Machine learning in cardiovascular flows modeling: Predicting arterial blood pressure from non-invasive 4D flow MRI data using physics-informed neural networks

Article *in* Computer Methods in Applied Mechanics and Engineering · January 2020

DOI: 10.1016/j.cma.2019.112623

Georgios Kissas
University of Pennsylvania
2 PUBLICATIONS **78** CITATIONS

SEE PROFILE

Yibo Yang
University of Pennsylvania
20 PUBLICATIONS **293** CITATIONS

SEE PROFILE

John A Detre
University of Pennsylvania
440 PUBLICATIONS **29,878** CITATIONS

SEE PROFILE

Paris Perdikaris
University of Pennsylvania
72 PUBLICATIONS **3,613** CITATIONS

SEE PROFILE

Available online at www.sciencedirect.com

ScienceDirect

Comput. Methods Appl. Mech. Engrg. 358 (2020) 112623

Computer methods in applied mechanics and engineering

www.elsevier.com/locate/cma

Machine learning in cardiovascular flows modeling: Predicting arterial blood pressure from non-invasive 4D flow MRI data using physics-informed neural networks

Georgios Kissas[a], Yibo Yang[a], Eileen Hwuang[b], Walter R. Witschey[c], John A. Detre[d], Paris Perdikaris[a],*

[a] Department of Mechanical Engineering and Applied Mechanics, University of Pennsylvania, Philadelphia, PA, USA
[b] Department of Bioengineering, University of Pennsylvania, Philadelphia, PA, USA
[c] Department of Radiology, Perelman School of Medicine, University of Pennsylvania, Philadelphia, PA, USA
[d] Department of Neurology, Perelman School of Medicine, University of Pennsylvania, Philadelphia, PA, USA

Available online 17 September 2019

Highlights

- We introduce physics-informed neural networks to solve conservation laws in graph topologies.
- We show how non-dimensionalization and normalization can mitigate vanishing gradient pathologies.
- For the first time, a physics-informed neural network is applied to real noisy clinical data.
- Arterial pressure is predicted from MRI data of blood velocity and wall displacement.
- Our model also allows for calibrating boundary conditions of conventional flow simulators.

Abstract

Advances in computational science offer a principled pipeline for predictive modeling of cardiovascular flows and aspire to provide a valuable tool for monitoring, diagnostics and surgical planning. Such models can be nowadays deployed on large patient-specific topologies of systemic arterial networks and return detailed predictions on flow patterns, wall shear stresses, and pulse wave propagation. However, their success heavily relies on tedious pre-processing and calibration procedures that typically induce a significant computational cost, thus hampering their clinical applicability. In this work we put forth a machine learning framework that enables the seamless synthesis of non-invasive *in-vivo* measurement techniques and computational flow dynamics models derived from first physical principles. We illustrate this new paradigm by showing how one-dimensional models of pulsatile flow can be used to constrain the output of deep neural networks such that their predictions satisfy the conservation of mass and momentum principles. Once trained on noisy and scattered clinical data of flow and wall displacement, these networks can return physically consistent predictions for velocity, pressure and wall displacement pulse wave propagation, all without the need to employ conventional simulators. A simple post-processing of these outputs can also provide a relatively cheap and effective way for estimating Windkessel model parameters that are required for the calibration of traditional computational

* Corresponding author.
E-mail address: pgp@seas.upenn.edu (P. Perdikaris).

https://doi.org/10.1016/j.cma.2019.112623

models. The effectiveness of the proposed techniques is demonstrated through a series of prototype benchmarks, as well as a realistic clinical case involving *in-vivo* measurements near the aorta/carotid bifurcation of a healthy human subject.

Keywords: Deep neural networks; Blood flow modeling; Pulse wave propagation; Data-driven modeling; Non-invasive diagnostics

1. Introduction

Recent advances in clinical measurement and computational modeling techniques introduce new capabilities for monitoring the human cardiovascular system from different perspectives such as disease surveys [1], bio-medical image processing [2,3], computational mathematics [4–7], bio-physics [8,9], etc. These studies reveal the crucial role played by blood flow, arterial wall mechanics and pressure wave propagation, and how their interplay directly characterizes the functionality of the cardiovascular system both in health and disease (e.g., hypertensive disorders [10]).

Understanding the inner workings of the cardiovascular system has been central to many studies involving clinical, interventional or computational approaches. For instance, Chan et al. [11] proposed placing sensors in the human body to achieve real time measuring of arterial velocity and wall displacement to monitor the health of patients. Although the collected *in-vivo* measurements can be highly accurate, such interventional techniques are sometimes expensive and suffer from limitations that are not easy to address, e.g., difficulties of placing probes in cerebral or uteroplacental arteries [12]. These limitations motivate the use of non-invasive measurement techniques such as bio-medical imaging, advances in which currently define the clinical standard of care [13–16]. To this end, one of the most commonly used techniques is Doppler ultrasound velocimetry [17] that enables the recovery of blood velocity wave propagation. Using information related to the pulsatility of the flow [18], clinicians are able to infer quantitative information about the pressure in the arterial vessels, e.g., large pulsatility is caused by large pressure [9]. Rather than Doppler ultrasound, more recent studies leverage advances in 4D flow Magnetic Resonance Imaging (MRI) [19] to recover the full three-dimensional velocity flow field. This method provides a more detailed representation of the flow, and also a larger spatial coverage that can capture quantities like vessel tortuosity in a more accurate manner. Moreover, structural characteristics like the cross-sectional area and vessel diameter can be recovered by segmenting 2D cine images [20] from MRI measurements. However, critical variables such as the pressure cannot be directly measured by a non-invasive technique [21], and accurate measurements are only accessible by inserting a catheter equipped with sensors inside the vessel of interest.

These difficulties of directly measuring quantities of interest like the pressure in an *in-vivo* and non-invasive manner have motivated the use of computer simulations and computational fluid dynamics models to predict them *in-silico*. Advances in algorithms and computing resources now allow us to perform detailed flow simulations in complex patient-specific arterial topologies using three-dimensional (3D) and/or one-dimensional (1D) formulations of the unsteady Navier–Stokes equations [4–7,22,23]. Such tools have been successfully validated against both *in-vitro* experiments [24] as well as *in-vivo* clinical data [25], and provide a valuable platform for parametric sensitivity studies [26]. Despite their predictive power, computational models have still not made their way into clinical practice primarily due to their high computational cost and the tedious procedures needed for their practical deployment, e.g., mesh generation, parameter calibration, etc. For instance, such models require the precise subscription of boundary conditions that effectively capture the downstream flow dynamics in small arteries and arterioles via the use of Windkessel or structured tree models [27–30]. Inaccurate calibration of the parameters associated with these boundary conditions is often the cause of brittleness in the resulting predictions, thus limiting the translational impact of computational models to the clinical domain.

