T using the steepest gradient descent optimization algorithm. To define the loss function, a "distance" metric, evaluating the difference between the deformed template $\tilde{A}^S_{\tau^e}$ and the target geometry A^T, is required. As the aorta geometries are composed of triangular meshes with grid centers $\{\mathbf{c}_i\}_{i=1}^m$ and edge normal vectors $\{\mathbf{n}_i\}_{i=1}^m$, the varifold distance metric d is defined as

$$d\left(\left\{\left(\mathbf{n}_i^S, \mathbf{c}_i^S\right)\right\}_{i=1}^{m^S}, \left\{\left(\mathbf{n}_j^T, \mathbf{c}_j^T\right)\right\}_{j=1}^{m^T}\right) = \sum_i^{m^S}\sum_j^{m^T} K\left(\mathbf{c}_i^S, \mathbf{c}_j^T\right)\cdot \frac{\left(\left(\mathbf{n}_i^S\right)^{\top}\cdot \mathbf{n}_j^T\right)^2}{||\mathbf{n}_i^S||\cdot||\mathbf{n}_j^T||}. \tag{4}$$

The metric is calculable regardless of whether point-to-point correspondence exists between two geometries. Namely, the total amount of grids m^S and m^T for source and target geometries can be different. The tolerance of the distance error is 1×10^{-4} for every deformation optimization. Once the diffeomorphism functions are solved, all target geometries can be approximated by the deformations from the template, establishing a correspondence for the original shape dataset. The LDDMM algorithm has been implemented in Deformetrica, an open-source software designed for statistical analysis of meshes.[40]

b. Projection to latent space.. To enable a concise parameterization of the input geometric space, PCA is applied to encode the 3D irregular shape onto the latent space. Specifically, $\{(x_j^i, y_j^i, z_j^i)\}_{j=1,\ldots,p}$, denoting the Cartesian coordinates of the point cloud on the aorta sample $\tilde{A}_i$, was flattened into a vector $\tilde{\boldsymbol{V}}^i = (x_1^i, y_1^i, z_1^i, \ldots, x_p^i, y_p^i, z_p^i)$. The entire 3D shape dataset can be squeezed into a matrix $\boldsymbol{A} = (\tilde{\boldsymbol{V}}_1^\top, \ldots, \tilde{\boldsymbol{V}}_N^\top)^\top$. Subsequently, PCA acting on A yields the eigenvectors $\{\boldsymbol{W}_i\}_{i=1,\ldots,z}$ and eigenvalues $\{\lambda_i\}_{i=1,\ldots,z}$, where z is the truncation integer controlling the number of primary components needed. Each aorta shape can be decomposed as follows:

$$\tilde{\boldsymbol{A}}^i \approx \boldsymbol{A}_{mean} + \sum_{j=1}^{z} \alpha_j^i \sqrt{\lambda_j} \boldsymbol{W}_j, \tag{5}$$

where the coefficient vector $\boldsymbol{\alpha}_i = (\alpha_1^i, \ldots, \alpha_z^i)$ represents the encoded shape, which, in other words, is the projection of sample A_i in the latent space. Let F and F^{-1} denote the PCA operator and its inverse, respectively. The reconstruction error E_{pca} is defined to evaluate the accuracy of the PCA transform

$$E_{pca} = \frac{||F^{-1}(F(\boldsymbol{A})) - \boldsymbol{A}||_{L2}}{||\boldsymbol{A}||_{L2}}. \tag{6}$$

c. Synthesizing virtual aorta geometries.. Motivated by the need of sufficient training samples for data-driven ML surrogate modeling, we propose a shape synthesizing method for producing a large amount of synthetic 3D aortas from a small set of original aorta shapes. The general idea is to sample from the latent geometric space spanned by the original aorta shapes and then synthesize new aortas by decoding the sampled latent shape vectors back to the Cartesian space, as illustrated in Fig. 2, where the geometric space is spanned by three original aorta shapes. To synthesize new geometries within the geometric space, we propose two different methods: (1) random linear interpolation and (2) uniform PCA sampling. The random linear interpolation approach linearly combines the n original samples in the latent space with randomly assigned weights to generate new samples $\boldsymbol{A}^*$,

$$\boldsymbol{A}^* = F^{-1}\left(\sum_{i=1}^{n} \omega_i \boldsymbol{\alpha_i}\right) = \boldsymbol{A}_{mean} + \sum_{i=1}^{n}\sum_{j=1}^{z} \omega_i \alpha_j^i \sqrt{\lambda_j} \boldsymbol{W}_j, \tag{7}$$

where $\boldsymbol{\omega} = [\omega_1, \ldots, \omega_n]^T$ is a randomly generated weight vector that satisfies $\sum_{i=1}^{z} \omega_i = 1$. As an alternative, uniform PCA sampling approach generates new samples by uniformly perturbing the PCA coefficients within the ranges bounded by the original geometries

$$\boldsymbol{A}^* = \boldsymbol{A}_{mean} + \sum_{j=1}^{z} \tilde{\alpha}_j^* \sqrt{\lambda_j} \boldsymbol{W}_j, \tag{8}$$

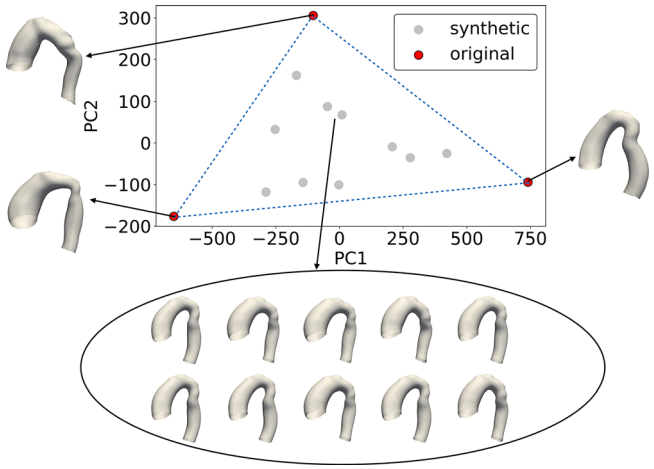

FIG. 2. Illustration of latent space distribution and visualization of generated synthetic aortas (gray circle) from 3 original aorta shapes (red circle), where PC1 and PC2 represent the first and second primary components of the latent geometric space, respectively.

where $\tilde{\boldsymbol{\alpha}}^* = [\tilde{\alpha}_1^*, \ldots, \tilde{\alpha}_z^*]^T$ is a randomly generated vector, and each dimension $\tilde{\alpha}_j^*$ is sampled from a uniform distribution U $[\min(\alpha_j^i|_{i=1}^n), \max(\alpha_j^i|_{i=1}^n)]$. Figure 2 plots a few generated samples within the 2D latent space defined by its first (PC1) and second (PC2) primary components and the corresponding synthetic geometries. The synthetic geometries (blue) are scattered within the triangular region bounded by three original geometries (orange).

B. Automatic CFD simulation routine

To systematically generate massive CFD simulation data, we develop a python routine to automate tedious simulation procedures, including geometry preprocessing, meshing, boundary setting, simulation, and post-processing. As the first step, preprocessing of the aorta geometries is needed in order to be suitable for CFD simulations. First, since the edges at the entry and the exit of the raw aorta channel could be jaggy, the entry is slightly extended to construct a circular rim at the inlet. Similarly, the exit of the aorta is also extended, which also avoids inverse flows at the outlet. Second, the aorta surface is smoothed to reduce bumpiness while still preserving its geometric characteristics. Third, caps are added to seal nonphysical holes on the aorta geometry. Once the geometry preprocessing is done, 3D unstructured triangular meshes are generated accordingly and then converted into *msh* format suitable for subsequent CFD simulations. These operations are implemented based on the Vascular Modeling Toolkit (VMTK),[41] which is an open-source module specialized in 3D reconstruction, geometric analysis, and mesh generation. A python wrapper is implemented to automate the entire procedure in parallel for the ensemble setting.

