

Reducing Geometric Uncertainty in Computational Hemodynamics by Deep Learning-Assisted Parallel-Chain MCMC

Pan Du^a, Jian-Xun Wang^{a,*}

^a*Department of Aerospace and Mechanical Engineering, University of Notre Dame, Notre Dame, IN*

Abstract

Computational hemodynamic modeling has been widely used in cardiovascular research and healthcare. However, the reliability of model predictions is largely dependent on the uncertainties of modeling parameters and boundary conditions, which should be carefully quantified and further reduced with available measurements. In this work, we focus on propagating and reducing the uncertainty of vascular geometries within a Bayesian framework. A novel deep learning (DL)-assisted parallel Markov chain Monte Carlo (MCMC) method is presented to enable efficient Bayesian posterior sampling and geometric uncertainty reduction. A DL model is built to approximate the geometry-to-hemodynamic map, which is trained actively using online data collected from parallel MCMC chains and utilized for early rejection of unlikely proposals to facilitate convergence with less expensive full-order model evaluations. Numerical studies on 2-D aortic flows are conducted to demonstrate the effectiveness and merit of the proposed method.

Keywords: Computational fluid dynamics, Cardiovascular modeling, Machine learning, Uncertainty quantification, Model inference

1. Introduction

Hemodynamics information, e.g., fractional flow reserve (FFR), wall shear stress, and blood flow pattern, is critical in cardiovascular research and personalized healthcare [1–3]. To quantify the functional information of hemodynamics from medical images, e.g., com-

*Corresponding author. Tel: +1 574-631-5302
 Email address: jwang33@nd.edu (Jian-Xun Wang)

puted tomography (CT) and magnetic resonance imaging (MRI), image-based computational fluid dynamics (CFD) modeling has become a paradigm in cardiovascular study [1]. The standard workflow of constructing an image-based model includes (1) extraction of vascular anatomy from images as the geometric boundary, (2) numerical discretization and parameter specification, (3) solving discretized partial differential equation (PDE) system, and (4) post-processing simulated results. In this whole process, uncertainties can be introduced from various sources, e.g., reconstructed geometries, inflow/outflow boundary conditions, and specified mechanical properties, which largely impact the reliability of the simulated solutions [4–6]. Quantifying and reducing these uncertainties in computational hemodynamics have begun to draw attention in recent years [7–15]. Most of the existing literature has focused on forward uncertainty propagation, i.e., investigating the sensitivity of the simulated hemodynamics to the modeling inputs. For example, Sankaran and Marsden [14] developed an adaptive collocation polynomial chaos expansion (PCE) method with sparse grids to forward propagate input uncertainties on several cardiovascular modeling problems. Fleeter et al. [16] proposed a multi-fidelity stochastic framework to estimate propagated uncertainties for both global and local hemodynamic indicators. Guzzetti et al. [9] integrated a transverse enriched pipe element reduced order model into a PCE framework for efficient uncertainty quantification in large-scale hemodynamics models.

When observations of the system outcomes are available, which might be indirect, sparse, and noisy, the uncertainty of the modeling inputs and simulated states can be reduced within a Bayesian framework, known as inverse uncertainty quantification (UQ). The general idea is to describe the data under uncertainties in the form of likelihood functions and then compute the posterior distributions of unknown modeling inputs and simulated states of interest, given specified priors. Inverse UQ is more meaningful as it enables the assimilation of imperfect data in a probabilistic way for model inference and uncertainty reduction, which has attracted increasing attention in the cardiovascular hemodynamic community [17–20]. Although there are many different ways to solve Bayesian data assimilation problems, e.g., variational inference [21], sequential or iterative Kalman inversion [18, 19], ensemble

average approaches [22], etc, they can only approximate the posterior statistics to certain extents. In order to accurately quantify the posterior distribution, Markov Chain Monte Carlo (MCMC) sampling is still the gold standard for Bayesian computing and uncertainty quantification [23]. MCMC only requires point-wise evaluations of the posterior density up to its normalization constant, making the algorithm code non-intrusive and easy to implement. However, an intrinsic limitation of the MCMC algorithm lies in its inherent sequential nature and the requirement of a huge sample size. This limitation makes it computationally infeasible for applications involving expensive computer models, e.g., computational hemodynamic simulations, since the MCMC typically requires hundreds of thousands (or more) samples to reach the convergence, each of which requires a CFD forward simulation that has to be done sequentially.

To alleviate the computational burden, one way is to replace the expensive CFD model with an efficient surrogate, which can be constructed based on, e.g., radial basis function [24], Gaussian process (or Kriging) [25], generalized polynomial chaos (PCE) [26], or multi-fidelity models [20]. However, traditional surrogate modeling techniques, e.g., Gaussian process (GP), can only handle problems with moderate dimensions, and thus the surrogate maps are often built for predictions of global quantities instead of local hemodynamics (e.g., velocity, pressure fields). In this work, we adopt deep learning (DL) for surrogate modeling of local hemodynamics as it has been demonstrated to be capable of approximating high-dimensional operators accurately [27, 28]. Although surrogate-based MCMC makes the Bayesian computing tractable, the approximation errors introduced by the surrogate may distort the computed posterior distribution. As an alternative, we leverage the DL-based surrogate modeling in a different way: instead of replacing the CFD with the surrogate emulator, we use the DL-based surrogate to accelerate the MCMC convergence by rejecting proposals at an early stage without running the expensive CFD simulations. The ideas of delayed acceptance have been explored previously in other contexts either using unconverged solutions [29, 30] or GP-based emulators [31, 32], showing asymptotically exact sampling behavior at the cost of potential speedup by requiring full-order model evaluation for each

accepted sample. To further improve the efficiency, making MCMC sampling in parallel is desirable. In the past decade, many different parallelizable MCMC algorithms have been proposed, which can be classified into three categories: (1) embarrassingly parallel MCMC (EP-MCMC) that splits the dataset into multiple partitions and sample the corresponding sub-posteriors in parallel [33–35], (2) multiple-proposal MCMC (MP-MCMC) that proposes multiple samples simultaneously in every iteration while evaluates their densities in parallel [36–39], and (3) parallel-chain MCMC (PC-MCMC) that runs multiple MCMC chains in parallel [40–43]. In this work, we formulated a DL-assisted delayed acceptance scheme within the parallel-chain MCMC framework based on an active learning strategy for geometric uncertainty reduction in computational hemodynamics. In particular, a deep neural network (DNN)-based surrogate model capturing local hemodynamics given geometric inputs is constructed and trained during MCMC sampling in an online manner. The trained DNN-surrogate will be used for the delayed acceptance to significantly improve the convergence. Moreover, multiple concurrent chains collaboratively provide sufficient training samples to actively improve the DNN surrogate during MCMC sampling, enabling a novel DL-assisted MCMC parallelism. The rest of the paper is organized as follows. The proposed DL-assisted parallel-chain MCMC algorithm is introduced in Section 2. Numerical results for geometric uncertainty reduction in aortic flows in 2-D irregular geometries are presented and discussed in Section 3. Finally, Section 4 concludes the paper.

2. Methodology

2.1. Overview

We present a novel Bayesian MCMC computing approach assisted by active deep learning (DL) to estimate and reduce input uncertainties in computational hemodynamic simulations based on noisy velocity measurements. Here, we restrict the discussion to the impact of the uncertainty of the input geometry, which is usually due to imaging segmentation errors and has been demonstrated as the most dominant uncertainty source in the computational hemodynamics [44, 45]. Specifically, the prior uncertainty is defined based on the parameterization of the geometric variations and will be updated to the posterior by assimilating

noisy velocity data via Bayes rule using the MCMC sampling. We leverage a DL-based geometry-to-flow surrogate to facilitate the Metropolis-Hasting MCMC convergence by rejecting the proposed samples early to avoid a large number of expensive CFD simulations for the likelihood evaluations. The DL surrogate is actively trained (i.e., iteratively improved) using newly generated CFD data, gradually collected from multiple parallel MCMC chains in an online manner. Note that the major focus of this work is algorithmic development of the method that reduces the uncertainty under a Bayesian framework given sparse, noisy measurements instead of clinical applications of aortic flows. Hence, we demonstrate the feasibility of the algorithm on simplified 2-D aorta geometry for proof-of-concept, which will be introduced in the following section.