This goal of this work is to put forth a new paradigm for seamlessly blending non-invasive measurement techniques such 4D flow MRI with computational fluid dynamics models, towards providing a cheap and effective tool for characterizing velocity and pressure pulse wave propagation in human systemic arteries. Leveraging recent advances in physics-informed machine learning [31–34], we put forth an algorithmic framework for producing physically consistent predictions of flow and pressure wave propagation directly from processing noisy and scattered measurements of blood velocity and wall displacement obtained by non-invasive 4D flow MRI. Specifically, we propose to employ deep neural networks to represent the unknown flow variables (blood velocity, wall displacement and pressure) in a given arterial network. We then train these networks to produce outputs that: (i) fit any available clinical data (typically velocity and wall displacement data at a few cross-sections), (ii) satisfy the underlying physical conservation laws as described by a reduced one-dimensional model of pulsatile blood flow [4,23], and (iii) ensure conservation and information propagation across interface points in the network (e.g., bifurcations, junctions, etc.).

In contrast to traditional computational fluid dynamics models, the proposed methodology does not involve the cumbersome generation of computational meshes, nor it requires the precise prescription of boundary conditions. Instead, it can directly use noisy and scattered measurements obtained by medical imaging at points other than the boundaries to accurately predict the pressure and recover flow information. Furthermore, as flow conditions in general do not vary greatly from one patient to another, pre-trained neural network models can be rapidly adapted to a new patient case. After the model is trained, all quantities of interest (blood velocity, wall displacement and pressure) can be predicted for every temporal or spatial point in the network. This directly enables the calculation of Windkessel parameters as a simple post-processing step, leading to a simple scheme for calibrating more detailed and expensive 3D models of blood flow.

The main contributions of this work can be summarized in the following bullet points:

- This is the first time that physics-informed neural networks are used to solve conservation laws in graph topologies (specifically, here we considered disjoint arterial domains connected by appropriate interface boundary conditions).
- In this paper the importance of non-dimensionalization and normalization is highlighted and a methodology to properly mitigate vanishing gradient pathologies in physics-informed neural networks.
- To our knowledge, this is the first time that a physics-informed neural network model is applied on an inference problem involving real noisy clinical data.
- We were able to reconstruct the spatio-temporal response of the absolute pressure wave in an arterial network, given scattered and noisy 4D flow MRI data of blood velocity and wall displacement at some cross-sections.
- This contribution is a first step towards utilizing physics-informed deep learning for performing non-invasive pressure diagnostics for cardiovascular disorders.
- The resulting surrogate also offers a simple and computationally efficient way for calibrating boundary conditions for conventional flow simulation models.

The rest of the paper is organized as follows. In Section 2, we provide an outline of the one-dimensional blood flow model employed in this work, as well as the deep learning techniques used for obtaining physically consistent predictions of pulse wave propagation directly from clinical measurements. In Section 3, we demonstrate the effectiveness of the proposed methods in a series of synthetic systematic studies, as well as a realistic clinical case involving *in-vivo* measurements near the aorta/carotid bifurcation of a healthy human subject. Conclusions and further discussions are given in Section 4. All code and data accompanying this manuscript will be made publicly available at https://github.com/PredictiveIntelligenceLab/1DBloodFlowPINNs.

2. Methods

2.1. A simplified model of pulsatile flow in arterial networks

Pulse wave propagation in arterial networks can be effectively modeled using one-dimensional (1D) reduced order models [6,23,35]. In order to achieve the order reduction, a series of assumptions need to be made [36]. First, we assume that the local curvature is small enough such that the geometry can be described using a Cartesian coordinate x, as shown in Fig. 1. Moreover, the fluid is incompressible and Newtonian, since we are considering geometries consisting of large arteries, so the density and dynamic viscosity are constant. Lastly, the structural properties of the artery are preserved at a cross-section. Following [4,36] we consider a reduced form of the incompressible Navier–Stokes equations. Conservation of mass and momentum can be expressed as a hyperbolic conservation law [36] that describes the evolution of blood velocity and cross-sectional area [4,23]. In order to close the system, a third equation accounting for the relation between the pressure and the area is used, which is derived by assuming

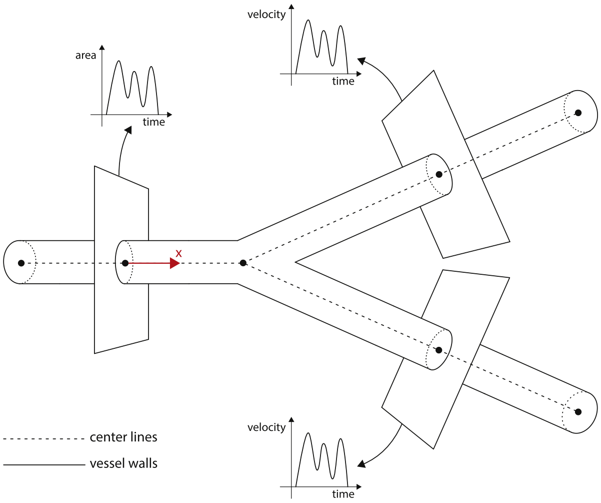

Fig. 1. *A schematic representation of a simple bifurcating arterial topology and its one-dimensional center-lines used for the reduced model.* Throughout this work, three dimensional geometries are recovered from MRI data and the corresponding center-lines are extracted by using the VMTK library [14]. Blood velocity data are typically obtained using Doppler ultrasound or 4D flow MRI, while the area data are parsed from 2D Cine images recovered by 4D flow MRI.