Subsequently, a large-size ensemble of CFD simulations is performed to obtain the corresponding FOIs for each sample of the generated dataset, based on the open-source CFD platform, OpenFOAM.[42] As handling a massive set of cases can easily be cumbersome and time-consuming, a python subroutine coupled with OpenFOAM is established, such that the procedure of each simulation,

including meshing, setting up cases, specifying boundary conditions, solving, and post-processing, is automatically executed and repeated through all samples in the dataset. The subroutine is written in a bash script, and the process is formulated as follows: (a) A template case is manually built with desired settings (e.g., numerical scheme, convergence threshold, boundary conditions). Once the script is called, a copy of the template case will be created and the geometry file will be transferred into OpenFOAM mesh format. (b) The SIMPLE solver is subsequently called to execute the simulation. (c) Several post-process utility transfer solution results to convenient data formats; that is, the probing function is used to directly obtain fluid quantities in an array file. This process can be conducted in parallel as well. During the post-processing step, the simulated field data on CFD meshes are projected onto the surface or volumetric mesh grids where the correspondences are established. In particular, surface pressure and wall shear stress are projected onto surface vertices and velocity data are projected onto the volumetric nodes via the nearest-neighbor interpolation. The volumetric nodes are obtained by uniformly sampling points between the centerline and the surface vertices. Finally, a large number of input geometries and corresponding flow solutions are obtained as the labeled dataset for surrogate modeling.

C. ML-based surrogate model

In order to build a fast forward map from the geometric space to the solution space of interest, we propose a supervised deep learning solution based on PCA encoding–decoding and fully connected deep neural networks.

1. Encoding–decoding formulation

Considering the significantly high dimensions of 3D shapes and solution fields in their discrete mesh forms, we adopt an encoding–decoding strategy to reduce the learning complexity. Specifically, the input geometries and output hemodynamic solution fields are encoded into low-dimensional latent spaces. To promote training efficiency, the encoder–decoder construction is decoupled with DNN-based surrogate training, and PCA (or POD) instead of DNN-based autoencoder is used for the encoding–decoding process. The PCA algorithm is implemented via the SciPy python package.[43] To balance the reconstruction accuracy and learning complexity, we keep the first seven (maximum number of basis is eight) primary components for the geometry input and ten (maximum number of basis is 1000) primary components for the FOIs (e.g., pressure, velocity, and WSS), preserving 99.99% and over 95% of the total PCA variation, respectively. The reconstruction error is 0.00% for the input and less than 1.5% for the output solution fields. The relationship between the encoded shape and solution fields is learned by a multilayer perceptron (MLP), where rectified linear unit (ReLu) is used as the activation function.

2. Learning architecture optimization

The DNN-based forward map is learned from the labeled dataset using the Adam optimizer, where the learning rate is adaptively changed based on the estimates of the moments. To achieve the best learning performance, we optimize the MLP architecture and other learning hyperparameters, including the number of layers, the number of neurons of each layer, batch size, and initial learning rate, using a Bayesian optimization algorithm. Specifically, RAY-tune,[44] an open-source python module for scalable hyperparameter tuning, is adopted here to achieve this goal. The general idea is to construct a posterior distribution of functions based on Gaussian processes within the prescribed ranges of the hyperparameters to be optimized. During the optimization, the posterior of hyperparameters will be updated given more and more training data, leading to the best learning configuration. The Bayesian optimization is superior to a brutal grid search method and can reach the optimized state in a fewer iterations. During the optimization, the asynchronous successive halving algorithm (ASHA) is utilized to aggressively terminate non-ideal trials in advance in the interest of saving time, and more details can be found in Ref. 45.

3. Evaluation of performance

To evaluate the performance of the ML-based surrogate model, relative mean square error (RMSE) is used to calculate the difference between prediction and label, which is defined as

$$RMSE = \frac{\sum_{i}^{N} (x_i - y_i)^2}{\sum_{i}^{N} y_i^2} \times 100\%, \tag{9}$$

where $\boldsymbol{X} = (x_1, \ldots, x_N)$ and $\boldsymbol{Y} = (y_1, \ldots, y_N)$ are predictions and labels of FOI on N vertices/grids, respectively. For scalar fields such as pressure, RMSE is calculated directly upon the pressure field. For vector fields like wall shear stress and velocity, an array flattened from the $N \times 3$ vector–matrix containing three directional components (x, y, and z directions) and the array of the vector magnitude are calculated to estimate the RMSE, referred to as vector RMSE and magnitude RMSE, respectively. When the error is evaluated upon a test dataset with multiple samples, the average of the RMSE of each sample is calculated.

III. RESULTS AND DISCUSSION

A. Statistical shape analysis and synthesis

We first register the eight original patient-specific aorta geometries by solving seven diffeomorphisms from a randomly chosen template to the rest of seven geometries. An example of such diffeomorphism is demonstrated in Fig. 3. The template A_1 is first sampled with 4000 points P_1 and so is the target geometry A_2 with P_2. The shape deformation $A(\tau)$ is solved with the boundary condition of $A(\tau = 0) = P_1$ and $A(\tau = 1) = P_2$. Several intermediate states of the function A are shown in the middle, in which the point clouds near the left side resemble P_1 and, likewise, point clouds near the right side resemble P_2. In addition, the intermediate states can be used to "interpolate" between 3D irregular, showing geometric morphing from aorta A_1 to aorta A_2.

Based on the set of eight original aorta geometries, a large number of virtual aortas are synthesized by sampling the geometric space spanned by the original shapes. Both the uniform PCA sampling and random interpolation methods are implemented, and the sample distributions of them are plotted in Fig. 4 to compare their coverage of the input space. The 1000 synthetic shapes are scattered in the coordinate plane of their PCA latent space with the first (PC1) and second (PC2) primary components. The synthetic samples obtained by the uniform PCA sampling are much more spread out over the coordinate

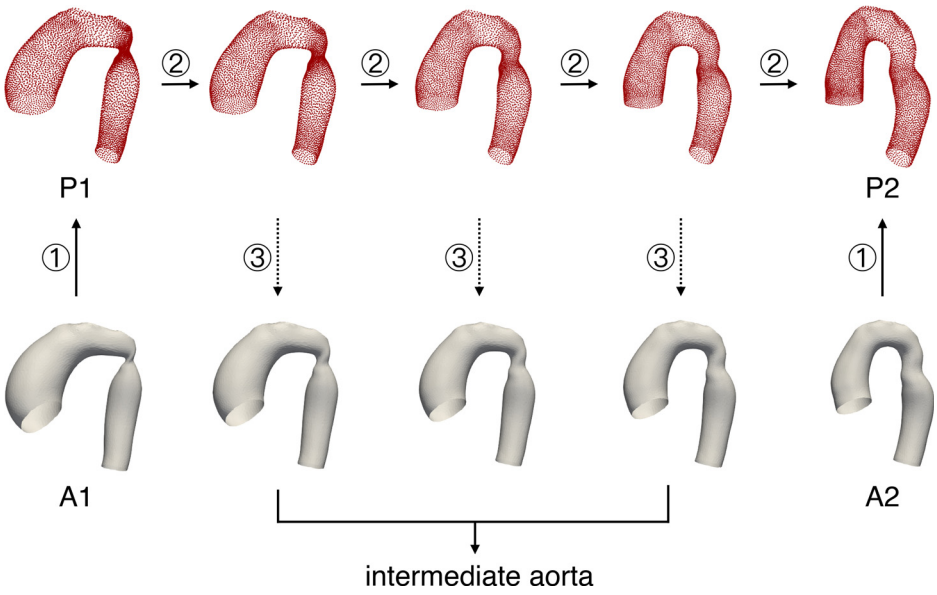

FIG. 3. Building correspondence: Step ①, sample vertices from surface; Step②, solve the diffeomorphism function; Step③, reconstruct intermediate aorta.

plane than those from the random interpolation method, indicating a better coverage of the input space and hence adopted for the data generation in this work.

Part of the synthetic geometries and the eight original geometries are shown in Fig. 5. The uniform sampling method successfully explores the input topological space and the synthetic geometries, though virtual, possess geometric characteristics of biological shapes of real aortas and fully displays the variation of the aorta shapes.

B. Surrogate CFD modeling using deep learning

Once the aorta geometry dataset is built, ensemble CFD simulations are conducted to generate labeled hemodynamics data, which will be used to train an ML-based surrogate.