2.2. Parametric 2-D aorta model

We use a 2-D aorta model to demonstrate the proposed inverse geometric UQ method. The parameterization of the aorta geometry is shown in Figure 1. To start with, a 3-D patient-specific aorta surface from [46] is reconstructed from medical images and pre-processed using the Vascular Modeling Toolkit (VMTK), a popular open-source module for 3-D geometric reconstruction, analysis and mesh generation. For simplicity, the top branches such as subclavian and carotid arteries are trimmed off, leaving a single-channel vessel with its surface wall smoothed and outlet extended to facilitate convergence in CFD simulations. The centerline of the vessel is extracted (highlighted in red color) from the aorta surface, and both the aorta surface and its centerline are projected to a flat plane, the orientation of which is determined by optimization to mostly preserve the side shape characteristics of the aorta. In particular, we take the centerline of the 3D aorta geometry and project points $\{\mathbf{p}_i\}_{i=1,\dots,m}$ (m denotes the total number of points on the centerline) of the centerline to the normal vector $\mathbf{n}$ of the projection plane. The projected coordinates can be expressed as $\{p'_i\} = \{\langle \mathbf{n}, \mathbf{p} \rangle_i\}$, where $\langle \cdot, \cdot \rangle$ is regular Euclidean dot product. We minimize the variance of $\{p'_i\}$ to obtain the optimized normal vector $\mathbf{n}_{opt}$, which in this case, is $[0.88, 0.43, 0.22]^T$. A sequence of eleven equally-spaced locations are picked on the centerline along the streamwise direction, including the inlet to the outlet, where normal vectors (perpendicular to the

tangent direction) are calculated. The geometric space is spanned by the centerline and radii at these locations. Hence, given a centerline C and a set of finite radius values $\{r_i\}_{i=1,2,\dots,11}$, one can easily obtain a new aorta geometry, where the surfaces are obtained based on the spline function with a specified resolution. To model the geometric uncertainty, we add a perturbation of the radius as a function of the centerline locations. We design a perturbation method that mimics the shape of an aorta with stenosis or aneurysm at the sixth location of the aorta. We set the degree of freedom (number of elements) of the perturbation vector as three, which is sufficient enough for this designing purpose and higher numbers are not considered because they reduce the efficiency of the MCMC algorithm. We assign one DOF to the sixth location to enable sharp curvature changes. Then we correlate the deformation of the upstream locations by perturbing them together in a sense of maintaining the topology of the whole upstream section during deformation, which can avoid creating unrealistic aorta shapes (e.g., extremely wavy vessel walls). Same for the downstream locations. Finally, we define a three-element perturbation vector $\boldsymbol{\theta} = \{\theta_1, \theta_2, \theta_3\}$, which are applied to the first five $\{r_i^m\}_{i=1,2,\dots,5}$ (in blue), the sixth r_6^m (in green), and the rest radii $\{r_i^m\}_{i=7,8,9,10,11}$ (in purple), respectively. Each component of $\boldsymbol{\theta}$ is drawn from a uniform distribution $\theta_i \sim \mathcal{U}(-\Delta r, \Delta r)$ as the prior, where $\Delta r = 2.57\text{mm}$ denotes the variation limit.

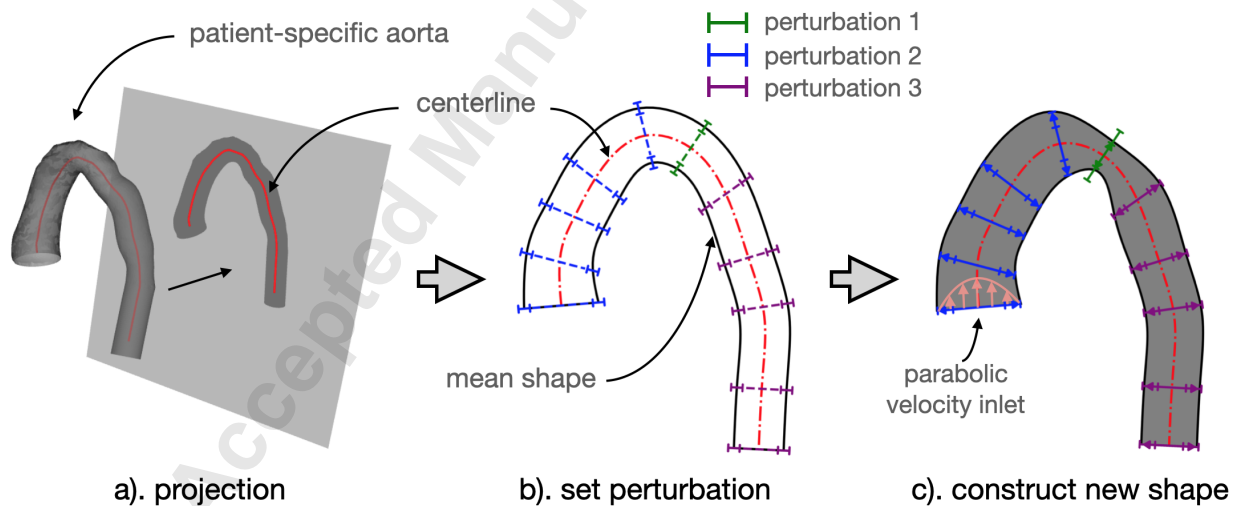


Figure 1: Illustration of the 2-D aorta model construction and shape parameterization

To systematically generate a significant amount of CFD data for training and validation purposes, we build a python routine that fully automates the tedious simulation procedure, including shape construction (mentioned above), mesh generation, boundary settings, simulation and post-processing. Specifically, we develop an in-house mesh generation software based on a constrained Delaunay triangulation algorithm [47]. The triangulated mesh will be automatically generated given the outline of the aorta shape, where the boundary mesh is obtained by extruding the internal ones along the normal direction with one unit length. The generated vtk mesh files are converted into the polyMesh format, which can be used for CFD simulations in OpenFOAM, an open-source CFD platform. The incompressible, steady Navier-Stokes equations are solved based on the Semi-Implicit Method for Pressure Linked Equations (SIMPLE) algorithm [48]. A parabolic velocity profile is prescribed at the inlet with a maximum velocity $U_m = 1m/s$, constant pressure outflow boundary condition is imposed at the outlet, and non-slip wall boundary condition is applied to the aorta walls, assumed to be rigid. A python wrapper is developed to enable the automation of CFD simulations in parallel. During the post-processing step, the simulated hemodynamic field data are projected onto the surface or volumetric mesh grids, where the correspondences are established by uniformly sampling points between the centerline and the surface vertices.

2.3. Fundamentals of MCMC Bayesian sampling

Given some (sparse) flow observations $\mathbf{d}$, we can infer a more accurate vessel shape and reduce the geometric uncertainty within a Bayesian framework,

$$p(\boldsymbol{\theta} | \mathbf{d}) \propto L(\boldsymbol{\theta} | \mathbf{d})p(\boldsymbol{\theta}), \quad (1)$$

where $\boldsymbol{\theta}$ is the parameterization of the geometric variation; $p(\boldsymbol{\theta})$ denotes the prior distribution of the geometry perturbation, which is often due to the image segmentation uncertainty and modeled as a uniform distribution in this work; $L(\boldsymbol{\theta} | \mathbf{d})$ is known as the likelihood function, which is often determined by the measurement uncertainty. In most cases, the posterior density $p(\boldsymbol{\theta} | \mathbf{d})$ can not be solved analytically and has to be obtained by Monte Carlo sampling. Markov Chain Monte Carlo (MCMC) sampling is the gold standard method for

Bayesian computing, whose general idea is to efficiently sample the posterior using a Markov chain. In particular, it starts from an initial sample $\theta_0 \in \Omega$ and explores the parameter space Ω using a transition kernel $T(\theta_i | \theta_{i+1})$. Typically, the transition kernel is computed by two steps: (i) propose a new sample θ_{i+1} based on the current θ_i using a proposal distribution $q(\theta_{i+1} | \theta_i)$ and (ii) determine whether or not accept the proposed sample based on the acceptance probability,

$$A(\theta_{i+1} | \theta_i) = \min \left\{ \frac{p(\theta_{i+1} | \mathbf{d})q(\theta_i | \theta_{i+1})}{p(\theta_i | \mathbf{d})q(\theta_{i+1} | \theta_i)}, 1 \right\}, \quad (2)$$

which is known as the classic Metropolis-Hasting MCMC algorithm. To accurately recover the posterior distribution, the Markov chain is supposed to run for enough iterations until sufficient samples are collected and the posterior landscape is fully explored. In each iteration, we need to calculate the likelihood density, which involves the evaluation of the forward model $\mathcal{F}(\cdot)$, as the measurement model is defined as,

$$\mathbf{d} = h(\mathcal{F}(\theta)) + \sigma_d, \quad (3)$$

where $h(\cdot)$ represents the state-to-observable map, and σ_d denotes the observation uncertainty. The computational cost of each forward model evaluation (i.e., CFD simulation) is nontrivial and can be significantly large in many cases (e.g., 3-D hemodynamics with flexible walls). It is computationally infeasible to conduct enormous forward simulations for MCMC, especially considering the sequential nature of the classic Metropolis-Hasting algorithm, making the matter worse.

2.4. Delayed-acceptance via DNN surrogate

To tackle the challenge of applying MCMC to problems involving expensive simulation, a straightforward remedy is to replace the costly forward model with an efficient surrogate emulator, which may risk distorting the posterior due to surrogate modeling errors. As an alternative, delayed-acceptance (DA) strategy can asymptotically reach the convergence by combining both the full-order and surrogate models for MCMC sampling. The general idea is to leverage the surrogate for screening of most unlikely proposals to avoid unnecessary full-order model (FOM) evaluations. Only for those proposals likely to be accepted will involve

expensive FOM evaluations. Let $\pi(\boldsymbol{\theta}) = p(\boldsymbol{\theta} | \mathbf{d})$ represent the posterior density obtained by the FOM. The DA Metropolis-Hastings algorithm consists of two steps: (i) a pre-accept step is performed using the posterior density $\tilde{\pi}(\boldsymbol{\theta})$ approximated based on the surrogate model, and the acceptance probability is

$$\tilde{A}(\boldsymbol{\theta}_{i+1} | \boldsymbol{\theta}_i) := \min \left\{ \frac{\tilde{\pi}(\boldsymbol{\theta}_{i+1})q(\boldsymbol{\theta}_i | \boldsymbol{\theta}_{i+1})}{\tilde{\pi}(\boldsymbol{\theta}_i)q(\boldsymbol{\theta}_{i+1} | \boldsymbol{\theta}_i)}, 1 \right\}. \quad (4)$$