a thin wall tube and using Laplace's law [36]. This reduced order model provides an accurate representation of the underlying transport processes and its effectiveness in correctly capturing pulse wave propagation phenomena has been validated against both *in-vitro* and *in-vivo* data [6,24,35]. The system derived by the above analysis takes the form [4,23]

$$\begin{aligned}
&\frac{\partial A}{\partial t}+\frac{\partial Au}{\partial x}=0,\\
&\frac{\partial u}{\partial t}+\alpha u\frac{\partial u}{\partial x}+\frac{u}{A}\frac{\partial(\alpha-1)uA}{\partial x}+\frac{1}{\rho}\frac{\partial p}{\partial x}-K_R\frac{u}{A}=0,\\
&p=p_{ext}+\beta(\sqrt{A}-\sqrt{A_0}),
\end{aligned} \tag{1}$$

where $A(x,t)$, $u(x,t)$ and $p(x,t)$ represent the cross-sectional area, the velocity and pressure, respectively. In addition, x and t represent the spatial and temporal coordinate in each vessel, respectively. Also, $\beta=\dfrac{\sqrt{\pi}h_0E}{(1-\nu^2)A_0}$ and A_0 denotes the vessel's sectional area at equilibrium, h_0 the wall thickness, E the Young's modulus, p_{ext} the external pressure, ν the Poisson ratio and ρ is the density of the blood. Values for h_0, E and ν are typically mined from the literature, while p_{ext} and A_0 are typically set to the diastolic pressure and wall displacement, respectively. Moreover, K_R is a friction parameter that depends on the velocity profile chosen, while α is a momentum flux correction factor which accounts for the nonlinearity of the sectional integration in terms of the local velocity (for blood flow we typically chose $K_R=-22\mu\pi$, where μ is the blood viscosity, and $\alpha=1.1$, see [23] for more details). For the examples presented in Section 3 the β parameter is computed based on the empirical relation put forth by Olufsen et al. [29]. We should also note that the proposed one-dimensional model neglects vessel tortuosity, curvature and other three-dimensional geometry effects. In cases where these play a major role, e.g., cerebral or coronary flows, our model in its present form will be inevitably prone to inaccuracies. Several works in the literature have attempted to extend one-dimensional models to account for three-dimensional geometry effects (see for e.g., [4,37]). Such extensions could be incorporated in our framework and are left to future work as they extend beyond the scope of this paper. For all cases presented in this work, we reckon that vessel tortuosity has a secondary effect in comparison with other sources of inaccuracy/uncertainty (e.g., noise in the experimental data, uncertainty on the geometry, the wall elasticity moduli, etc.).

2.2. Physics-informed neural networks

Physics informed neural networks (PINNs) are a new paradigm that leverages recent advances in deep learning to infer solutions, parameters and/or constitutive laws involving partial differential equations (PDEs) [31,38–40]. In this framework, the solution of partial differential equations is parametrized by a neural network that is trained to match the measurements of the system, while being constrained to approximately satisfy the underlying physical laws. In our particular case, we define one neural network $f(x,t;\theta^j)$ to represent the solution of Eq. (1) for vessel $\#j$ in our arterial network. Here θ^j denotes the parameters of the neural network for vessel $\#j$. The network aims to approximate the following mapping:

$$[x,t] \xmapsto{f_{\theta^j}} [A^j(x,t), u^j(x,t), p^j(x,t)]$$

In correspondence to Eq. (1), the following residuals can be defined:

$$\begin{aligned} r_A(x,t) &:= \frac{\partial A}{\partial t} + \frac{\partial Au}{\partial x}, \\ r_u(x,t) &:= \frac{\partial u}{\partial t} + \alpha u \frac{\partial u}{\partial x} + \frac{u}{A}\frac{\partial(\alpha-1)uA}{\partial x} + \frac{1}{\rho}\frac{\partial p}{\partial x} - K_R\frac{u}{A}, \\ r_p(x,t) &:= p - p_{ext} - \beta(\sqrt{A} - \sqrt{A_0}). \end{aligned} \tag{2}$$

These residuals can be then used as constraints during the training of the neural networks $f(x,t;\theta^j)$ in order to encourage them to produce physically consistent predictions. In addition to minimizing the residuals, the neural networks are also constrained to fit any available non-invasive clinical measurements that may be noisy and scattered in space and time. In order to illustrate our model, we refer the reader to figures Fig. 2 and Fig. 3. For example, in each vessel $\#j$, we have scattered measurements of $A(x,t)$ (corresponding to the black $\times$'s in Fig. 3), namely $\{(x_i,t_i), A^j(x_i,t_i)\}$, $i=1,\ldots,N_A^j$, and $u(x,t)$ (corresponding to the red $\times$'s in Fig. 3), namely $\{(x_i,t_i), u^j(x_i,t_i)\}$, $i=1,\ldots,N_u^j$, along with a larger number of collocation points $\{(x_i,t_i), r^j_{A,u,p}(x_i,t_i)=0\}$, $i=1,\ldots,N_r^j$, that aim to satisfy the constraints at a finite set of N_r^j collocation nodes. In the examples presented in this paper, we want to emphasize that we will only use measurements for $A(x,t)$ and $u(x,t)$, but not for $p(x,t)$, as these are the quantities that often cannot be reliably measured non-invasively in the clinic.

The formulation of the one-dimensional model employed in this work introduces an explicit correlation between the absolute pressure, the cross-sectional area and the blood velocity, as shown in Eq. (1). Utilizing momentum conservation alone as a constraint is not sufficient for identifying the absolute pressure as only the pressure gradient appears in the momentum equation. However, the linear elastic constitutive law for the wall displacements (see Eq. (1)) directly relates the arterial wall displacement with the absolute pressure in each cross section. This constitutive relation is coupled to the mass and momentum conservation laws, and it is exactly these correlations that we aim to exploit via the use of physics-informed neural networks in order to infer the absolute pressure from velocity and cross-sectional area measurements. Of course the sensible choice of sensible values for p_{ext} and A_0 plays a role in the convergence speed of the proposed algorithm, but this choice is not unique. In the synthetic examples presented in this work p_{ext} was set to zero and A_0 was set to a reference value at equilibrium. For the realistic aorta/carotid bifurcation case the values of p_{ext} and A_0 in each vessel were chosen according to nominal values for the diastolic pressure and area found in the literature (see e.g. [41]). More details on how to construct the respective terms of the loss function for training the neural networks $f(x,t;\theta^j)$ will be given in the next section.