1. Numerical setup

In this work, we made the following assumptions: (1) the blood flow is Newtonian and governed by the steady-state incompressible Navier–Stokes equations; (2) the vessel walls are rigid; (3) the inlet velocity has a parabolic profile with a maximum magnitude of 1 m/s, which is close to the peak blood flow velocity during systole of one cardiac cycle in a human body. A zero-gradient boundary condition is applied at the inlet for pressure and a constant reference pressure with outflow boundary condition is set at the outlet. The OpenFOAM, an open-source C++ library for FVM, is used for CFD simulations. In particular, the semi-implicit method for pressure linked equation (SIMPLE) algorithms[46] were used for solving the incompressible Navier–Stokes equations, and the Rhie and Chow interpolation with collocated grids was employed to prevent the pressure–velocity decoupling.[47] All CFD simulations were run in parallel [four central processing units (CPUs) per case], and each simulation takes around 1.2×10^3 s to reach the convergence. Three decoupled MLPs are constructed to predict the velocity, surface pressure, and WSS fields separately. The network architectures and learning parameters are optimized using RAY-tune module, which is given in Table V. The DNNs are trained on an Nvidia RTX A6000 GPU with a minimum of 2000 epochs to ensure convergence.

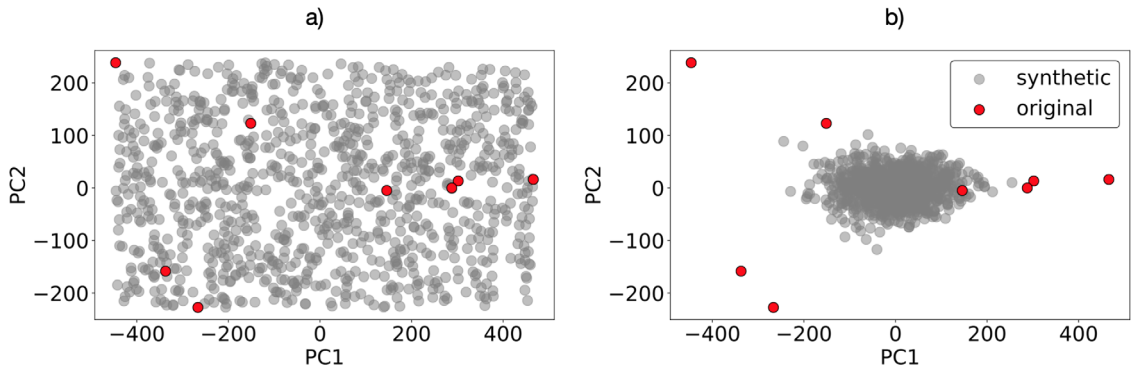

FIG. 4. Latent space distribution of synthetic (gray circle) and original (red circle) samples using (a) uniform PCA sampling and (b) random interpolation sampling.

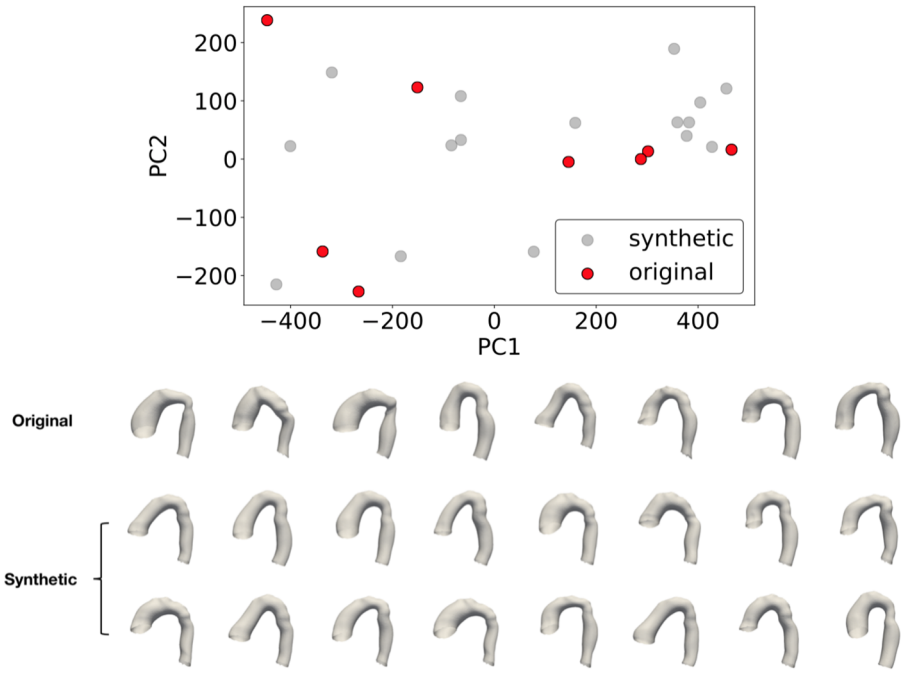

FIG. 5. Latent space distribution (top) and visualization (bottom) of synthetic (gray circle) and original (red circle) aorta samples.

2. Learning performance study

As a demonstration, we generated a dataset of $N_{tot} = 1000$ aorta geometries and CFD simulated labels for training and validation study. The dataset is divided into a training set (TS) and testing set with the ratio of $r = N_{train}/(N_{train} + N_{test})$, where N_{train} and N_{test} denote the size of training set and test set, respectively. The learning-prediction performance of the trained DNN surrogates with different training-testing ratios is reported in Table I, where the averaged MSEs of the DNN-predicted surface pressure, velocity, and WSS fields are compared. Considering the randomness of the DNN initialization, the training processes for all DNNs are repeated by ten times with different random initializations, and the averaged prediction errors over these trials are reported for a more rigorous assessment. It can be seen that the relative errors of the NN surrogate prediction are very low once the DNNs are sufficiently trained. For velocity field prediction, the prediction error is less than 1% even only 60% of the dataset is used for training. Compared with the velocity, the errors of surface field predictions such as surface pressure and WSS fields are slightly higher since they are more sensitive to the shape of vessel walls and the boundary layer conditions, which are not easy to capture well. Also, for vector-valued fluid quantities like velocity and WSS, the magnitude errors are smaller than the vector errors by about 30%. Viewing the columns for each flow quantity, the prediction accuracy increases as the training size grows, which is as expected. For example, the pressure error climbs from 1.48% at $r = 90\%$ to 1.83% at $r = 60\%$ as r decreases. In general, the prediction of the FOIs is accurate, indicating the great capability of the proposed DNN-based surrogate model.

TABLE I. DNN prediction errors (averaged MSE) with different training to testing size ratios for hemodynamics fields of interest, including surface pressure (ϵ_P), velocity vector (ϵ_V), velocity magnitude (ϵ_{VM}), wall shear stress (ϵ_{WSS}), and wall shear stress magnitude (ϵ_{WSSM}).

$r = \frac{N_{train}}{N_{train}+N_{test}}$ (%)	ϵ_P (%)	ϵ_V (%)	ϵ_{VM} (%)	ϵ_{WSS} (%)	ϵ_{WSSM} (%)
r = 60	1.83	0.735	0.504	1.88	1.33
r = 70	1.81	0.726	0.500	1.86	1.31
r = 80	1.67	0.734	0.505	1.86	1.32
r = 90	1.48	0.640	0.434	1.74	1.26

3. Predicting local hemodynamic information

In contrast to traditional surrogate model (e.g., GP models) or reduced-space model (e.g., LP/1D models), the proposed ML-based surrogate model is able to rapidly predict local hemodynamic information such as blood flow pattern, surface pressure, and WSS distributions over the vessel walls. For illustration, we randomly selected four test aorta geometries, and the hemodynamic fields predicted by the DNN surrogate ($r = 80\%$) are compared with the CFD reference (referred to as ground truth), as shown in Figs. 6 and 7. In addition, the absolute error contours are also plotted out by subtracting the ground truth from the prediction. The averaged and relative error is also shown in Table II.