(ii) if the proposal is accepted by $\tilde{A}$, the final accept/reject decision will be made by the FOM based on the acceptance ratio A^* ,

$$A^*(\boldsymbol{\theta}_{i+1} | \boldsymbol{\theta}_i) := \min \left\{ \frac{\pi(\boldsymbol{\theta}_{i+1})\tilde{\pi}(\boldsymbol{\theta}_i)}{\pi(\boldsymbol{\theta}_i)\tilde{\pi}(\boldsymbol{\theta}_{i+1})}, 1 \right\} \quad (5)$$

Note that the product of the probabilities of these two sequential steps equals to acceptance probability in the original Metropolis-Hasting algorithm (i.e., $A = \tilde{A}A^*$), implying that the detail balance is still satisfied and it will eventually converge to the target distribution. A deep learning based-surrogate model is built in an encoding-decoding fashion to rapidly predict hemodynamics given aorta shapes. In particular, we encode the 2-D aorta shape into a 3-D parameter vector $\boldsymbol{\theta}$ as described above. Meanwhile, the velocity fields computed from CFD simulations are also encoded into the latent space using primary component analysis (PCA). A total of 98% energy is preserved, resulting a decoding error less than 1%. The mapping from the encoded shape to the flow field is learned by a multilayer perceptron (MLP). The MLP uses Rectified Linear Unit (ReLU) as the activation function and is trained using Adam optimizer where the learning rate is adaptively changed based on the estimates of the moments. In order to achieve the best performance, we optimize the hyperparameters of the MLP, including the number of layers, number of neurons of each layer, batch size and initial learning rate based on the Bayesian optimization. The Asynchronous Successive Halving Algorithm (ASHA) is utilized to aggressively terminate non-ideal trials in advance to facilitate convergence. The network hyperparameter optimization process is implemented in RAY-tune [49], which is an open-source python package designed for scalable parameter tuning. The optimized parameters are summarized as: batch size = 8, number of layers = 4, initial learning rate = 6.5×10^{-5} , and the optimized MLP structure is given in Table A.3.

To evaluate the performance of the DL-based surrogate model, the relative mean square error (RMSE) is used to calculate the difference between the surrogate prediction and CFD reference,

$$RMSE = \frac{\sum_i^N (x_i - y_i)^2}{\sum_i^N y_i^2} \times 100\%, \quad (6)$$

where $\mathbf{X} = (x_1, \dots, x_N)$ and $\mathbf{Y} = (y_1, \dots, y_N)$ are prediction and reference on N vertices/grids.

2.5. Active learning with parallel MCMC chains

The DA strategy may fail when there is a large discrepancy between the surrogate and FOM predictions due to a significant decrease in the acceptance ratio. On the other hand, the FOM will be evaluated anyway in the DA-MCMC process for the proposed samples accepted by the surrogate model. Therefore, the natural idea is to leverage these gradually accumulated FOM solutions as data to train the DL surrogate online, known as the active learning. Specifically, during each MCMC iteration, when a new proposal passes the first step of surrogate screening, one FOM propagation will be executed, producing one high-fidelity (HF) flow data. These newly generated HF CFD solutions will be collected to enrich existing training and testing sets. The current DL surrogate will be tested on the constantly-updated test set as long as a new group of N_{new} data are collected. If the testing error exceeds the threshold of ϵ_{test}^* , the newly collected data will be added into the existing training set, which will be used to refine the DL surrogate model actively. Note that the neural network does not need to be re-trained from scratch. Instead, a transfer learning strategy can be used to enable fast online model refinement. For example, the trained parameters of the current MLP are saved and can be reloaded as the initial condition or even frozen to a certain extent when the surrogate model needs to be further re-trained due to the addition of training data. In this way, the training cost of the online update is extensively reduced. The details of the proposed algorithm and pseudo code are given in Algorithm 1.

Active online training brings two major merits: i) During initialization of the DL surrogate, data are collected uniformly over the entire parameter space, which may result in insufficiency near high-density regions. However, online training data collected from MCMC iterations are more likely to locate in high-density regions, and thus the surrogate can be

refined in the area of interest. ii). When multiple MCMC chains run simultaneously (i.e., parallel-chain MCMC), it brings no additional merits other than leveraging multiple CPU cores for classic Metropolis-Hastings or its DA variants. However, active learning-based DA Metropolis-Hastings algorithms can further leverage the parallel-chain setting, where a single surrogate can be updated over the new data collected across multiple chains, facilitating surrogate model update and convergence. Suppose the posterior distribution has more than one mode, and the parallel MCMC chains may move in different high-density regions simultaneously. The surrogate model will quickly improve over multiple high-density regions, resulting in a vast improvement in the UQ performance.

3. Numerical Results

3.1. Surrogate CFD modeling of 2D aortic flows

The variation range of each input shape parameter $\theta_i \in \boldsymbol{\theta}$ is normalized to $[-1, 1]$ and a uniform distribution is given as the prior (i.e., $\theta_i \sim \mathbf{U}(-1, 1), \forall i \in \{1, 2, 3\}$). In order to fully explore the input parameter space, we draw 10000 samples and create corresponding aorta meshes and solution fields using the developed python routine. Starting from the shape parameter $\boldsymbol{\theta}$, each python process, including construction outline, meshing, simulation and post-processing, takes around 6 seconds on a single core on Intel(R) Xeon(R) Gold 6138 CPU. The dataset is generated by conducting a batch of 40 cases in parallel at one time. After the generation of the dataset, the velocity field data is projected into latent space using PCA. The fitted PCA transfer function is saved and serves as a transformation tool between physical and the encoded velocity data. Utilizing RAY-tune tool, the training and MLP hyperparameters are fully optimized. Subsequently, the MLP is trained on training sets of different sizes and tested on a test set of 100 unseen samples to investigate the sensitivity of the surrogate testing error with respect to training size. Each training process is conducted on an Nvidia RTX A6000 GPU with a minimum of 1500 epochs to ensure convergence.

The model testing results are shown in Figure 2, where ϵ_V and ϵ_{VM} represent the RMSE error of the velocity vector field and the velocity magnitude field. As expected, the results

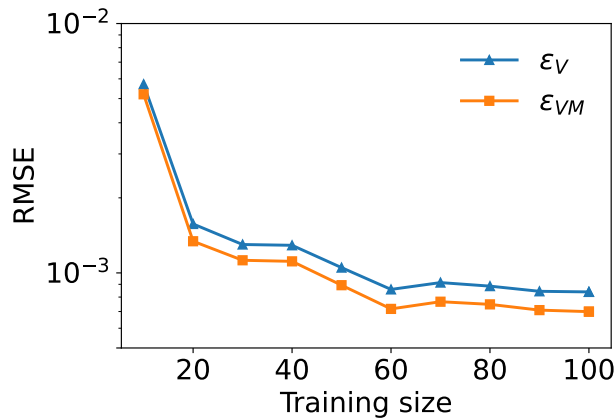


Figure 2: DNN prediction errors (averaged MSE) versus different training sizes. Testing errors of the interior velocity vector field ϵ_V and velocity magnitude ϵ_{VM}

show a decrease for both errors as the dataset size grows. The error of the predicted velocity vector is slightly higher than that of the velocity magnitude prediction. The DL model trained on the entire CFD dataset ($N = 10000$) has a test error of 5.1×10^{-5} . In percentage, the error equals 0.005% in terms of ARMSE of the velocity magnitude, which is very small, indicating a great consistency between the surrogate model and the CFD model. The velocity contours are shown in Figure 3, where we randomly select five test aorta geometries, and the hemodynamic fields predicted by the DL model are compared with the CFD reference (referred to as ground truth). From comparison one can find out that the predicted velocity contours agree well with the CFD ground truth across all selected samples.