2.3. Loss function: Measurements

This part of the loss function corresponds to fitting the clinical measurements obtained for some of the vessels in the network. In Fig. 3 these are denoted as black and red $\times$'s, the color depending on the quantity we have measurement for. In this particular case, this term encourages the output of the neural network A and u to match the measurements of area and velocity obtained by a clinical procedure (e.g., segmenting 2D cine images and Doppler ultrasound [17]). This part of the loss function has the form (take vessel $\#j$ as example):

$$\mathcal{L}^j_{\text{measurements}} = \frac{1}{N_u^j}\sum_{i=1}^{N_u^j}(u^j(x_i,t_i) - u^j(x_i,t_i;\theta^j))^2 + \frac{1}{N_A^j}\sum_{i=1}^{N_A^j}(A^j(x_i,t_i) - A^j(x_i,t_i;\theta^j))^2, \quad j=1,\ldots,D_M \tag{3}$$

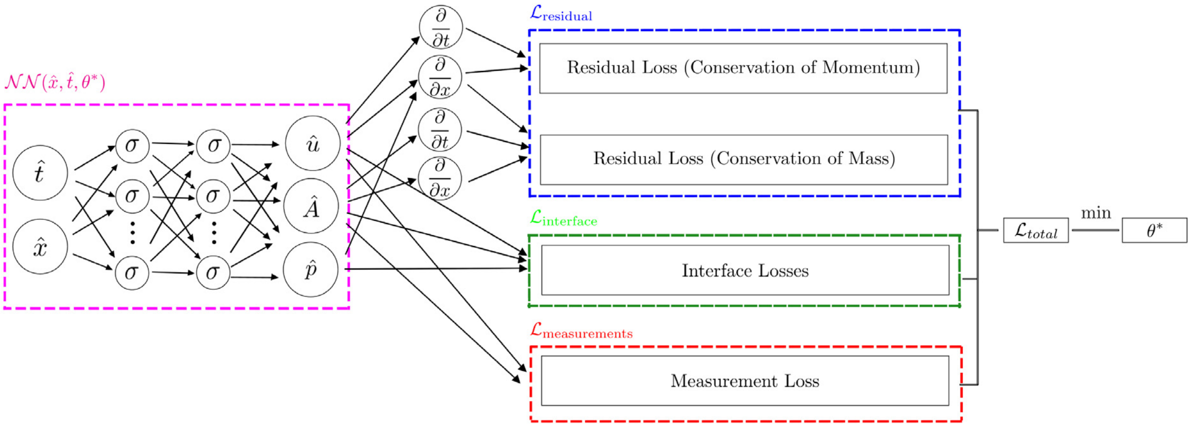
$\mathcal{NN}(\hat{x},\hat{t},\theta^*)$
$\hat{t}$
$\hat{x}$
σ
$\hat{u}$
$\hat{A}$
$\hat{p}$
$\frac{\partial}{\partial t}$
$\frac{\partial}{\partial x}$
$\mathcal{L}_{\text{residual}}$
Residual Loss (Conservation of Momentum)
Residual Loss (Conservation of Mass)
$\mathcal{L}_{\text{interface}}$
Interface Losses
$\mathcal{L}_{\text{measurements}}$
Measurement Loss
$\mathcal{L}_{total}$
min
θ^*

Fig. 2. *A schematic representation of the proposed algorithm.* A diagram describing the process followed in the proposed method is presented. The blue colored box represents the part of the loss function that corresponds to the residual losses, the green colored box represents the part of the loss function that corresponds to the interface losses and the red colored box represents the measurement loss. The different parts construct the complete loss function, by minimizing which the neural network parameters are tuned. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

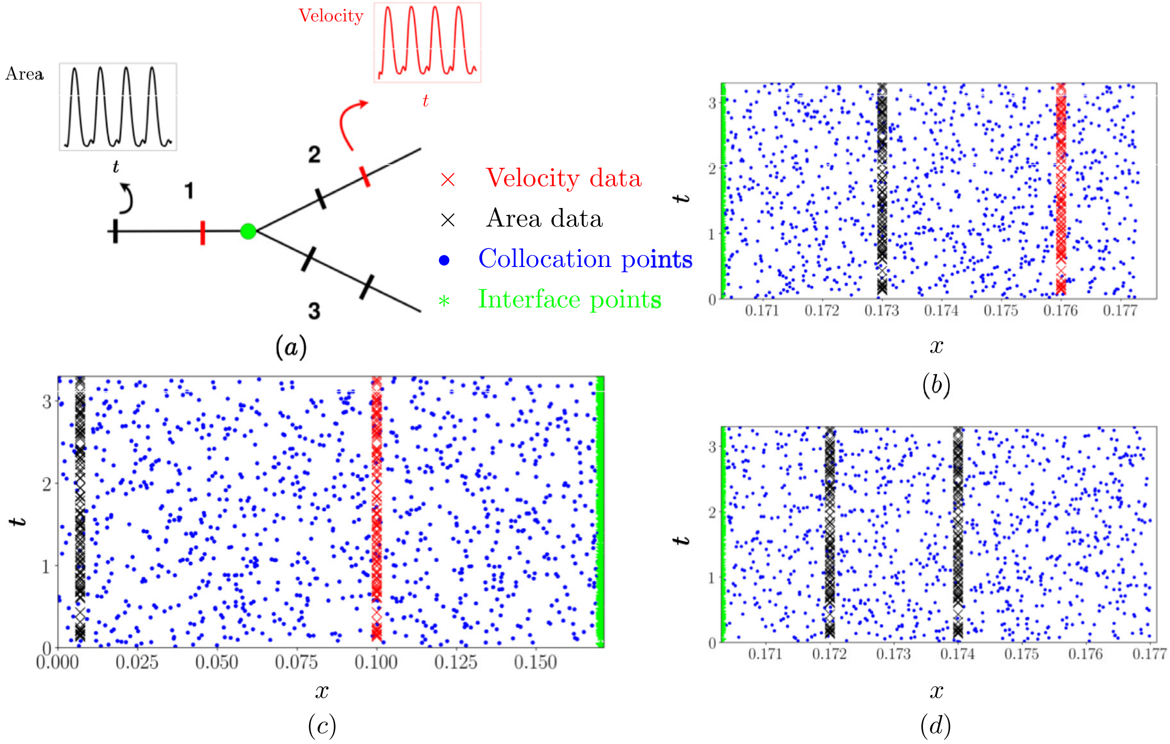

Fig. 3. *Graph of a prototype arterial network where 1 vessel splits into 2 vessels at the bifurcation point:* (a) Structure of the simple arterial network consisting of 3 vessels with 1 bifurcation point. (b) Domain of vessel #2. (c) Domain of vessel #1. (d) Domain of vessel #3. Blue points represent the collocation points. Green stars represent the interface points at the bifurcation. Red crosses indicate the locations of the velocity measurements. Black crosses show the locations for the area measurements. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

where N_A^j, N_u^j represent the number of measurements for A and u in vessel $\#j$ respectively. Also, D_M denotes total number of vessels in which we have measurements. In the above equation $u^j(x_i, t_i; \theta^j)$ and $A^j(x_i, t_i; \theta^j)$ represent the outputs given the parameters of the neural network for vessel $\#j$. Minimizing this term will encourage the neural networks to fit the available measurements.