For pressures, the surrogate model yields very close surface distribution to the ground truth, yet slight differences exist at the stenosis of the descending aorta. From the error contours, this discrepancy is very mild on samples 1–4, but notable on sample 5, where the severity of stenosis is much higher than other samples. This indicates that the surrogate model is slightly less accurate in capturing flow features with large pressure gradients, since more flow vortices are induced by sharp changes of surface curvatures complicating the flow physics. This contrast is conspicuous when comparing the maximum (sample 1) and minimum (sample 5)

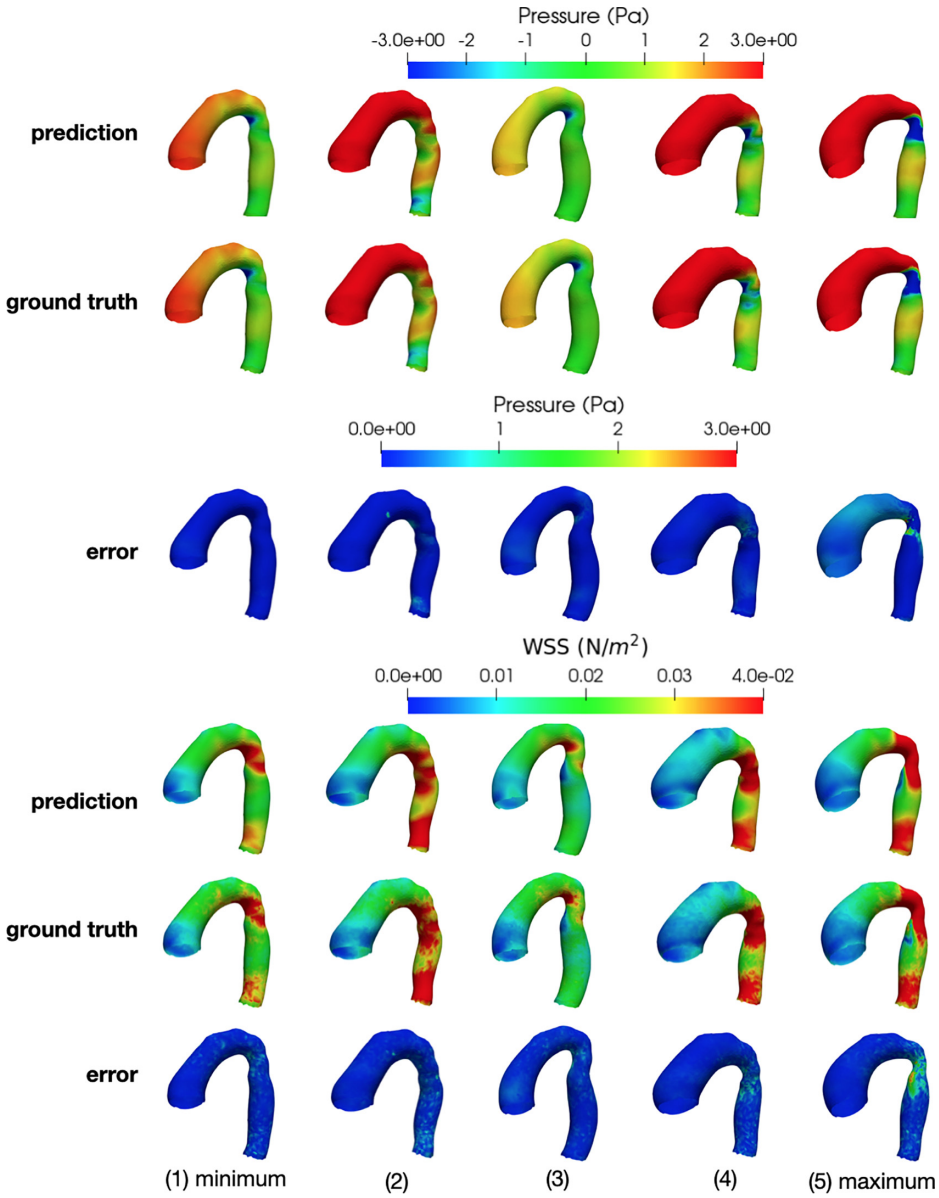

FIG. 6. ML surrogate predictions of pressure (top) and wall shear stress (bottom) are compared with the CFD ground truth: prediction (1st and 4th row), CFD ground truth (2nd and 5th row), and prediction error (3rd and 6th row). Samples 1 and 5 represent minimum and maximum predicted solutions, respectively.

predicted solution samples in Fig. 6. Sample 5 has a drastic change in surface shape near the narrowing section resulting large pressure gradients near the aortic arch, whereas sample one has a rather smooth arch with less narrowing condition, leading to a much more accurate pressure prediction. From Table II, one can see the maximum pressure error is 8.93%, much larger than the average value (1.48%). The maximum error can be reduced if one increases the sample density locally since the flow pattern changes rapidly at the maximum sample location in the input space surrogate prediction of the WSS distribution is in a good agreement with the ground truth. The pattern and low/high WSS regions can be accurately captured. However, the surrogate model tends to smooth out the small WSS fluctuations compared with the CFD reference. It appears that high-frequency components of the output are partially trimmed off when interpolating among neighboring training samples based on the non-linear relations learned by the neural network. Similar to pressure, notable inconsistency is also observed near the stenosis on the descending aorta for sample 3. The velocity contours on the cross section and internal velocity vector fields are visualized in Fig. 7. Table II shows a maximum WSS prediction error of 3.74% and 2.23% for relative vector and magnitude error, respectively, which are also reflected on aortas with sharp surface curvature changes. Again, the predicted velocity cross-sectional contours agree with the CFD ground truth very well, and the overall velocity vector fields from the neural network and CFD are almost identical for all samples, although the error contours of planar velocity show slight prediction errors near the stenosis region for sample 3.

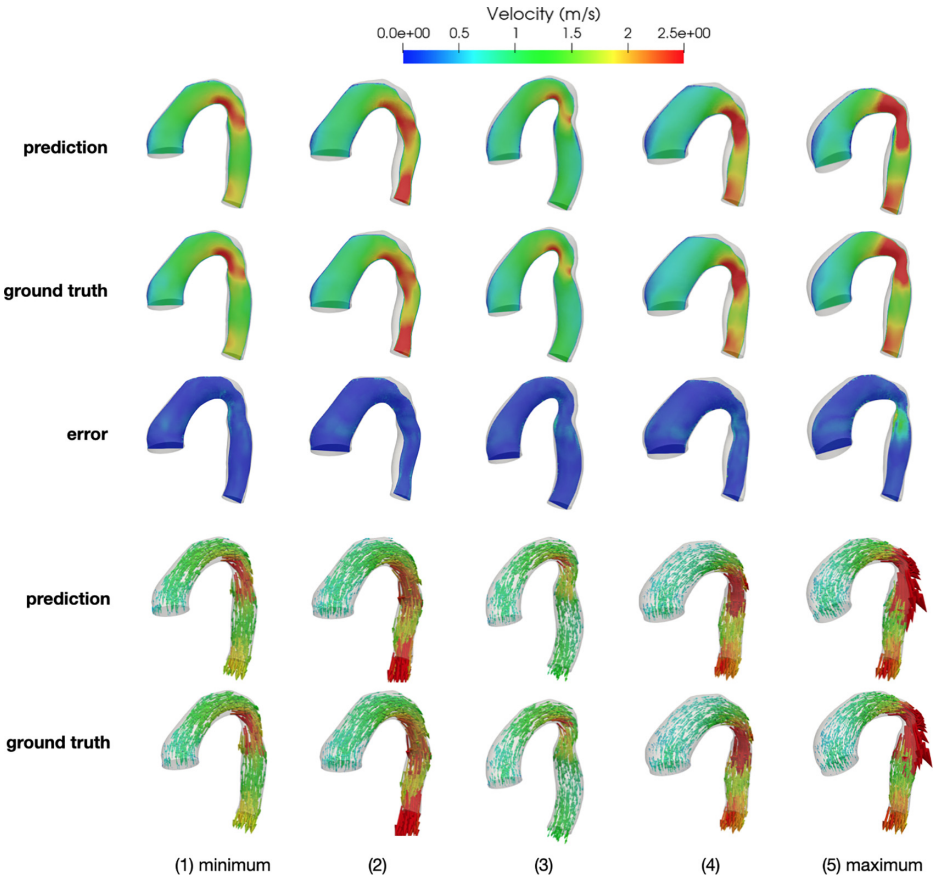

FIG. 7. Cross-sectional velocity magnitude (top) and velocity vector field (bottom) comparison between prediction and CFD ground truth: prediction (1st and 4th row), ground truth (2nd and 5th row), and error (3rd row). Samples 1 and 5 represent minimum and maximum predicted solutions, respectively.