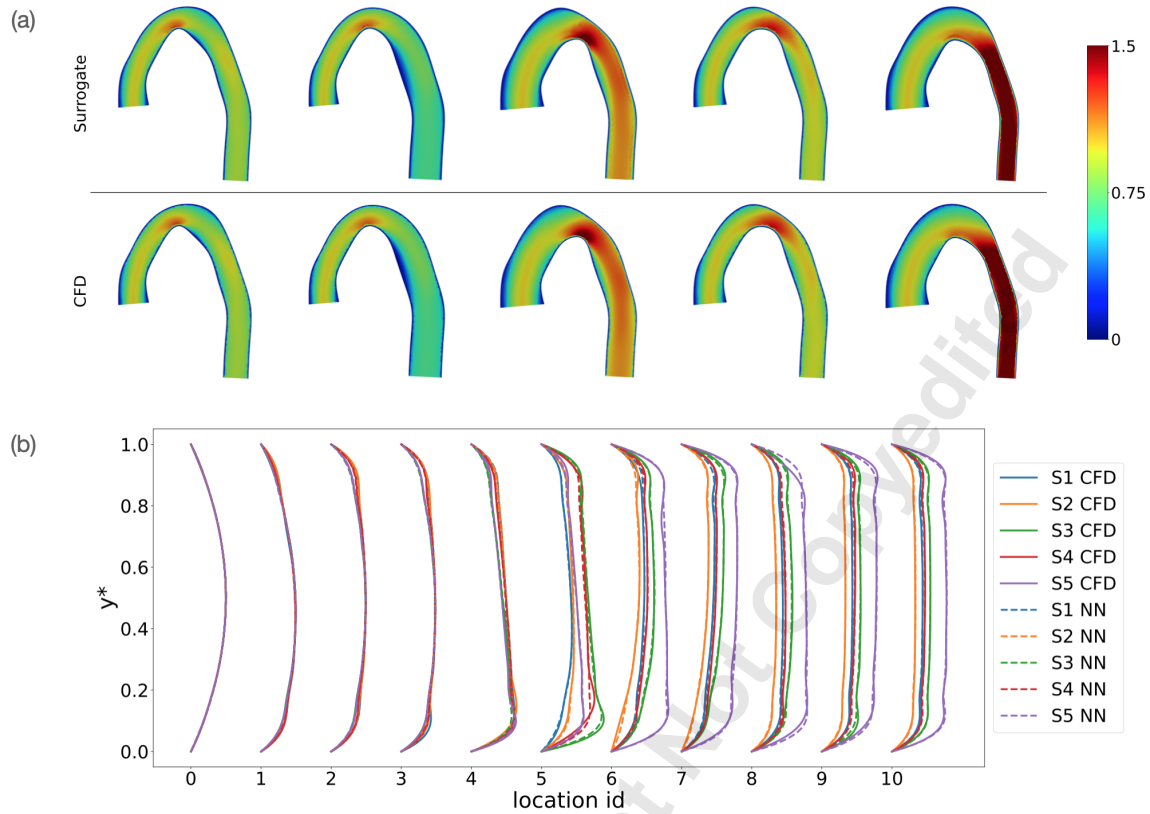


Figure 3: Comparison of the DNN-based surrogate model predictions and the CFD ground truth. (a) velocity magnitude contours: prediction (top row), CFD ground truth (bottom row). (b) velocity profiles on all 11 locations along the centerline: e.g., “S1 NN” and “S1 CFD” denote the surrogate and the ground truth for the first sample (first column in (a)) respectively.

Quantitatively, we also compare the velocity profiles over cross-sections at all 11 locations between the predictions and the ground truth in Figure 3. As expected, the predicted velocity profiles agree well to the ground truth for all five samples at 11 locations. Small discrepancies can be found at the 6-8th locations, where the sharp curvature of the aorta shape induces large gradient flows. The boundary layer at location id 8 is less accurate, which may affects wall shear stress calculations. This can be improved by adding finer boundary layer meshes which will be addressed in the more realistic CFD models in future work. We also calculated the velocity magnitude ARMSE for all the five examples as shown in table 1. The result shows maximum discrepancy near the sixth location as expected. Note that the error at

the inlet is zero because the inlet velocity is already prescribed. Although each FOM CFD simulation is relatively cheap, it still poses a great computational challenge to systematical analysis of MCMC algorithms in different settings. Since the DL model fully trained on the entire CFD dataset ($N = 10000$) yields close results to the CFD ground truth, we will use it as the synthetic “CFD model” (i.e., FOM) for subsequent MCMC experiments.

Locations(No.)	1(Inlet)	2	3	4	5	6	7	8	9	10	11(Outlet)
ARMSE	0%	0.015%	0.011%	0.017%	0.032%	0.15%	0.052%	0.024%	0.020%	0.025%	0.090%

Table 1: Quantitative comparison of ARMSE of the velocity magnitude on 11 locations along the centerline between prediction (surrogate) and ground truth (CFD)

3.2. Geometric uncertainty propagation and reduction

Each MCMC simulation is performed with more than 10000 samples accepted to ensure a low sampling error. The proposal distribution for every MCMC trial is a multivariate normal distribution over three shape parameters with no correlation structures imposed. The proposal variance is set as 0.1 for all three dimensions such that good mixing is observed. The velocity magnitude information on 50 uniformly sampled locations is “observed” independently with 5% Gaussian noises. The observation data are synthesized from the CFD simulation with a specified “true” shape $\theta = [-0.1, 0.3, -0.2]^T$, assumed unknown in the MCMC inference process. The likelihood function is computed by taking the product of the conditional density of all observations given θ . The classic Metropolis-Hasting algorithm coupled with synthetic FOM is used to generate converged MCMC chains, which is used as the reference (ground truth) results for all MCMC experiments. The proposed method, delayed acceptance Metropolis-Hasting MCMC with online training (DA.OLT), is run with the same settings and will be compared against the reference results. In the proposed method, the initial surrogate model is trained on the set with the size of 10, which has a large prediction error (Figure 2). Two chains are run in parallel on two different CPU cores and the online update interval dt is 10 steps. The testing error threshold is set to be $\epsilon_{test}^* = 1 \times 10^{-5}$. Note that every online refinement of the DL surrogate is performed on the

GPU in parallel to the main sampling process. Also, benefiting from the continuous transfer learning setup, each update is set to run only for 100 epochs, much less than the 1500 epochs for the initial training from scratch. Finally, the online update stopped at the 80th refinement step, where the test error falls below the threshold ϵ_{test}^* . More detailed statistics of the proposed *DA_OLT* Metropolis-Hastings MCMC algorithm is plotted in Figure C.9 in the Appendix C. The posterior sampling results obtained by the proposed *DA_OLT* method are compared against the ground truth in Figure 4. To show how the geometric uncertainty

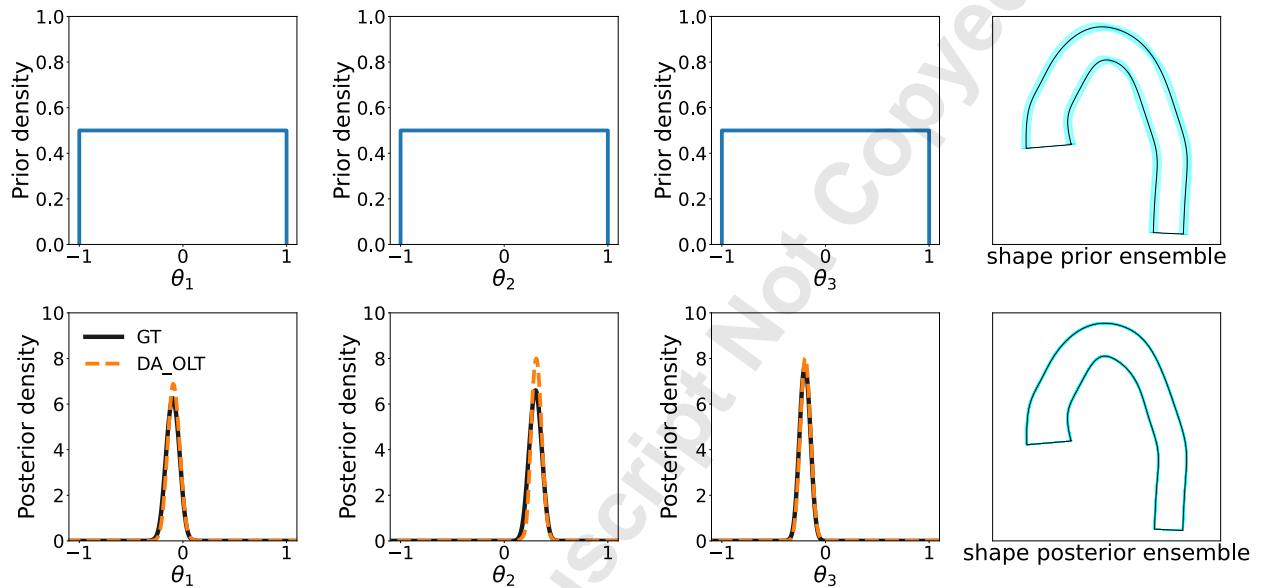


Figure 4: Comparison of prior (top row) and posterior (bottom row) of the *DA_OLT* and GT MCMC algorithms: density for the shape θ (1st,2nd and 3rd column), shape ensemble (4th column)

311

is reduced, we present the prior distributions of all three shape parameters in the first row, whereas the posterior distributions are plotted in the second row. Note that all sampled distributions are fitted by a Gaussian mixture density via sklearn GaussianMixture module with the same hyperparameter setting. Moreover, 1000 randomly selected aorta shape samples are plotted at the rightmost column in Figure 4, where the black line is the mean shape. The non-informative uniform prior distributions over the three parameters are updated to the posterior ones, which agree well with the ground truth solutions. Compared to the prior shape ensemble, the uncertainty range of the posterior ensemble is very small and

concentrated to the synthetic truth, showing that the true aorta shape can be inferred with a significant reduction of uncertainty. The good agreement between the posterior distributions obtained by the proposed *DA_OLT* and ground truth (GT) demonstrates the effectiveness of the proposed method. However, slight discrepancies can be seen in the distributions of the second parameter θ_2 , which corresponds to the perturbation of the sixth radius. This might be because the curvature is rapidly changed near the location of the sixth radius, posing challenges in surrogate modeling.