2.4. Loss function: Collocation points

This part of the loss function corresponds to the collocation points, see the blue dots in Fig. 3. These are points that we randomly choose inside the arterial domains using a latin-hypercube sampling strategy [42]. Over these collocation points, we impose the physical constraints by encouraging the right hand side of Eq. (2) to be equal to zero. The partial derivatives in the residual expression (Eq. (2)) can be computed using automatic differentiation [43,44]. This objective is imposed to encourage the neural networks to find a particular set of parameters that make their predictions consistent with the underlying differential equations, which translates to having the minimum residual. By satisfying this condition together with matching the measurements at particular points in Section 2.3, we can obtain a model that is capable of inferring the solution at any spatial coordinate of the domain and any time. This collocation loss function takes the following form (take vessel $\#j$ as example):

$$\mathcal{L}_{\text{residual}}^j = \frac{1}{N_r^j}\sum_{i=1}^{N_r^j}(r_A^j(x_i, t_i; \theta^j))^2 + \frac{1}{N_r^j}\sum_{i=1}^{N_r^j}(r_u^j(x_i, t_i; \theta^j))^2 + \frac{1}{N_r^j}\sum_{i=1}^{N_r^j}(r_p^j(x_i, t_i; \theta^j))^2, \qquad j = 1, \dots, D \tag{4}$$

where N_r^j represents the number of collocation points in vessel $\#j$, meaning many blue dots we have in Fig. 3. Also, D denotes the total number of vessels in our arterial network. The terms $r_A^j(x_i, t_i; \theta^j)$, $r_u^j(x_i, t_i; \theta^j)$ and $r_p^j(x_i, t_i; \theta^j)$ represent the residual of area, velocity and pressure defined in Eq. (2), respectively.

2.5. Loss function: Interfaces

One-dimensional models can be extended to treat splitting and merging arteries by imposing proper boundary conditions [36]. In this case, one needs to provide boundary conditions for each artery at the interface points to ensure conservation. In conventional numerical methods (e.g., Discontinuous Galerkin method [23,45]), the velocity u and the area A can be discontinuous at the interface points (e.g., bifurcations, junctions), so in order to find the values, a Riemann problem has to be solved. This is typically done by employing the characteristic variables of the hyperbolic system of equations (1), accounting for a decoupled system of scalar equations, so that the traveling waves can reach the splitting points [23]. The proposed method can work without using information on the characteristics, instead just imposing the momentum and mass conservation equations suffices. To illustrate our workflow, let us consider the case where the artery #1 splits to #2 and #3, see Fig. 3. For each vessel j, the area, velocity, pressure and density are denoted as $[A^j, u^j, p^j]$, the spatial and temporal variables are denoted as $[x_i, t_i]$. In reference to Fig. 3, this corresponds to the green stars and the i index on how many of these we use to impose the continuity. The boundary conditions at the bifurcation points are derived by the conservation of momentum and mass [36] for the parent and daughter vessels:

$$\begin{aligned} &A_1u_1 = A_2u_2 + A_3u_3,\\ &p_1 + \frac{1}{2}\rho u_1^2 = p_2 + \frac{1}{2}\rho u_2^2,\\ &p_1 + \frac{1}{2}\rho u_1^2 = p_3 + \frac{1}{2}\rho u_3^2. \end{aligned} \tag{5}$$

In the case of more than one bifurcations, these conditions are imposed to every splitting point using the appropriate notation, depending on the numbering of the domains. In the case of junctions, instead of bifurcations, or mixed change of geometry, system (Eq. (5)) takes the form of the appropriate condition following the conservation laws. In general, the interface loss has the form (take bifurcation point $\#k$ as example):

$$\mathcal{L}_{\text{interfaces}}^k = \frac{1}{N_b^k}\sum_{i=1}^{N_b^k}(A_1^k(x_k, t_i; \theta_1^k)u_1^k(x_k, t_i; \theta_1^k) - A_2^k(x_k, t_i; \theta_2^k)u_2^k(x_k, t_i; \theta_2^k) - A_3^k(x_k, t_i; \theta_3^k)u_3^k(x_k, t_i; \theta_3^k))^2 \tag{6}$$

$$+ \frac{1}{N_b^k} \sum_{i=1}^{N_b^k} (p_1^k(x_k, t_i; \theta_1^k) + \frac{1}{2} u_1^k(x_k, t_i; \theta_1^k)^2 - p_2^k(x_k, t_i; \theta_2^k) - \frac{1}{2} u_2^k(x_k, t_i; \theta_2^k)^2)^2 \tag{7}$$

$$+ \frac{1}{N_b^k} \sum_{i=1}^{N_b^k} (p_1^k(x_k, t_i; \theta_1^k) + \frac{1}{2} u_1^k(x_k, t_i; \theta_1^k)^2 - p_3^k(x_k, t_i; \theta_3^k) - \frac{1}{2} u_3^k(x_k, t_i; \theta_3^k)^2)^2,$$

$$k = 1, \ldots, D_I \tag{8}$$

where the indices 1, 2, 3 in $\mathcal{L}_b^k$ denote the father and daughter vessels, respectively, at each bifurcation. Also, D_I denotes the total number of bifurcation points in our arterial network. N_b^k represents the number of collocation points on the interface boundaries, i.e. how many green stars we have in Fig. 3. So, in the above equation if we choose for example p_1^1 that means we calculate the pressure at the interface point with index $k = 1$ using the network corresponding to the father vessel, in the notion of Fig. 3 this would correspond to domain 1. For the interface, we feed the neural network the inputs $[x_k, t]$, where x_k is the coordinate of the interface point. Minimizing the interface loss encourages the neural network to satisfy the conservation laws at the interface points.

The interface loss function ensures the information of the measurements in one domain can be propagated throughout the neighboring domains. In the original formulation of Sherwin et al. [36] a Riemann problem was solved on the interfaces due to the hyperbolic nature of the wave propagation equations. Here we have experimented with including a constraint ensuring the propagation of characteristic variables at interfaces, but we did not observe any substantial improvement in terms of accuracy and computational efficiency, thus we choose to not include these additional constraints in the neural networks' loss function.

2.6. *Loss function: Accounting for all constraints*

In the previous sections, we provided details on how to construct the individual terms of the loss function corresponding to the measurements, physical constraints and continuity, that the model should satisfy. In this section, we will show how to combine these individual terms to create the joint form of the loss function, together with some examples to clarify the procedure. The loss function of our physics informed neural networks (PINNs) takes the form:

$$\mathcal{L} = \sum_{j=1}^{D_M} \mathcal{L}_{\text{measurements}}^j + \sum_{j=1}^{D} \mathcal{L}_{\text{residual}}^j + \sum_{k=1}^{D_I} \mathcal{L}_{\text{interfaces}}^k, \tag{9}$$

according to the definition in previous section, D represents the total number of domains, $D_M \subset D$ denotes the indices of domains where we have data and D_I corresponds to the number of interfaces in the arterial networks. The index j denotes the identity of the domain under consideration and the index k is helping in keeping track of the interface points. Index k does not refer to specific neural network, because A, u and p on the bifurcation points are calculated through the father and daughter vessels by their respective network. This complete form of the loss function that encourages the model to learn the underlining physics of the problem by using measurements inside the domain, not necessarily at the boundaries e.g., Fig. 3, and by respecting the imposed physical constraints. We will present how the loss function is constructed for some simple cases in order for the method to be more clear to the reader.