TABLE II. Averaged and maximum DNN prediction errors for 900 training size and 100 testing size for surface pressure (ϵ_P), velocity vector (ϵ_V), velocity magnitude (ϵ_{VM}), wall shear stress (ϵ_{WSS}), and wall shear stress magnitude (ϵ_{WSSM}).

RMSE	ϵ_P (%)	ϵ_V (%)	ϵ_{VM} (%)	ϵ_{WSS} (%)	ϵ_{WSSM} (%)
Maximum	8.93	1.37	0.768	3.74	2.23
Averaged	1.48	0.640	0.434	1.74	1.26

TABLE III. Reconstruction and prediction error for geometry (ϵ_G), surface pressure (ϵ_P), velocity vector (ϵ_V), and wall shear stress (ϵ_W).

ARMSE	ϵ_G (%)	ϵ_P (%)	ϵ_V (%)	ϵ_{WSS} (%)
Reconstruction	0.00	0.74	0.61	1.5
Prediction	···	1.48	0.64	1.74

The maximum error for velocity prediction is also about one time larger than the averaged error, which is induced by flow inconsistency near the stenosis region as expected.

We also briefly discuss the effect of PCA reconstruction error on final prediction performance. Table III shows the comparison between the reconstruction error and the prediction error. The results show that the reconstruction error takes up a significant part of the prediction vector error, especially for the velocity. The final prediction error majorly comes from three sources: shape encoding, NN fitting, and the field reconstruction step. The field reconstruction error results from the truncation of PCA bases and grid interpolation. Because the training samples have slight orientation differences, the shape encoding error exists but is very small compared to other steps. The NN fitting error is determined by the NN architecture and learning parameters, which is minimized thanks to the Ray-tune module. The field reconstruction error introduces an additional error to the final prediction.

4. Training cost and prediction speedup

Table IV shows the time cost for every step in the surrogate pipeline. The mappings between shape input and FOIs, including pressure, velocity, and WSS, are learned using separated neural networks, and each takes about two minutes to be fully trained on an Nvidia RTX-3090 GPU. The network inference for solution field prediction takes less than 0.001 2 s. By contrast, a full-resolution CFD simulation needs about 20 min to reach the steady state, not counting the overhead for mesh generation and case setup, which could also be time-consuming. As a result, the fully trained DNN surrogate model has 100 000× speedup over the full CFD simulation in the online inference phase. We should note that a fair comparison should also consider the offline cost of

TABLE IV. Time cost for the pipeline for one sample (2nd row) and N samples (3rd row). Steps include: Preprocess (geometry manipulation); mesh generation; CFD simulation; NN training; prediction. Total time cost is in 4th row.

Time	Preprocess	Mesh generation	CFD simulation	NN training	Prediction
One sample	12.0 s	1.7 min	20.2 min	···	0.001 2 s
N = 1000 samples	3.3 h	28.33 h	336 h	12.1 min	1.2 s
Total		Offline time = 367.83 h			Online time = 1.2 s

constructing the surrogate, which includes the cost for network training and label generation. As for network training, the cost is negligible compared to that of a CFD simulation, thanks to the light DNN structure defined in the latent space. However, the cost for data generation depends on the number of training labels required, which is determined by the desired trade-off between the efficiency and accuracy of the surrogate. For example, supposing 1000 CFD labels are used for network training, the CFD data generation cost is $1000 \times 20(\text{min}) = 336(\text{h})$ min, which is significantly greater than that of a single CFD simulation. Nonetheless, the surrogate model is built for many-query applications (e.g., optimization, uncertainty quantification, inverse modeling), and the huge speedup of single model evaluation can be leveraged when a large number of model evaluations are required as the offline cost is paid off at once. For example, Markov chain Monte Carlo (MCMC), as the gold standard Bayesian inference method for forward and inverse UQ, usually requires hundreds of thousands of model evaluations to reach convergence, which well justifies the data generation at the cost of hundreds of full-order CFD evaluations. Moreover, it is worth noting that the training dataset size is application-dependent and the data generation cost can be significantly reduced if one allocates the computational budget wisely. A comprehensive study on the use of the proposed surrogate model in specific UQ or optimization applications is out of the scope of this work and will be conducted in the future.

C. Comparison between DNN-based and bi-fidelity surrogate models

Finally, we investigate how well the proposed DNN-based surrogate model performs compared with other state-of-art surrogate models that can provide full-field hemodynamic predictions. To enable the full-field prediction capability with the same resolution as that of full-order CFD simulations, a multi-fidelity strategy is often adopted for surrogate modeling. The multi-fidelity paradigm is pioneered by Kennedy and O'Hagan[48] decades ago, originally from a statistical point of view (e.g., GP-based multi-model approach). Very recently, multi-fidelity strategy has been applied for surrogate modeling in inference and uncertainty quantification (UQ) problems in cardiovascular biomechanics problems.[14,15,49–51] In particular, Gao and Wang[14] developed a bi-fidelity (BF) surrogate modeling approach for 3D patient-specific hemodynamic simulations, which has been demonstrated effective for both forward and inverse UQ problems.[14,15] The theoretical backbone of the BF surrogate is based on the multi-fidelity paradigm, which leverages the accuracy of high-fidelity CFD solutions and the efficiency of low-fidelity (LF) CFD solutions, largely reducing the total computation cost compared with conventional full-order CFD simulations. Here, we compared the proposed DNN-based surrogate with the state-of-the-art BF surrogate model, which are built with the same amount of training labels. It is worth noting that the training overhead and prediction cost of the BF surrogate are always higher than those of the DNN-based surrogate with the same amount of training data (i.e., high-fidelity CFD labels), because low-fidelity CFD simulations have to be performed for both training and inference. Specifically, to build the BF surrogate model, in addition to high-fidelity (HF) CFD training data, another 1000 low-fidelity (LF) CFD simulations are conducted on low-resolution meshes. As the low-res meshes are very coarse and the convergence criterion is set to be large, the LF simulation cost is much lower than that of a HF simulation. The LF data are used to explore the input geometric space to determine important points, where HF CFD simulations are performed for training. More details of the BF surrogate construction can be found in Ref. 14.

To find out how the prediction errors change with the training size, the training set size (TS) changes from 300 to 900, and the testing set is a fixed group of 100 aortas randomly generated. The comparison of performance is shown in Fig. 8. Both surrogate models have reasonably

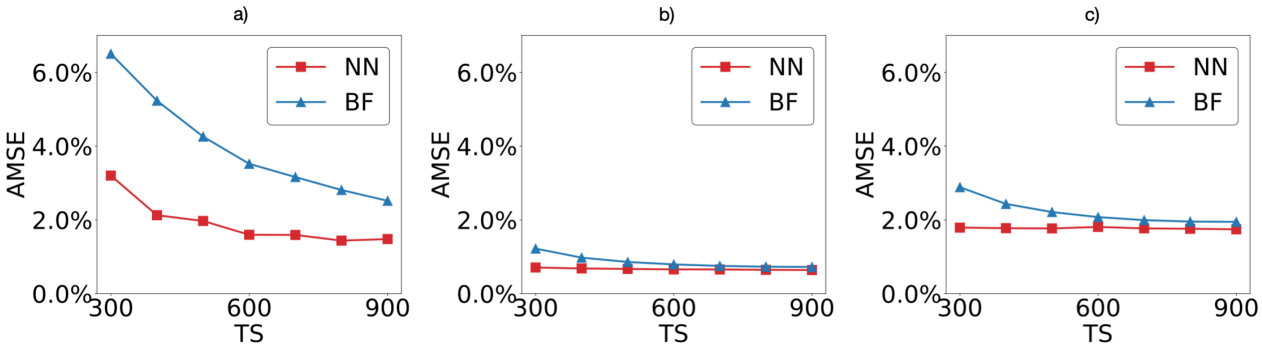

FIG. 8. Comparison between neural network (NN) and bi-fidelity model (BF) for predicted (a) pressure, (b) velocity, and (c) wall shear stress, respectively. Y and X axes denote averaged mean square error (AMSE) and training set size (TS).

good predictions, but the proposed DNN-based surrogate model outperforms the BF surrogate model for all FOIs under all different training scenarios. Both surrogate models reach their best accuracy at $TS = 900$ for all FOIs. The prediction error of the BF model quickly increases as the training sample size decreases, while this trend is less notable for the DNN-based model, particularly for velocity and WSS predictions, indicating a more robust behavior in the small data regime. For both models, the learning performance for the 3D velocity vector field is slightly better than that of surface field outputs, such as pressure and WSS, due to the fact that they are more sensitive to the local changes of the input geometry cross-difference patients. Moreover, the BF prediction cost is higher than the DNN-based surrogate, since each BF prediction requires an LF CFD simulation. Therefore, the proposed DNN-based surrogate model is superior to the BF model in terms of accuracy and efficiency, though both models can predict all FOIs reasonably well.