In addition to the input shape uncertainty, we also investigate the posterior distributions of the maximum and mean velocity magnitudes (V_{max} and V_{mean}) by propagating the geometric uncertainty forward to flow via the FOM simulations. The propagated prior and posterior distributions of V_{max} and V_{mean} are shown in Figure 5. We observed that the

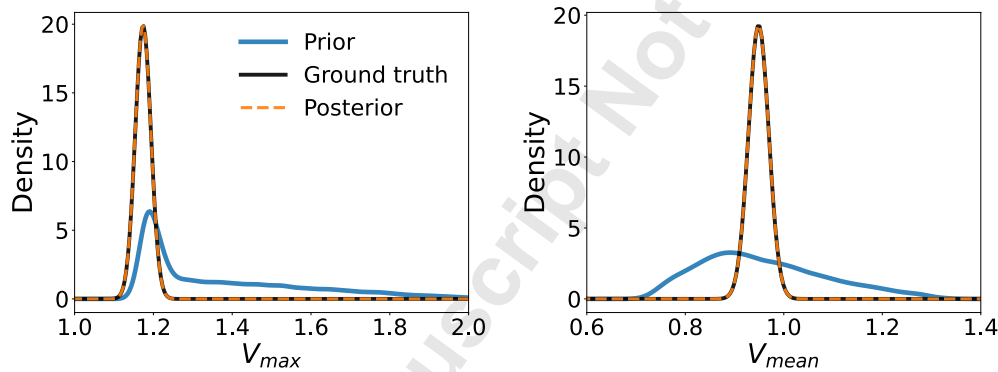


Figure 5: Comparison of prior and posterior of the *DA_OLT* and GT MCMC algorithms: density for maximum velocity V_{max} (left) and maximum velocity (right)

non-informative prior is more spread out for both maximum and mean velocity magnitudes, while their posteriors are significantly concentrated to 1.2 m/s and 0.96 m/s, respectively, indicating a notable uncertainty reduction. The density contraction is very obvious for the maximum velocity, implying the fact that a large proportion of the CFD samples has the maximum velocity near 1.2 m/s. Again, the posterior distributions obtained by the proposed *DA_OLT* perfectly agree with reference using FOM. A collection of five randomly selected aorta shapes from prior and posterior distributions are shown in Figure 6, where the velocity magnitude contours are also plotted.

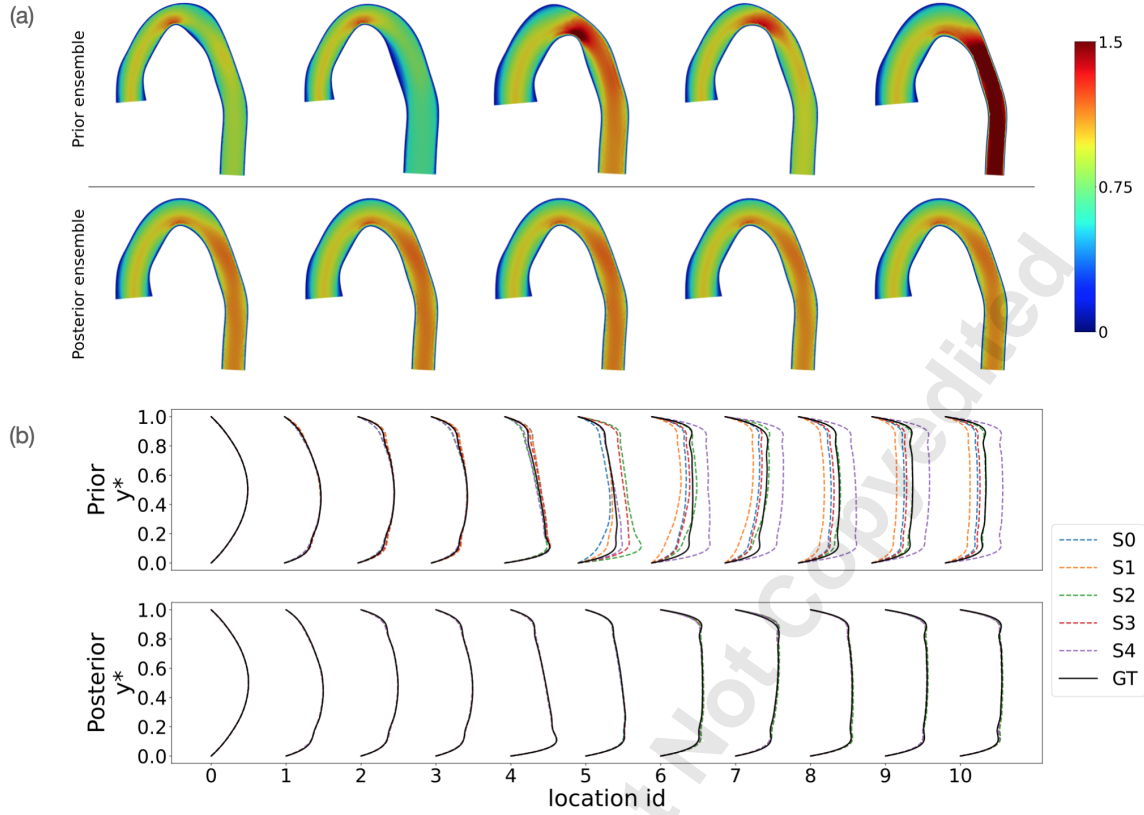


Figure 6: Comparison of the prior and posterior ensemble. (a) velocity magnitude field: Prior ensemble (top row), posterior ensemble (bottom row). (b) velocity profile at all 11 locations, e.g., “S1” denotes the first sample (first column in (a)) and the ground truth denotes the CFD results at parameter $\theta = [-0.1, 0.3, -0.2]^T$.

The top row shows five instances from the prior ensemble, while the collection of the posterior samples from *DA_OLT* is listed at the bottom. The variation of the velocity field is significantly diminished compared to the prior instances. In addition, we added the velocity profile comparisons between the prior and posterior ensemble and the ground truth ($\theta = [-0.1, 0.3, -0.2]^T$) in Figure 6. The results show a large variance of velocity profiles for the prior ensemble, whereas the posterior profiles are much more consistent and closer to the ground truth, indicating significant uncertainty reduction by the inverse UQ process. We also compare the ARMSE of the velocity magnitude on 11 locations along the centerline between the ensembles (prior and posterior) to the ground truth. The result shows that the average differences between the prior ensemble and the ground truth are larger than that

between the posterior ensemble and the ground truth. In other words, the prior samples have very large uncertainties at 11 locations whereas the posterior samples have much less uncertainty and high similarities to the ground truth. In general, the inverse UQ process can increase the reliability of the velocity field prediction and hence facilitate CFD-supported clinical diagnosis of cardiovascular diseases.

Locations(No.)	1(Inlet)	2	3	4	5	6	7	8	9	10	11(Outlet)
prior ARMSE	0%	0.064%	0.18%	0.32%	6.1%	9.7%	8.2%	7.7%	7.8%	8.1%	8.2%
posterior ARMSE	0%	0.0076%	0.0067%	0.0079%	0.010%	0.030%	0.035%	0.023%	0.016%	0.023%	0.030%

Table 2: Quantitative comparison of ARMSE of the velocity magnitude on 11 locations along the centerline between (prior and posterior) ensemble and ground truth (CFD)

4. Discussion

4.1. Performance comparison of different MCMC methods

This section will mainly compare and discuss the performance of DL-assisted Metropolis-Hastings MCMC with different settings. Besides the reference MCMC (GT) and the *DA.OLT* method, two other MCMC algorithms are added to the comparison: one is the classic Metropolis-Hasting with a low-fidelity surrogate model trained by 10 samples (LF MCMC), and the other is a DA Metropolis-Hastings with the same LF surrogate model without an online update (DA). Figure 7 shows the posterior distributions for the shape parameter θ and the maximum velocity V_{max} obtained by different MCMC methods. In terms of the posterior mean, a straightforward observation is that the LF MCMC mean deviates from the ground truth to a large extent, especially for the shape parameters, whereas the Metropolis-Hastings algorithms featuring Delayed Acceptance (DA) always accurately capture the posterior mean. Note that the detailed balance is not satisfied for the LF MCMC method, which results in an erroneous posterior distribution approximation. In contrast, the MCMC method featuring Delayed Acceptance satisfies the detailed balance, accounting for a much more accurate prediction of the mean. Viewing from the posterior shape, the proposed method stands out among all the methods. The *DA* MCMC method tends to heavily over-predict the posterior

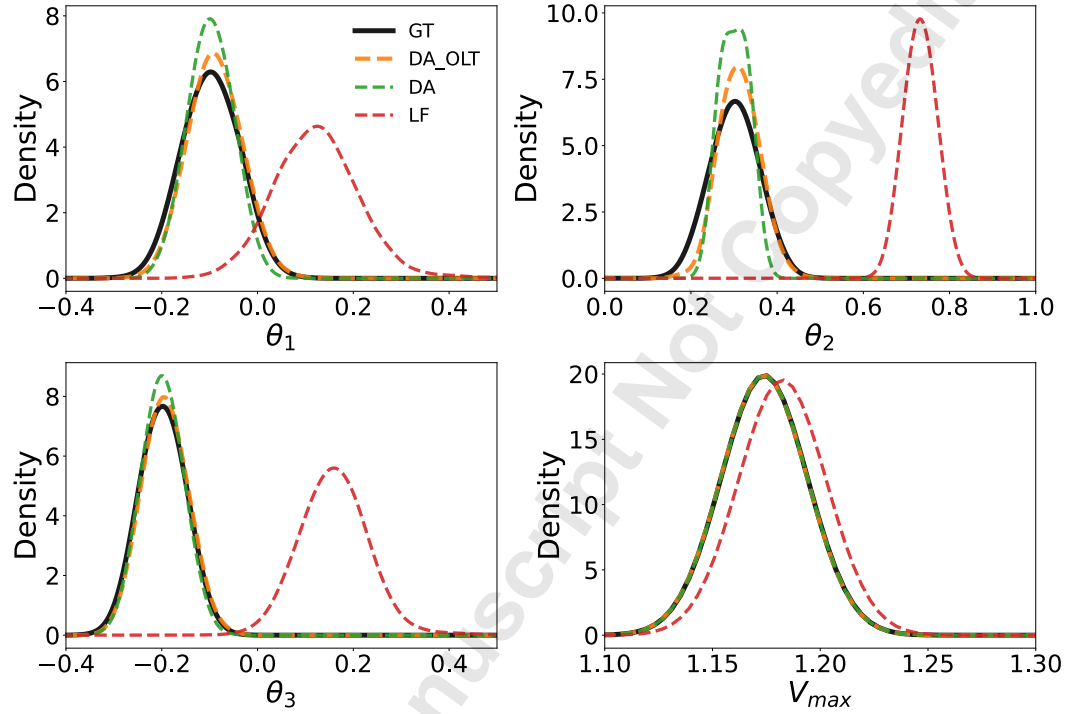


Figure 7: Comparison of posterior among different MCMC algorithms (GT:ground truth/high-fidelity MCMC, DA_OLT : delayed acceptance online training MCMC, DA: delayed acceptance MCMC, LF: low fidelity MCMC): density for the shape θ (upper left, upper right and lower left), maximum velocity V_{max} (lower right)

peak for shape parameters. Obviously, the presence of the LF filtering step alters the trace of the states and consequently results in improper sample coverage over the high-density region. For the velocity density plots, however, both the DA and *DA_OLT* methods are very close to the reference (GT).