For a single artery, the loss function for our method is the following:

$$\mathcal{L} = \mathcal{L}_{\text{measurements}} + \mathcal{L}_{\text{residual}}. \tag{10}$$

In this particular geometry there are no bifurcations, so $k = 0$, thus the term corresponding to splitting, $\mathcal{L}_{\text{interface}}^k$, is neglected.

For the case that we have three vessels and one bifurcation, see Fig. 3, we specify the loss function as:

$$\mathcal{L} = \sum_{j=1}^{3} \mathcal{L}_{\text{measurements}}^j + \sum_{k=1}^{1} \mathcal{L}_{\text{interfaces}}^k + \sum_{j=1}^{3} \mathcal{L}_{\text{residual}}^j, \tag{11}$$

where the first summation corresponds to the domains where we have measurements for the velocity and cross-sectional area. The second summation encourages the model to satisfy the continuity equation on the bifurcation

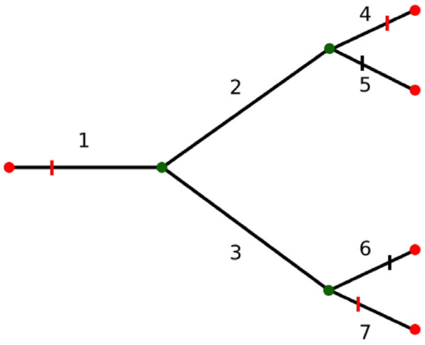

Fig. 4. *Graph of a complex arterial network which involves 7 vessels and 3 bifurcation points:* The red points represent the boundaries of the arterial networks. The gray points represent the bifurcation points where we impose the continuity boundary conditions. The number on each vessel is the corresponding label for this vessel. The vertical lines denote the middle points of the vessels and also the spatial locations for which we have training data, e.g., for the velocity (red) and the wall displacement (black). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

point. The third summation represents the physics-informed constraints in all three vessels. The second term is written as a summation, for consistency, but in this case, $k = 1$, so there is only one component of $\mathcal{L}^k_{\text{interface}}$.

For a network with 7 vessels and 3 bifurcations, see Fig. 4, we specify the loss function as:

$$\mathcal{L} = \sum_{j=1}^{7} \mathcal{L}^j_{\text{measurements}} + \sum_{k=1}^{3} \mathcal{L}^k_{\text{interfaces}} + \sum_{j=1}^{7} \mathcal{L}^j_{\text{residual}}. \tag{12}$$

Here, the first summation corresponds to the seven vessels where we have measurements of the velocity and area. The second summation correspond to the three bifurcation points. The third summation is representing the physics-informed constraints on seven different arteries. The loss is minimized by encouraging the model to reconstruct the measurements provided at certain points where the measurements are obtained and to respect the constraints imposed by the physics of the problem, e.g., the differential and continuity equations. The loss function in Eq. (12) is minimized using stochastic gradient descent algorithm [46] or its modern variants [47,48] that are widely used in current deep learning research.

2.7. Non-dimensionalization and normalization

In Eq. (1), the order of magnitude of the different physical quantities, velocity, cross-sectional area and pressure, have a significant relative difference, e.g., $P \sim 10^6$ Pa, $A \sim 10^{-5}$ m^2 and $u \sim 10$ m/s, which casts great difficulty on the training of the neural network [49]. The significant difference in magnitude of the parameters creates a systematic problem for the training of the physics-informed neural network, as this difference in scales will have a severe impact on the magnitude of the back-propagated gradients that adjust the neural network parameters during training. To overcome this problem, we employ a non-dimensionalization and normalization technique with the purpose of scaling the input and the output of the neural networks in a proper scale (e.g., $(\hat{A}, \hat{u}, \hat{p}, x_*, t_*) \sim \mathcal{O}(1)$) and normalizing the spatial and temporal coordinates to have zero mean and unit variance for training the neural networks more efficiently. Although there could be a way to weight the components of the loss function to mitigate the bias casted into the loss function due to this discrepancy across scales, this process would require a lot of guess-work and tuning. On the other hand, the proposed non-dimensionalization strategy achieves the goal of normalizing the variables in a physically justified and intuitive manner that adheres to the requirements of standard neural net initialization strategies (e.g., Xavier initialization) and yields a robust workflow that is free from ad-hoc hacks and guesswork. For the purpose of non-dimensionalization we introduce characteristic variables, which are commonly used in multi-scale physics modeling [50] in order to simplify the equations. For this problem we need a characteristic length L and a characteristic velocity U.

We will choose the characteristic length to be the square root of the mean of the equilibrium cross-sectional area of the network vessels. In order to choose the characteristic velocity we make use of the physiological condition that the wave speed in a vessel has to be one order of magnitude larger than the length [23], given that, in the normalized length case $c = 10$. Thus:

$$L = \sqrt{\frac{1}{N}\sum_{j=1}^{D}(A_0^j)}, \qquad U = 10,$$

where $j = 1, \ldots, D$. At this point we define the quantities:

$$\hat{u} = \frac{u}{U}, \quad \hat{A} = \frac{A}{A^o}, \quad \hat{p} = \frac{p}{p_0}, \quad x_* = \frac{x}{L}, \quad t_* = \frac{t}{T}, \tag{13}$$

where $p_0 = \rho U^2$, $T = \dfrac{L}{U}$ and $A^o = L^2$.

Now, all dependent variables take values with $\mathcal{O}(1)$ magnitude. The next step will be performed in accordance to standard deep learning practices for ensuring the robustness of the neural network training via back-propagation. Specifically, according to the seminal work of Glorot and Bengio [49], normalizing the input to have zero mean and unit variance mitigates the pathology of vanishing gradients in deep networks. Regarding the choice of activation function, given the relatively shallow networks used in this work, we would expect that the use of a hyperbolic tangent activation should lead to robust results [51].