Besides the bi-fidelity and NN surrogate models compared in this work. There are many other surrogate models one can explore. One most relevant kind of model is the NN-based multi-fidelity model, where either the bi-fidelity element or the NN element is dominant. An example of a bi-fidelity dominant method is the work of Lu and Zhu.[52] They noticed that the LF and HF coefficients are not identical but highly correlated. Therefore, they used a NN to map the relationship between the input (input parameters and LF coefficients) and the output (HF coefficients). Instead of regarding the LF and HF coefficients to be identical in the original bi-fidelity model, this method approximates the difference between LF and HF basis and can improve the prediction accuracy. For NN dominant methods, the NN is approximating the relationship between input parameters and output solution in the high-fidelity space. Then, an LF solution is evaluated and added to the middle layers of the NN. For example, Pepper *et al.*[53] created a knowledge-based neural network (KBaNN) to learn high-fidelity fluid field given input parameters, where NN mapping and LF CFD solution are calculated in parallel and then combined to produce the output. In general, those models can be called as hybrid models, which leverage both the numerical solution and the machine learning. Recently, researchers begin to combine differentiable solvers with NN structure to create differential hybrid surrogate models,[54] which are supposed surpass the performance for mere NN models while still maintain high efficiency.

D. Limitations and future work

1. Geometry parameterization

In this work, we have successfully demonstrated the feasibility of the proposed pipeline (PCA and MLP) focusing on the geometric space for patients suffering from COA. However, the current geometry parameterization has its limitations. For example, the encoded PCA vector changes for operations like translation or rotation. Although we aligned the geometries to a template using the ICP algorithm to avoid this, the orientations of the synthetic training samples are, though very close, different from the template geometry, resulting in a decrease in prediction accuracy. An alternative way is to use graph neural networks (GNNs) for the geometry representation, which is translation- and rotation-invariant and contains more topology information compared to point clouds.

In addition, the LDDMM algorithm in the current pipeline is a surface-based approach and can be extended to large deformations with geometries that have a single channel (e.g., aorta) or one volume (e.g., left atrium). However, for geometries with multiple channels, the LDDMM is no longer applicable as the correspondence cannot be well built. Many works have been done to apply centerline-based parameterization for those cases. For example, Thamsen *et al.*[55] created a synthetic database of aortas with carotid artery to the subclavian artery. However, there is insufficient work on parameterization using a surface-based representation for complicated geometries. In future work, we aim to develop a hierarchical algorithm that can synthesize complex geometries such as aorta with branches.

In addition to PCA, there are various tools to compress high-dimensional data such as auto-encoder. In this work, we choose PCA because it has linearly independent bases and high efficiency. Considering we are establishing the whole surrogate pipeline from scratch in this work, the PCA encoder is sufficient for the current stage. However, more sophisticated encoding tools are necessary for future development since we expect to handle more complicated data in the future. More intuitively, because the fluid quantities are valued on an unstructured mesh, we can build a GNN from the unstructured mesh and implement graph embedding to encode the irregular high-dimensional data.

2. Simulation authenticity

Another important topic is the applicability of the model to situations where realistic flow behaviors (e.g., transient, non-Newtonian, FSI) are modeled. In general, when considering factors such as transient flow, which is crucial for reflection wave analysis, the mapping unit (MLP) will have to be replaced by a recurrent neural network (RNN) such as long short-term memory (LSTM) or gated recurrent unit (GRU). If one encounters gradient loss due to a long time sequence or wants to compute in parallel, a transformer can be used to process the inlet time sequence at one time. For the non-Newtonian assumption of the blood flow, the current MLP can be extended to this situation as long as a specific non-Newtonian model (e.g., the generalized power law model) is used while generating the CFD results as the training data. However, if such model is unknown, one can embed NN into a solver to learn the stress–shear rate relationship from the ground truth data

3. Machine learning

Observing the prediction results in Figs. 6 and 7, the prediction flow predictions are smoother than the ground truth. The incapability of capturing high-frequency components in the flow field is induced by several limitations of the current method: (a) the PCA encoding method reconstructs flow field from truncated primary components, which is more likely to keep averaged values of the fluid fields. (b) The number of training samples is not large enough and the interpolation over sparsely distributed samples is likely to induce underfitting and yield smooth flow fields.

Also we note that the stenosis area at the aortic arch requires the most prediction accuracy and meanwhile has the largest discrepancy in this work. This is because to the current PCA+MLP architecture encodes the geometry and flow field as a whole and uses MLP to fully connect all information in the mapping. As a result, the surrogate model yields good predictions globally and inevitably has less accuracy near the region

of interest. To address this issue, we can represent the geometry with a graph and use graph convolution to learn the relationship between the neighborhood geometric features and the local flow patterns.

4. Uncertainty

In this section, we briefly discuss the uncertainty in the model including epistemic and aleatoric components. The former kind of uncertainty comes from the simplified boundary conditions such as rigid wall assumption, parabolic inlet flow, and constant reference pressure at the outlet; non-Newtonian behavior from the blood flow, whereas the aleatoric uncertainty can come from: noise from image scans such as phase-contrast magnetic resonance angiogram (PC-MRA) scans; human artifacts during image segmentation; randomness in numerical discretization; random distribution of the training samples; and random initialization of the MLP. In terms of factorizing those uncertainties, the current model can only separate part of the resources. For example, we can train the NN multiple times with random parameter initializations and take the average of the learned parameters to remove NN initialization uncertainty from others. We can increase more training samples to suppress the uncertainty from the training sample distribution.

IV. CONCLUSION

In this study, we developed a statistical shape modeling (SSM) approach to parameterize the geometric space spanned by a set of 3D patient aorta geometries and systematically synthesize a large number of virtual aorta geometries. The results show that the proposed SSM method successfully established correspondence among aorta geometries from different patients and the uniform PCA sampling algorithm sufficiently explored the input space, generating abundant synthetic aorta shapes that fully cover the geometric variation. Subsequently, an ML-based surrogate model is proposed to predict comprehensive local hemodynamic information (e.g., surface pressure, velocity, and WSS fields), where an encoding-decoding structure is built based on PCA and multiple MLPs are constructed to learn the non-linear mappings between the input geometries and flow fields of interest within the latent spaces. The PCA encoder and decoders managed to reduce the dimension of the mesh-based 3D aorta shapes and field outputs to a large extent and maintained a reconstruction error as low as 1%. The MLPs are well optimized in terms of their hyperparameters and trained on different sizes of sample sets. The testing results prove that the proposed surrogate model is able to yield hemodynamic field predictions in a good agreement with the ground truth obtained from full-order CFD simulations, yet the accuracy is slightly lower near the stenosis of the descending aorta, where flow features are more complicated due to sharp pressure gradients. The accuracy of the proposed model can be improved by increasing the size of training set. The proposed DNN-based surrogate is compared with the state-of-the-art bi-fidelity (BF) surrogate model, and the DNN surrogate outperforms the BF surrogate in terms of both efficiency and accuracy for all FOIs under different training scenarios. Furthermore, the proposed model exhibits a robust behavior when the number of samples drops, whereas the BF model easily loses its accuracy, especially for surface pressure predictions. In conclusion, the proposed DNN-based surrogate model shows a great capability of approximating the non-linearity of the CFD simulations for cardiovascular flows. Considering extremely low costs of DNN for inference, the trained DNN-based surrogate is able to provide high-resolution local hemodynamic information in split of second, showing a great potential for applications requiring massive model queries, such model inference, optimization, and uncertainty quantification problems. Although the current work focus on aortic flows, the proposed methods can be applied to hemodynamic modeling for other anatomies in general.