4.2. Efficiency of active-learning delayed-acceptance MCMC

Another vital topic to discuss is the improvement of efficiency over the classic FOM-based Metropolis-Hasting method (GT) for all methods. Since the computational cost of the LF forward model is trivial compared to the FOM simulation, we care about how many realizations of FOM are needed in order to obtain the desired number of MC samples. Hence we create a new criterion called “effective acceptance ratio (EAR)”, simply defined as,

$$EAR = \frac{\text{number of accepted samples}}{\text{number of realizations of the HF model}} \quad (7)$$

Apparently, for the GT and the LF MCMC methods, $EAR = A$ (defined in Eq. 2). Whereas $EAR = A^*$ (defined in Eq. 5) the DA and *DA_OLT* MCMC methods. The Effective acceptance ratios are plotted in Figure 8 (left). As expected, the LF MCMC method has the same level of EAR compared to the ground truth. However, the DA MCMC method has a surprisingly low acceptance ratio when a LF surrogate is used, which means the required FOM queries is extensively higher than that of the classic Metropolis MCMC. In fact, the intention of the delayed acceptance technique is to improve the effective acceptance ratio (second stage) by introducing a surrogate screening step (first stage). However, the addition of the screening step may backfire when the surrogate model’s prediction error is significant. Active learning can come into play to timely improve the LF surrogate by leveraging gradually collected FOM solutions to address this issue. As shown in Figure 8 (left), the proposed *DA_OLT* method raises the EAR to 45.6%, which is about five times higher than the reference MCMC. Note that the EAR will increase if the accuracy of the surrogate model grows and will asymptotically reach 100% if the surrogate has the same accuracy as the FOM.

We further evaluate the extent of efficiency improvement of the proposed method at different ratios $R_{HF/LF}$ of cost between the FOM and surrogate forward models. A normalized

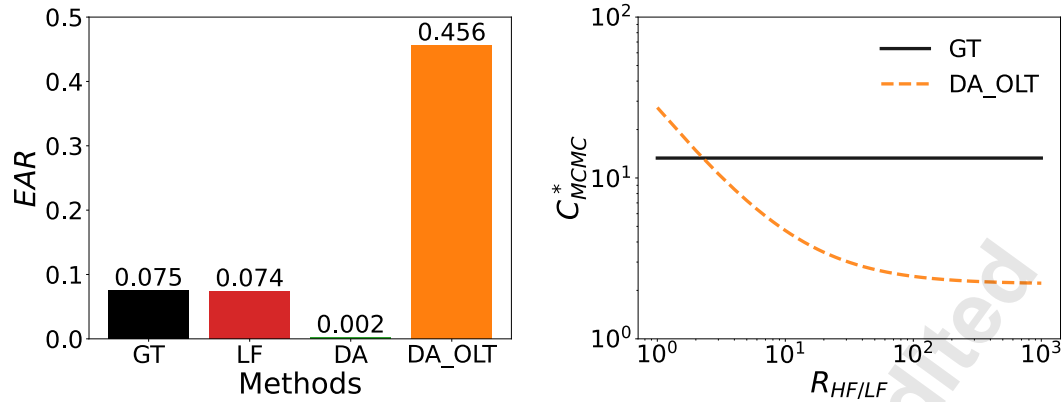


Figure 8: Comparison of effective acceptance ratio EAR of posterior among different MCMC algorithms (left), Relationship of normalized cost C_{MCMC}^* and HF-to-LF ratio $R_{(HF/LF)}$ for DA_OLT and GT

cost C_{MCMC}^* is defined which represents the total cost of the MCMC simulation normalized by the cost of executing FOM queries for the accepted samples. The relationship of those two variables are demonstrated in the right panel of Figure 8. The result indicates the efficiency of the proposed DA_OLT MCMC method will be significantly improved over the reference as the FOM becomes more and more expensive. On the other hand, the cost of the DA_OLT MCMC method can be higher than the reference MCMC method when the surrogate model's cost is close to the FOM due to two-stage acceptance/rejection determination. The intersection of two curves is located at $R_{HF/LF}^* = 2.35$. Usually, for DL-based surrogate models for complex computational hemodynamics, the $R_{HF/LF}$ value is much higher than the critical value $R_{HF/LF}^* = 2.35$ due to the extremely fast inference speed of the neural network and high cost of 3-D CFD simulations. For example, the surrogate model and the CFD forward model cost about 0.006 and 6 seconds in the current pipeline, resulting in a high to low fidelity model ratio of $R_{HF/LF} = 6/0.006 = 1000$. Suppose the MCMC has to collect 10000 samples to be reliable, then a standard MCMC with CFD forward model requires 221.6 hours, whereas the MCMC_OLT method costs about 37 hours, resulting in a reduction in time of 83.3% Note that in this work, we adopted the high fidelity (HF) model as the neural network with 10000 training data, so the HF model costs 0.006 as well and the actual time costs in this work are 13.3 minutes and 27.4 minutes for classic MCMC and the

propose DA_OLT respectively. Assuming in a more realistic application where the forward model costs 6 minutes and the surrogate model's cost stays the same, the computational costs for classic MCMC and proposed DA_OLT method are 13295 and 2193 hours respectively, bringing an efficiency boost of 83.5%. In conclusion, our proposed method will bring a significant efficiency boost compared to the conventional Metropolis-Hastings MCMC method for forward and inverse UQ problems.

4.3. Limitations for patient-specific applications

In the previous results, we demonstrate the feasibility of the algorithm using synthetic data on simplified 2-D aorta geometries as an example for proof of concept. However, it is noteworthy to discuss whether the method can be extended to patient-specific hemodynamic applications. Admittedly, the CFD simulation in the present work is a simplified numerical example and has not considered many realistic aspects for real patient-specific cases, e.g., 3D patient-specific geometry, measured pulsatile inflow boundary conditions, fluid-structure interactions (FSI), etc. Hence, extending the proposed algorithm to patient-specific settings requires several improvements: 1). The CFD forward model can be updated to patient-specific settings. E.g., setting the CFD simulation to be 3-D and transient, assigning measure pulsatile inlet flow boundary conditions based on clinical measurements, adding Windkessel boundary conditions at the outlet, enabling FSI, etc. 2). Use a more sophisticated parameterization method to describe complex 3-D geometries (e.g., 3-D aorta with branches). 3). Use a more sophisticated neural network (e.g., Graph neural network (GNN)) to learn the mapping between 3-D input geometry and field outputs. In future work, we expect to address those issues to apply the proposed algorithm to clinical usage. In addition, in the context of patient-specific hemodynamics simulation, information such as flow rate data obtained from 2D PC MRI at certain cross-sections or 3D/4D flow MRI data can be assimilated using the proposed method to reduce the uncertainty of the geometries of interest and enhance the model predictive accuracy. One can also evaluate the similarity between the predicted aorta geometry shape and the flow field to direct image data (e.g., 4D flow MRI).

5. Conclusion

This work presents a Bayesian framework for geometric uncertainty reduction using DL-assisted MCMC sampling. First, shape parameterization is performed on the aorta geometries based on a patient-specific 3-D aorta sample, from which multiple aorta geometries are created by perturbation of radii at different sections. An automatic python routine is established to encapsulate all necessary simulation procedures to simulate flow information from a given aorta shape θ . The geometry-to-flow surrogate model is built upon the CFD dataset to learn the non-linear relationship between the input shape to the flow solution fields. The trained surrogate model exhibits an increase of accuracy as the size of the training set grows. We propose a Metropolis-Hastings algorithm featuring delayed acceptance and active learning (*DA-OLT*), enabling the inference of the aorta shape and uncertainty reduction based on observed velocity information at sparse locations. The results show a significant uncertainty reduction given a non-informative prior. A good consistency is observed between the proposed method and the reference Metropolis-Hasting MCMC (GT MCMC) in terms of the posterior approximation. The proposed method is compared with Delayed Acceptance Metropolis-Hastings MCMC (DA MCMC) and standard Metropolis-Hastings MCMC algorithms equipped with the same surrogate model without an active learning component (LF MCMC). Regarding accuracy, the proposed method stands out among those methods, while the LF MCMC method completely fails due to high bias of the surrogate. As for the efficiency, again, the proposed method brings a huge efficiency improvement, whereas the standard DA Metropolis-Hastings MCMC method failed in this specific setting due to the large discrepancy between the surrogate and FOM solutions. The cost of the proposed method is further analyzed by inspecting the normalized cost change as a function of the HF-to-LF model cost ratio. It appears that the efficiency boost is more conspicuous when the FOM is more expensive. Typically when a DNN-bases surrogate model is embedded into the proposed DA scheme with active learning, a considerable promotion of efficiency is guaranteed.