Thus, in this step we will apply the above mentioned technique to x_* and t_*, in the vessel $\#j$, and get:

$$\hat{x}^j = \frac{x_*^j - \mu_{x_*}^j}{\sigma_{x_*}^j}, \quad \hat{t} = \frac{t_* - \mu_{t^*}}{\sigma_{t_*}}, \tag{14}$$

where $\mu_{x_*}^j$, μ_{t_*} are the mean values and $\sigma_{x_*}^j$, σ_{t_*} the standard deviation of the spatial and temporal coordinates for the vessel $\#j$, respectively, and $\hat{x}^j$, $\hat{t}$ the scaled inputs. Using the variables stated above, we derive the updated system of equations (take vessel $\#j$ as example):

$$\begin{aligned}
&\frac{1}{\sigma_{t_*}}\frac{\partial \hat{A}^j}{\partial \hat{t}} + \frac{1}{\sigma_{x_*}^j}\hat{A}^j\frac{\partial \hat{u}^j}{\partial \hat{x}^j} + \frac{1}{\sigma_{x_*}^j}\hat{u}^j\frac{\partial \hat{A}^j}{\partial \hat{x}^j} = 0,\\
&\frac{1}{\sigma_{t_*}}\frac{\partial \hat{u}^j}{\partial \hat{t}} + \frac{1}{\sigma_{x_*}^j}\alpha\hat{u}^j\frac{\partial \hat{u}^j}{\partial \hat{x}^j} + \frac{1}{\sigma_{x_*}^j}\hat{u}^j\frac{\partial(\alpha-1)\hat{u}^j\hat{A}^j}{\partial \hat{x}^j} + \frac{1}{\sigma_{x_*}^j}\frac{\partial \hat{p}^j}{\partial \hat{x}^j} - \frac{K_R}{LU}\frac{\hat{u}^j}{\hat{A}^j} = 0,\\
&\hat{p}^j = \frac{1}{p_0}(p_{ext} + \beta(\sqrt{\hat{A}^j A^o} - \sqrt{A_0})), \qquad j = 1, \ldots, D.
\end{aligned} \tag{15}$$

In this non-dimensional and normalized form, all the variables and inputs are scaled to order $O(1)$. This is what effectively enables the training of our physics-informed neural networks in this complex setting. Likewise, at the predicting stage, we first scale the inputs by the characteristic variables as Eq. (13), i.e. x by L, then normalize them following Eq. (14). Finally, we revert the predicted quantities $(\hat{A}^j, \hat{u}^j, \hat{p}^j)$ to their original form (A^j, u^j, p^j) by multiplying with the characteristic variables i.e. $p = \hat{p}p_0$.

In order to be consistent with the derivation above, we have to follow the same procedure for every condition we impose to the model. In this notion we derive the non-dimensional continuity conditions, by inserting the non-dimensionalizing quantities into the conservation laws. By doing so, we get:

$$\hat{A}_1\hat{u}_1 = \hat{A}_2\hat{u}_2 + \hat{A}_3\hat{u}_3, \tag{16}$$

$$\hat{p}_1 + \frac{1}{2}(\hat{u}_1)^2 = \hat{p}_2 + \frac{1}{2}(\hat{u}_2)^2, \tag{17}$$

$$\hat{p}_1 + \frac{1}{2}(\hat{u}_1)^2 = \hat{p}_3 + \frac{1}{2}(\hat{u}_3)^2. \tag{18}$$

The above system of equations must be satisfied at any interface, which is achieved by the second term in the loss function in Eq. (9) that forces the model to respect the conservation laws. Finally, the non-dimensional equations (15)–(18) define the optimization objectives that are used in Eq. (9) for minimizing the residual at the collocation, bifurcation and measurement points respectively. As mentioned above, we have to multiply the predictions of the network by the scaling parameters $p = \hat{p}p_0$, $u = \hat{u}U$ and $A = \hat{A}A^0$, when doing inference or else we get a scaled version of the solutions.

2.8. Windkessel model parameter identification as post processing

A crucial aspect in calibrating computational models of blood flow is related to the proper choice of boundary conditions. Due to the excessive computational cost of imaging and performing pulsatile flow simulation across the entire human arterial tree, typically only truncated topologies are considered. This introduces the need of modeling the neglected downstream dynamics in smaller arteries and arterioles; a task that is typically accomplished using lumped parameter Windkessel models [27]. There exist other families of boundary conditions, such as coronary boundary conditions [52] and closed-loop boundary conditions [53], which are important but not employed in this work. Windkessel models provide us with a way to account for arterial blood pressure in terms of the compliance of the large elastic arteries and the resistance of the smaller arteries, by making an analogy with an electric circuit. In this setup, the compliance plays the role of a capacitor and the resistance of a resistor and the whole circuit helps to correctly model the influence of the small downstream vessels to the blood circulation. In our case we consider the popular three-element Windkessel model with governing equation (take vessel #j as example):

$$p^j + R^j C^j \frac{dp^j}{dt} - (R^j + Z^j)Q^j - p_{\text{inf}} - R^j C^j Z^j \frac{dQ^j}{dt} = 0, \qquad j = 1, \ldots D_o, \tag{19}$$

where D_o represents the total number of outlets where we want to identify the parameters. In the equation above, $Q^j = A^j u^j$ is the flow rate at the outlet, $p_{\text{inf}} = 666.5$ Pa denotes the downstream pressure that is constant for all vessels. Also, Z^j represents the characteristic impedance [54] that can be easily computed using $Z^j = \frac{\rho c_0^j}{A_0^j}$, C^j the total arterial compliance and R^j the systemic vascular resistance. The characteristic impedance is chosen in a way that allows the incoming wave to reach R^j and C^j without being reflected [54]. The main challenge here relates to choosing the compliance and resistance parameters C^j and R^j such that physiologically correct results are sought.

Some of the work done in estimating Windkessel model parameters is based on modeling assumptions. Grinberg et al. [28] proposed a method of imposing a simple resistance boundary condition using the ratio of flow rates of terminal vessels. They also showed that this method is robust and stable and scales to large arterial systems. The accuracy can be guaranteed by choosing appropriate values for the resistance at each outlet. Ismail et al. [55] proposed a method for tuning the parameters of a three-element Windkessel parameter for a coupled 3D/0D system by implementing an inverse analysis based on adjoint methods, using the patient-specific pressure information as a target. Spilker et al. [56] proposed a multidomain method that gets coupled with a reduced order, a three-element Windkessel, model in which by imposing characteristic information of input pressure, one can tune hemodynamic simulations. The pressure characteristics, such as minimum or maximum pressure, are imposed in the input of the domain. Also Spilker et al. [57] proposed a procedure for imposing and tuning impedance type boundary conditions by adjusting the length of morphometry produced trees to match the outflow. Yu et al. [58] presented an online estimator based on extended Kalman filter that uses sets of measurements to estimate state variables and at the same time recover the model parameters. Pant et al. [59] presented an algorithm that iterates between a three dimensional and a zero-dimensional model with known pressure, at specific points, to recover the values of three-element Windkessel model using Unscented Kalman filters. Schiavazzi et al. [60] proposed a method for automatically estimating Norwood circulation model parameters and their uncertainty related to the clinical measurements. In this framework an analysis over the confidence of the estimations can be performed using a differential evolution adaptive Metropolis estimation and patient specific measurements on pressure and flow.