ACKNOWLEDGMENTS

The authors would like to acknowledge the funds from National Science Foundation under Award Nos. CMMI-1934300 and OAC-2047127, and ND Lucy Family Institute for Data Science and Society Research Accelerate Funding in supporting this study.

AUTHOR DECLARATIONS

Conflict of Interest

The authors have no conflicts to disclose.

Author Contributions

Pan Du: Formal analysis (equal); Investigation (equal); Methodology (equal); Software (equal); Validation (equal); Visualization (equal); Writing – original draft (equal); Writing – review & editing (equal). **Xiaozhi Zhu:** Data curation (equal); Methodology (supporting); Software (supporting); Writing – review & editing (supporting). **Jian-Xun Wang:** Conceptualization (equal); Funding acquisition (equal); Investigation (equal); Methodology (equal); Project administration (equal); Supervision (equal); Validation (equal); Writing – original draft (equal); Writing – review & editing (lead).

DATA AVAILABILITY

The data that support the findings of this study are available from the corresponding author upon reasonable request.

APPENDIX: OPTIMIZED MLP STRUCTURES AND HYPERPARAMETERS

TABLE V. The optimized MLP structure and hyperparameters based on Bayesian tuning.

FOI	DNN layers	In and out features	Batch size	Learning rate
Pressure	Linear layer	7,64	8	1×10^{-3}
	ReLu	⋯		
	Linear layer	64,8		
	ReLu	⋯		
	Linear layer	8,256		
	ReLu	⋯		
	Linear layer	265,10		
Velocity	Linear layer	7,64	8	4.07×10^{-4}
	ReLu	⋯		
	Linear layer	64,512		
	ReLu	⋯		
	Linear layer	512,256		
	ReLu	⋯		
	Linear layer	265,10		

TABLE V. (*Continued.*)

FOI	DNN layers	In and out features	Batch size	Learning rate
WSS	Linear layer	7,512	8	2.30×10^{-4}
	ReLu	⋯		
	Linear layer	512,16		
	ReLu	⋯		
	Linear layer	16,512		
	ReLu	⋯		
	Linear layer	512,32		
	ReLu	⋯		
	Linear layer	32,16		
	ReLu	⋯		
	Linear layer	16,256		
	ReLu	⋯		
	Linear layer	256,10		

REFERENCES

[1]D. Mozaffarian, E. J. Benjamin, A. S. Go, D. K. Arnett, M. J. Blaha, M. Cushman, S. De Ferranti, J.-P. Després, H. J. Fullerton, V. J. Howard *et al.*, "Heart disease and stroke statistics-2015 update: A report from the American Heart Association," Circulation **131**(4), 434–441 (2015).

[2]K. S. Nayak, J.-F. Nielsen, M. A. Bernstein, M. Markl, P. D. Gatehouse, R. M. Botnar, D. Saloner, C. Lorenz, H. Wen, B. S. Hu *et al.*, "Cardiovascular magnetic resonance phase contrast imaging," J. Cardiovasc. Magn. Reson. **17**(1), 71 (2015).

[3]C. A. Taylor and C. Figueroa, "Patient-specific modeling of cardiovascular mechanics," Annu. Rev. Biomed. Eng. **11**, 109–134 (2009).

[4]D. A. Steinman and C. A. Taylor, "Flow imaging and computing: Large artery hemodynamics," Ann. Biomed. Eng. **33**(12), 1704–1709 (2005).

[5]J. Xiang, L. Antiga, N. Varble, K. V. Snyder, E. I. Levy, A. H. Siddiqui, and H. Meng, "AView: An image-based clinical computational tool for intracranial aneurysm flow visualization and clinical management," Ann. Biomed. Eng. **44**(4), 1085–1096 (2016).

[6]A. Quarteroni, A. Manzoni, and C. Vergara, "The cardiovascular system: Mathematical modelling, numerical algorithms and clinical applications," Acta Numer. **26**, 365–590 (2017).

[7]C. A. Taylor, T. A. Fonte, and J. K. Min, "Computational fluid dynamics applied to cardiac computed tomography for noninvasive quantification of fractional flow reserve: Scientific basis," J. Am. Coll. Cardiol. **61**(22), 2233–2241 (2013).

[8]K. M. Chinnaiyan, T. Akasaka, T. Amano, J. J. Bax, P. Blanke, B. De Bruyne, T. Kawasaki, J. Leipsic, H. Matsuo, Y. Morino *et al.*, "Rationale, design and goals of the heartflow assessing diagnostic value of non-invasive FFR_{CT} in coronary care (ADVANCE) registry," J. Cardiovasc. Comput. Tomogr. **11**(1), 62–67 (2017).

[9]T. W. Hokken, J. M. Ribeiro, P. P. De Jaegere, and N. M. Van Mieghem, "Precision medicine in interventional cardiology," Interventional Cardiol. Rev. **15**, e03 (2020).

[10]S. Sankaran and A. L. Marsden, "A stochastic collocation method for uncertainty quantification and propagation in cardiovascular simulations," J. Biomech. Eng. **133**(3), 031001 (2011).

[11]S. Sankaran and A. L. Marsden, "The impact of uncertainty on shape optimization of idealized bypass graft models in unsteady flow," Phys. Fluids **22**(12), 121902 (2010).

[12]J. S. Tran, D. E. Schiavazzi, A. B. Ramachandra, A. M. Kahn, and A. L. Marsden, "Automated tuning for parameter identification and uncertainty quantification in multi-scale coronary simulations," Comput. Fluids **142**, 128–138 (2017).

[13]D. Schiavazzi, A. Doostan, G. Iaccarino, and A. Marsden, "A generalized multi-resolution expansion for uncertainty propagation with application to cardiovascular modeling," Comput. Methods Appl. Mech. Eng. **314**, 196–221 (2017).

[14]H. Gao and J.-X. Wang, "A bi-fidelity ensemble kalman method for PDE-constrained inverse problems in computational mechanics," Comput. Mech. **67**(4), 1115–1131 (2021).

[15]H. Gao, X. Zhu, and J.-X. Wang, "A bi-fidelity surrogate modeling approach for uncertainty propagation in three-dimensional hemodynamic simulations," Comput. Methods Appl. Mech. Eng. **366**, 113047 (2020).

[16]M. Mirramezani, S. L. Diamond, H. I. Litt, and S. C. Shadden, "Reduced order models for transstenotic pressure drop in the coronary arteries," J. Biomech. Eng. **141**(3), 031005 (2019).

[17]M. Mirramezani and S. C. Shadden, "A distributed lumped parameter model of blood flow," Ann. Biomed. Eng. **48**(12), 2870–2886 (2020).

[18]M. R. Pfaller, J. Pham, A. Verma, N. M. Wilson, D. W. Parker, W. Yang, and A. L. Marsden, "Automated generation of 0D and 1D reduced-order models of patient-specific blood flow," arXiv:2111.04878 (2021).

[19]A. Manzoni, A. Quarteroni, and G. Rozza, "Model reduction techniques for fast blood flow simulation in parametrized geometries," Int. J. Numer. Methods Biomed. Eng. **28**(6–7), 604–625 (2012).

[20]T. Lassila, A. Manzoni, A. Quarteroni, and G. Rozza, "A reduced computational and geometrical framework for inverse problems in hemodynamics," Int. J. Numer. Methods Biomed. Eng. **29**(7), 741–776 (2013).

[21]F. Ballarin and G. Rozza, "POD–Galerkin monolithic reduced order models for parametrized fluid-structure interaction problems," Int. J. Numer. Methods Fluids **82**(12), 1010–1034 (2016).