In general, this work focuses on the algorithmic development of a method to reduce the

uncertainty of reconstructed geometry via assimilating sparse, noisy flow measurement data using a Bayesian framework. The major highlight of the algorithm is that it improves the UQ process, where only one version of the forward model (either full-order or reduced-order) is repetitively evaluated, and enables combining models with different levels of complexity into one UQ algorithm. In addition, the active-learning feature further reduces the cost of training data generation via selectively collecting the data in an online manner and gains an extra efficiency boost in parallel MCMC settings. Achieving patient-specific settings for the CFD forward model is out of the scope of this paper. The current pipeline is subjected to multiple limitations for its application in clinical situations, where the forward model is usually patient-specific. As a result, certain improvements have to be made for clinical applications (e.g., using patient-specific settings for the forward model, updating the parameterization for complex 3-D geometry and improving the neural network architecture to learn more complicated mappings between 3-D geometry and the field outputs), which will be addressed in future work.

Acknowledgment

The authors would like to acknowledge the funds from National Science Foundation, United States of America under award numbers CMMI-1934300 and OAC-2047127, the Air Force Office of Scientific Research (AFOSR), United States of America under award number FA9550-22-1-0065, and startup funds from the College of Engineering at University of Notre Dame in supporting this study.

Compliance with Ethical Standards

Conflict of Interest: The authors declare that they have no conflict of interest.

Appendix A. Optimized deep neural network architecture

NN Type	Layers	Characteristics
MLP	Linear Layer	Input feature = 3 input feature = 8
	Relu	-
	Linear Layer	Input feature = 8 input feature = 256
	Relu	-
	Linear Layer	Input feature = 256 input feature = 128
	Relu	-
	Linear Layer	Input feature = 128 input feature = 20

Table A.3: The optimized MLP structure based on Bayesian tuning

494 **Appendix B. Proposed DL-assisted parallel-chain MCMC algorithm**

Algorithm 1 Active Learning Delayed Acceptance Metropolis-Hastings MCMC

Data: Starting point $\theta = \theta_0 \in \Omega$, test error threshold ϵ_{test}^* , online update period dt ; online dataset $OLT = []$,

initial accepted samples set $AS = []$

Result: Accepted sample set AS

for each MCMC iteration $i = 1, 2, \dots$ **do**

 Draw next state θ_{i+1} ; Calculate surrogate model posterior $\tilde{\pi}(\theta_{i+1})$ and acceptance ratio $\tilde{A}(\theta_i | \theta_{i+1})$;

 Generate a random number $a_1 \sim \mathbf{U}(0, 1)$;

if $\tilde{A}(\theta_i | \theta_{i+1}) > a_1$ **then**

 Calculate high-fidelity model posterior $\pi(\theta_{i+1})$ and acceptance ratio $A^*(\theta_i | \theta_{i+1})$;

 Generate a random number $a_2 \sim \mathbf{U}(0, 1)$;

 Append sample θ_{i+1} and its solution $\mathcal{F}(\theta_{i+1})$ to online dataset OLT ;

if $A^*(\theta_i | \theta_{i+1}) > a_2$ **then**

 Accept the new sample $\theta_i = \theta_{i+1}$;

 Append sample θ_{i+1} to Accepted sample set AS ;

else

 Reject sample θ_{i+1}

end

else

 Reject sample θ_{i+1}

end

if $\text{len}(OLT)\%dt = 0$ **then**

 test the surrogate model $\tilde{\mathcal{F}}(\theta_{i+1})$ on the dataset OLT ; **if** test error $> \epsilon_{test}^*$ **then**

 add new data to dataset OLT , train surrogate model

else

 stop the training update

end

end

if training is finished **then**

 update surrogate model $\tilde{\mathcal{F}} = \tilde{\mathcal{F}}'$

end

end

495 Appendix C. DA.OLT algorithm statistics

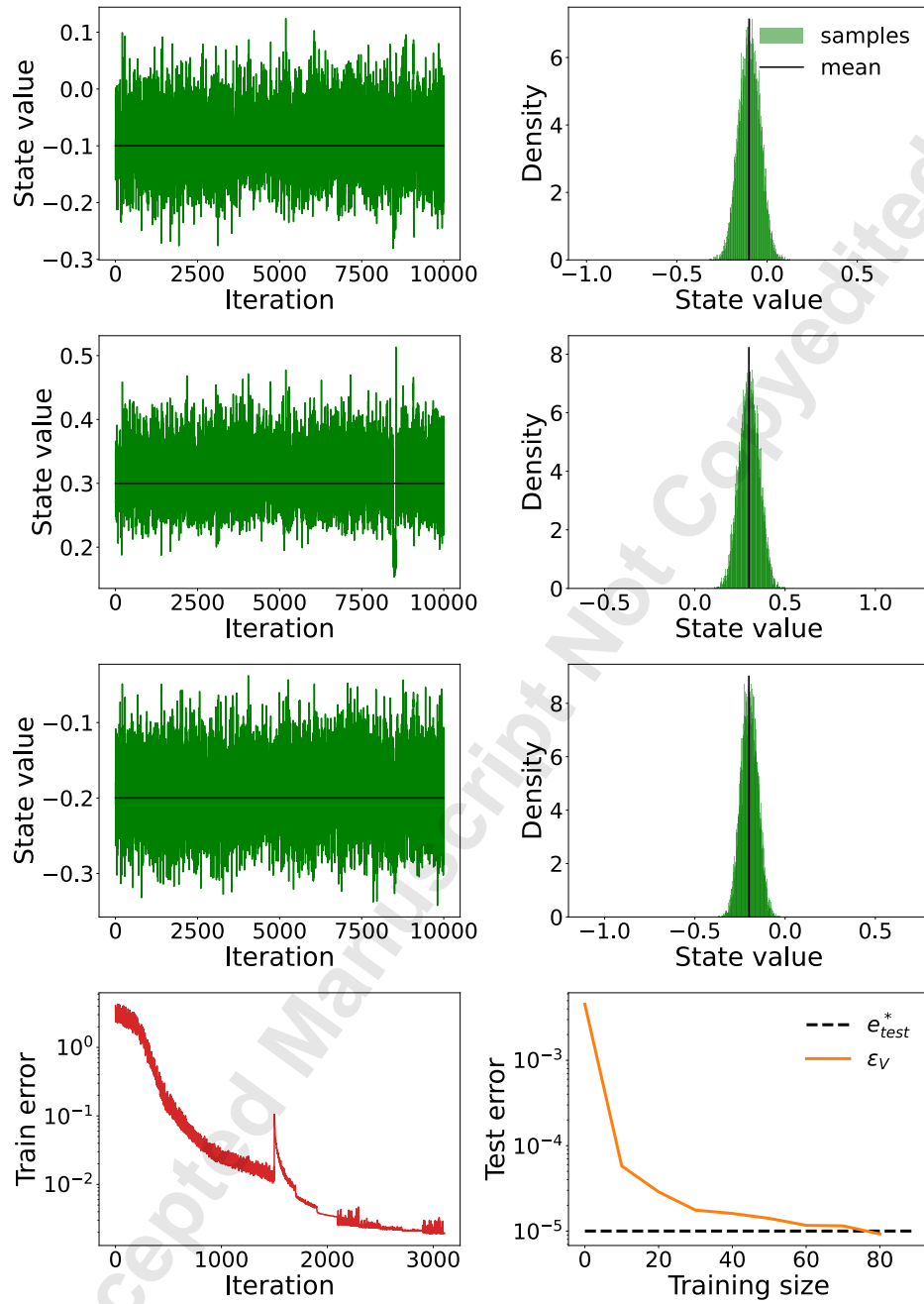


Figure C.9: Statistics for *DA.OLT* MCMC algorithm: trace of state (left column, first 3 rows), posterior sample distribution of state (right column, first 3 rows), training error history (left, 4th row), online test error ϵ_{test}^* and threshold ϵ_V^* (right, 4th row)

References

[1] D. A. Steinman, C. A. Taylor, Flow imaging and computing: large artery hemodynamics, *Annals of biomedical engineering* 33 (12) (2005) 1704–1709.

[2] J. Xiang, L. Antiga, N. Varble, K. V. Snyder, E. I. Levy, A. H. Siddiqui, H. Meng, A view: an image-based clinical computational tool for intracranial aneurysm flow visualization and clinical management, *Annals of biomedical engineering* 44 (4) (2016) 1085–1096.

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

[4] P. Berg, S. Saalfeld, S. Voß, O. Beuing, G. Janiga, A review on the reliability of hemodynamic modeling in intracranial aneurysms: why computational fluid dynamics alone cannot solve the equation, *Neurosurgical Focus* 47 (1) (2019) E15.

[5] D. A. Steinman, V. M. Pereira, How patient specific are patient-specific computational models of cerebral aneurysms? an overview of sources of error and variability, *Neurosurgical Focus* 47 (1) (2019) E14.

[6] D. A. Steinman, F. Migliavacca, Special issue on verification, validation, and uncertainty quantification of cardiovascular models: towards effective vvuq for translating cardiovascular modelling to clinical utility, *Cardiovascular engineering and technology* 9 (4) (2018) 539–543.

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

[8] H. Gao, X. Zhu, J.-X. Wang, A bi-fidelity surrogate modeling approach for uncertainty propagation in three-dimensional hemodynamic simulations, *Computer Methods in Applied Mechanics and Engineering* 366 (2020) 113047.

- 520 [9] S. Guzzetti, L. M. Alvarez, P. Blanco, K. T. Carlberg, A. Veneziani, Propagating uncer-
 521 tainties in large-scale hemodynamics models via network uncertainty quantification and
 522 reduced-order modeling, *Computer Methods in Applied Mechanics and Engineering* 358
 523 (2020) 112626.
- 524 [10] G. Bertaglia, V. Caleffi, L. Pareschi, A. Valiani, Uncertainty quantification of viscoelas-
 525 tic parameters in arterial hemodynamics with the a-fsi blood flow model, *Journal of*
 526 *Computational Physics* 430 (2021) 110102.
- 527 [11] J. S. Tran, D. E. Schiavazzi, A. M. Kahn, A. L. Marsden, Uncertainty quantification of
 528 simulated biomechanical stimuli in coronary artery bypass grafts, *Computer methods*
 529 *in applied mechanics and engineering* 345 (2019) 402–428.
- 530 [12] J. S. Tran, D. E. Schiavazzi, A. B. Ramachandra, A. M. Kahn, A. L. Marsden, Auto-
 531 mated tuning for parameter identification and uncertainty quantification in multi-scale
 532 coronary simulations, *Computers & fluids* 142 (2017) 128–138.
- 533 [13] S. Sankaran, H. J. Kim, G. Choi, C. A. Taylor, Uncertainty quantification in coronary
 534 blood flow simulations: impact of geometry, boundary conditions and blood viscosity,
 535 *Journal of biomechanics* 49 (12) (2016) 2540–2547.
- 536 [14] S. Sankaran, A. L. Marsden, A stochastic collocation method for uncertainty quantifica-
 537 tion and propagation in cardiovascular simulations, *Journal of biomechanical engineer-*
 538 *ing* 133 (3) (2011).
- 539 [15] S. Sankaran, A. L. Marsden, The impact of uncertainty on shape optimization of ideal-
 540 ized bypass graft models in unsteady flow, *Physics of fluids* 22 (12) (2010) 121902.
- 541 [16] C. M. Fleeter, G. Geraci, D. E. Schiavazzi, A. M. Kahn, A. L. Marsden, Multilevel
 542 and multifidelity uncertainty quantification for cardiovascular hemodynamics, *Com-*
 543 *puter methods in applied mechanics and engineering* 365 (2020) 113030.
- 544 [17] D. Lucor, O. P. Le Maître, Cardiovascular modeling with adapted parametric inference,
 545 *ESAIM: Proceedings and Surveys* 62 (2018) 91–107.

[18] J.-X. Wang, X. Hu, S. C. Shadden, Data-augmented modeling of intracranial pressure, *Annals of biomedical engineering* 47 (3) (2019) 714–730.

[19] A. Arnold, C. Battista, D. Bia, Y. Z. German, R. L. Armentano, H. Tran, M. S. Olufsen, Uncertainty quantification in a patient-specific one-dimensional arterial network model: Enkf-based inflow estimator, *Journal of Verification, Validation and Uncertainty Quantification* 2 (1) (2017).

[20] J. Seo, C. Fleeter, A. M. Kahn, A. L. Marsden, D. E. Schiavazzi, Multifidelity estimators for coronary circulation models under clinically informed data uncertainty, *International Journal for Uncertainty Quantification* 10 (5) (2020).

[21] L. Sun, J.-X. Wang, Physics-constrained bayesian neural network for fluid flow reconstruction with sparse and noisy data, *Theoretical and Applied Mechanics Letters* 10 (3) (2020) 161–169.

[22] B. Lakshminarayanan, A. Pritzel, C. Blundell, Simple and scalable predictive uncertainty estimation using deep ensembles, *Advances in neural information processing systems* 30 (2017).

[23] B. Carpenter, A. Gelman, M. D. Hoffman, D. Lee, B. Goodrich, M. Betancourt, M. Brubaker, J. Guo, P. Li, A. Riddell, Stan: A probabilistic programming language, *Journal of statistical software* 76 (1) (2017).

[24] N. Bliznyuk, D. Ruppert, C. Shoemaker, R. Regis, S. Wild, P. Mugunthan, Bayesian calibration and uncertainty analysis for computationally expensive models using optimization and radial basis function approximation, *Journal of Computational and Graphical Statistics* 17 (2) (2008) 270–294.

[25] J. Zhang, A. A. Taflanidis, Adaptive kriging stochastic sampling and density approximation and its application to rare-event estimation, *ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part A: Civil Engineering* 4 (3) (2018) 04018021.

- [26] A. H. Elsheikh, I. Hoteit, M. F. Wheeler, Efficient bayesian inference of subsurface flow models using nested sampling and sparse polynomial chaos surrogates, *Computer Methods in Applied Mechanics and Engineering* 269 (2014) 515–537.
- [27] L. Sun, H. Gao, S. Pan, J.-X. Wang, Surrogate modeling for fluid flows based on physics-constrained deep learning without simulation data, *Computer Methods in Applied Mechanics and Engineering* 361 (2020) 112732.
- [28] P. Du, X. Zhu, J.-X. Wang, Deep learning-based surrogate model for 3-d patient-specific computational fluid dynamics, *arXiv* (2022).
- [29] T. Cui, C. Fox, M. O’sullivan, Bayesian calibration of a large-scale geothermal reservoir model by a new adaptive delayed acceptance metropolis hastings algorithm, *Water Resources Research* 47 (10) (2011).
- [30] T. Cui, Y. M. Marzouk, K. E. Willcox, Data-driven model reduction for the bayesian solution of inverse problems, *International Journal for Numerical Methods in Engineering* 102 (5) (2015) 966–990.
- [31] J. Zhang, A. A. Taflanidis, Accelerating mcmc via kriging-based adaptive independent proposals and delayed rejection, *Computer Methods in Applied Mechanics and Engineering* 355 (2019) 1124–1147.
- [32] P. R. Conrad, A. D. Davis, Y. M. Marzouk, N. S. Pillai, A. Smith, Parallel local approximation mcmc for expensive models, *SIAM/ASA Journal on Uncertainty Quantification* 6 (1) (2018) 339–373.
- [33] X. Wang, F. Guo, K. A. Heller, D. B. Dunson, Parallelizing mcmc with random partition trees, *Advances in neural information processing systems* 28 (2015).
- [34] S. L. Scott, A. W. Blocker, F. V. Bonassi, H. A. Chipman, E. I. George, R. E. McCulloch, Bayes and big data: The consensus monte carlo algorithm, *International Journal of Management Science and Engineering Management* 11 (2) (2016) 78–88.

- [35] D. Mesquita, P. Blomstedt, S. Kaski, Embarrassingly parallel mcmc using deep invertible transformations, in: *Uncertainty in Artificial Intelligence*, PMLR, 2020, pp. 1244–1252.
- [36] B. Calderhead, A general construction for parallelizing metropolis- hasting algorithms, *Proceedings of the National Academy of Sciences* 111 (49) (2014) 17408–17413.
- [37] S. Yang, Y. Chen, E. Bernton, J. S. Liu, On parallelizable markov chain monte carlo algorithms with waste-recycling, *Statistics and Computing* 28 (5) (2018) 1073–1081.
- [38] T. Schwedes, B. Calderhead, Rao-blackwellised parallel mcmc, in: *International Conference on Artificial Intelligence and Statistics*, PMLR, 2021, pp. 3448–3456.
- [39] Y. Ni, Y. Deng, S. Li, Pmba: A parallel mcmc bayesian computing accelerator, *IEEE Access* 9 (2021) 65536–65546.
- [40] D. J. Wilkinson, *Parallel bayesian computation*, *Statistics Textbooks and Monographs* 184 (2006) 477.
- [41] J. Ye, A. Wallace, J. Thompson, Parallel markov chain monte carlo computation for varying-dimension signal analysis, in: *2009 17th European Signal Processing Conference*, IEEE, 2009, pp. 2673–2677.
- [42] P. Jacob, C. P. Robert, M. H. Smith, Using parallel computation to improve independent metropolis–hastings based estimation, *Journal of Computational and Graphical Statistics* 20 (3) (2011) 616–635.
- [43] L. Martino, V. Elvira, D. Luengo, J. Corander, F. Louzada, Orthogonal parallel mcmc methods for sampling and optimization, *Digital Signal Processing* 58 (2016) 64–84.
- [44] S. Sankaran, L. Grady, C. A. Taylor, Fast computation of hemodynamic sensitivity to lumen segmentation uncertainty, *IEEE transactions on medical imaging* 34 (12) (2015) 2562–2571.

[45] M. Cilla, M. Casales, E. Peña, M. Martínez, M. Malvè, A parametric model for studying
the aorta hemodynamics by means of the computational fluid dynamics, *Journal of*
Biomechanics 103 (2020) 109691.

[46] J. Brüning, F. Hellmeier, P. Yevtushenko, T. Kühne, L. Goubergrits, Uncertainty quan-
tification for non-invasive assessment of pressure drop across a coarctation of the aorta
using cfd, *Cardiovascular engineering and technology* 9 (4) (2018) 582–596.

[47] N. Schlömer, pygalmesh: Python interface for cgal’s meshing tools.

[48] R. H. Pletcher, J. C. Tannehill, D. Anderson, *Computational fluid mechanics and heat*
transfer, CRC press, 2012.

[49] R. Liaw, E. Liang, R. Nishihara, P. Moritz, J. E. Gonzalez, I. Stoica, Tune: A research
platform for distributed model selection and training, *arXiv preprint arXiv:1807.05118*
(2018).