[22]F. Ballarin, E. Faggiano, S. Ippolito, A. Manzoni, A. Quarteroni, G. Rozza, and R. Scrofani, "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. **315**, 609–628 (2016).

[23]T. Lassila, A. Manzoni, A. Quarteroni, and G. Rozza, "Model order reduction in fluid dynamics: Challenges and perspectives," *Reduced Order Methods Modeling Computational Reduction* (Springer, 2014), pp. 235–273.

[24]S. Chaturantabut and D. C. Sorensen, "Nonlinear model reduction via discrete empirical interpolation," SIAM J. Sci. Comput. **32**(5), 2737–2764 (2010).

[25]H. Gao, J.-X. Wang, and M. J. Zahr, "Non-intrusive model reduction of large-scale, nonlinear dynamical systems using deep learning," Physica D **412**, 132614 (2020).

[26]F. Scarselli and A. C. Tsoi, "Universal approximation using feedforward neural networks: A survey of some existing methods, and some new results," Neural Networks **11**(1), 15–37 (1998).

[27]M. Hutzenthaler, A. Jentzen, T. Kruse, T. Anh Nguyen, and P. von Wurstemberger, "Overcoming the curse of dimensionality in the numerical approximation of semilinear parabolic partial differential equations," Proc. R. Soc. A **476**(2244), 20190630 (2020).

[28]L. Liang, M. Liu, C. Martin, J. A. Elefteriades, and W. Sun, "A machine learning approach to investigate the relationship between shape features and numerically predicted risk of ascending aortic aneurysm," Biomech. Model. Mechanobiol. **16**(5), 1519–1533 (2017).

[29]L. Liang, M. Liu, C. Martin, and W. Sun, "A deep learning approach to estimate stress distribution: A fast and accurate surrogate of finite-element analysis," J. R. Soc. Interface **15**(138), 20170844 (2018).

[30]A. Madani, A. Bakhaty, J. Kim, Y. Mubarak, and M. R. Mofrad, "Bridging finite element and machine learning modeling: Stress prediction of arterial walls in atherosclerosis," J. Biomech. Eng. **141**(8), 084502 (2019).

[31]A. Balu, S. Nallagonda, F. Xu, A. Krishnamurthy, M.-C. Hsu, and S. Sarkar, "A deep learning framework for design and analysis of surgical bioprosthetic heart valves," Sci. Rep. **9**(1), 18560 (2019).

[32]F. Sahli Costabal, Y. Yang, P. Perdikaris, D. E. Hurtado, and E. Kuhl, "Physics-informed neural networks for cardiac activation mapping," Front. Phys. **8**, 42 (2020).

[33]G. Kissas, Y. Yang, E. Hwuang, W. R. Witschey, J. A. Detre, and P. Perdikaris, "Machine learning in cardiovascular flows modeling: Predicting arterial blood pressure from non-invasive 4D flow MRI data using physics-informed neural networks," Comput. Methods Appl. Mech. Eng. **358**, 112623 (2020).

[34]L. Liang, W. Mao, and W. Sun, "A feasibility study of deep learning for predicting hemodynamics of human thoracic aorta," J. Biomech. **99**, 109544 (2020).

[35]G. Li, H. Wang, M. Zhang, S. Tupin, A. Qiao, Y. Liu, M. Ohta, and H. Anzai, "Prediction of 3D cardiovascular hemodynamics before and after coronary artery bypass surgery via deep learning," Commun. Biol. **4**(1), 99 (2021).
[36]J. Brüning, F. Hellmeier, P. Yevtushenko, T. Kühne, and L. Goubergrits, "Uncertainty quantification for non-invasive assessment of pressure drop across a coarctation of the aorta using CFD," Cardiovasc. Eng. Technol. **9**(4), 582–596 (2018).
[37]S. Valette and J. Chassery, "Approximated centroidal Voronoi diagrams for uniform polygonal mesh coarsening," Comput. Graphics Forum **23**(3), 381–389 (2004).
[38]S. Bouaziz, A. Tagliasacchi, and M. Pauly, "Sparse iterative closest point," Comput. Graphics Forum **32**, 113–123 (2013).
[39]M. F. Beg, M. I. Miller, A. Trouvé, and L. Younes, "Computing large deformation metric mappings via geodesic flows of diffeomorphisms," Int. J. Comput. Vision **61**(2), 139–157 (2005).
[40]S. Durrleman, M. Prastawa, N. Charon, J. R. Korenberg, S. Joshi, G. Gerig, and A. Trouvé, "Morphometry of anatomical shape complexes with dense deformations and sparse parameters," NeuroImage **101**, 35–49 (2014).
[41]L. Antiga, M. Piccinelli, L. Botti, B. Ene-Iordache, A. Remuzzi, and D. A. Steinman, "An image-based modeling framework for patient-specific computational hemodynamics," Med. Biol. Eng. Comput. **46**(11), 1097–1112 (2008).
[42]H. G. Weller, G. Tabor, H. Jasak, and C. Fureby, "A tensorial approach to computational continuum mechanics using object-oriented techniques," Comput. Phys. **12**(6), 620–631 (1998).
[43]P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright *et al.*, "Scipy 1.0: Fundamental algorithms for scientific computing in python," Nat. Methods **17**(3), 261–272 (2020).
[44]R. Liaw, E. Liang, R. Nishihara, P. Moritz, J. E. Gonzalez, and I. Stoica, "Tune: A research platform for distributed model selection and training," arXiv:1807.05118 (2018).
[45]L. Li *et al.*, "A system for massively parallel hyperparameter tuning," in *Proceedings of Machine Learning and Systems* (MLSys, 2020), Vol. **2**, pp. 230–246.
[46]R. H. Pletcher, J. C. Tannehill, and D. Anderson, *Computational Fluid Mechanics and Heat Transfer* (CRC Press, 2012).
[47]C. M. Rhie and W.-L. Chow, "Numerical study of the turbulent flow past an airfoil with trailing edge separation," AIAA J. **21**(11), 1525–1532 (1983).
[48]M. C. Kennedy and A. O'Hagan, "Bayesian calibration of computer models," J. R. Stat. Soc., Ser. B **63**(63), 425–464 (2001).
[49]J. Biehler, M. W. Gee, and W. A. Wall, "Towards efficient uncertainty quantification in complex and large-scale biomechanical problems based on a Bayesian multi-fidelity scheme," Biomech. Model. Mechanobiol. **14**(3), 489–513 (2015).
[50]J. Biehler, S. Kehl, M. W. Gee, F. Schmies, J. Pelisek, A. Maier, C. Reeps, H.-H. Eckstein, and W. A. Wall, "Probabilistic noninvasive prediction of wall properties of abdominal aortic aneurysms using Bayesian regression," Biomech. Model. Mechanobiol. **16**(1), 45–61 (2017).
[51]C. M. Fleeter, G. Geraci, D. E. Schiavazzi, A. M. Kahn, and A. L. Marsden, "Multilevel and multifidelity uncertainty quantification for cardiovascular hemodynamics," Comput. Methods Appl. Mech. Eng. **365**, 113030 (2020).
[52]C. Lu and X. Zhu, "Bifidelity data-assisted neural networks in nonintrusive reduced-order modeling," J. Sci. Comput. **87**(1), 8 (2021).
[53]N. Pepper, A. Gaymann, S. Sharma, and F. Montomoli, "Local bi-fidelity field approximation with knowledge based neural networks for computational fluid dynamics," Sci. Rep. **11**(1), 14459 (2021).
[54]F. D. A. Belbute-Peres, T. Economon, and Z. Kolter, "Combining differentiable PDE solvers and graph neural networks for fluid flow prediction," in *International Conference on Machine Learning* (PMLR, 2020), pp. 2402–2411.
[55]B. Thamsen, P. Yevtushenko, L. Gundelwein, A. A. A. Setio, H. Lamecker, M. Kelm, M. Schafstedde, T. Heimann, T. Kühne, and L. Goubergrits, "Synthetic database of aortic morphometry and hemodynamics: Overcoming medical imaging data availability," IEEE Trans. Med. Imaging **40**(5), 1438–1449 (2021).