All the aforementioned works require the precise prescription of inflow and boundary conditions and rely on the repeated evaluation of conventional solvers for different combinations of R^j and C^j until a parametric setting that yields a reasonable match to clinical data is identified. For networks with more than a handful of outlets this defines a tedious and prohibitively costly calibration procedure. In contrast, our proposed methodology can yield a cheap and effective procedure for calibrating Windkessel model parameters without the need of employing conventional simulators. Specifically, the training of the proposed physics-informed neural networks (as described in Section 2) does not require the a-priori specification of Windkessel-type outflow boundary conditions. Moreover, once the neural networks are trained on scattered velocity and wall displacement data, they can yield physically consistent predictions for the propagating pressure wave. This information can be then used for identifying the resistance and compliance parameters C^j and R^j at each outlet by solving the differential equation (19) as a post-processing step.

To this end, here we propose a simple method for identifying the parameters R^j and C^j in Eq. (19) by adaptive grid search given the predicted outflow data p^j and Q^j produced by the physics-informed neural networks representing the flow solution at each outlet. Once the model is trained we can predict p^j and Q^j at every location in the domain and at any times. Thus we have access to the data as a time series $[p_0^j, \ldots, p_{N_t-1}^j]$ and $[Q_0^j, \ldots, Q_{N_t-1}^j]$ via the neural network output, where N_t is the length of the time series. We select a periodic part of the data and fit Q^j using Fourier series consisting of fifty modes and then compute the time derivative of Q^j with respect to time t. Thus, we are setting up a post processing step that leverages the solution for Q^j and its derivative $\frac{dQ^j}{dt}$ at each temporal discretization and use the ODE solver to compute $\tilde{p^j}$ using some combination of parameters (R^j, C^j). In this case, we compare the difference between $\tilde{p^j}$ and the pressure predicted by the model. Adaptive mesh refinement of the parameter space can be used to discover the optimal solution for (R^j, C^j) by iteratively exploring new (R^j, C^j) meshes after computing the loss function. We define the loss function as a function of (R^j, C^j) at N_t points $[t_1, \ldots t_{N_t}]$ randomly selected within the interval $t \in [0, T]$:

$$\text{Loss}(R^j, C^j) = \frac{1}{N_t}\sum_{m=1}^{N_t} \|\tilde{p}^j(t_{N_t}) - p^j(t_{N_t})\|^2 \tag{20}$$

In this case, we compute $\text{Loss}(R^j, C^j)$ at the parameter grid and find the set of values for which this loss gets minimized. Then we explore an new interval $R^j_{new} \in [R^j_{optimal} - 0.5R^j_{optimal}, R^j_{optimal} + 0.5R^j_{optimal}]$, $C^j_{new} \in [C^j_{optimal} - 0.5C^j_{optimal}, C^j_{optimal} + 0.5C^j_{optimal}]$ and refine the mesh in this vicinity. This is done for a number of consecutive times. In our case, we consider the number of consecutive times as 5, in order to reach the parameter set that provides us with the smallest value of loss in Eq. (20). The computational time required for discovering the Windkessel parameters is about 10 min per outlet using alaptop Thinkpad P52s (CPU: Intel i7-8650U, RAM: 32 GiB, GPU: NVIDIA Quadro P500 2GiB).

2.9. *Computational cost*

The cost of our computational workflow is primarily dominated by the training time of the physics-informed neural networks, which, for the largest geometry considered in this work, summed up to approximately 7 h on a single NVIDIA Tesla P100 GPU card (starting from a random network initialization). For each arterial topology this can be considered as a one-time offline cost, that could be further reduced by leveraging state-of-the-art GPU hardware and parallel multi-GPU implementations. Moreover, transfer learning techniques can be employed to reduce the training cost for subsequent simulations (e.g. different patient data-sets, slightly different flow conditions, parametric studies, etc.), as pre-trained networks can be used to initialize new simulations (instead of using a random initialization). The cost of performing predictions with a trained neural network is also negligible compared to its training cost, as the prediction step merely consists of a few matrix multiplications that can be completed in a fraction of a second on a GPU card. Finally, the 10 min cost of calibrating Windkessel model parameters at each outlet using a trained neural network as a surrogate is small compared to the total training cost, and is trivially parallelizable across multiple outlets. This cost is also reasonable in comparison to established approaches for parameter calibration that typically involve the repeated solution of forward partial differential equations models in an adjoint optimization loop, or employ filtering techniques that require the availability of invasive pressure measurements.

3. Results

The proposed method will be tested in three different cases. In the first case, we consider a prototype arterial network resembling a carotid bifurcation that consists of 3 vessels with 1 bifurcation as shown in Fig. 3. In the second case, we consider a complex arterial network resembling an idealized pelvic geometry that consists of 7 vessels and 3 bifurcations as shown in Fig. 4. For both of these two cases the data used to train the model are synthetic data generated using a conventional Discontinuous Galerkin simulator [23]. More specifically, the data set is created by considering the results of the Discontinuous Galerkin method for velocity and wall displacement at a steady state cardiac cycle. For the first case, we also provide a series of comprehensive systematic studies in order to quantify the accuracy and robustness of the proposed methods. Finally, we also present results for a real clinical case utilizing 4D flow MRI data collected near the aorta/carotid bifurcation of a healthy human subject. In all of our numerical studies the viscous losses coefficient K_R was set to zero and α to one under the assumption

Table 1
Neural network parameter choices for different examples.

Example	# layers/ # neurons per layer	Learning rate	Batch size	Number of iterations
Y-shaped	7/100	$10^{-3}/10^{-4}$	1024	90,000/40,000
Pelvic	7/100	$10^{-3}/10^{-4}$	1024	280,000/40,000
Real noisy data	7/100	$10^{-3}/10^{-4}$	1024	75,000/35,000

In Table 1 the network structure, learning rate, batch size and number of iterations are presented for all the examples provided in the manuscript. For the learning rate and number of iterations parameters the slash character serves as an indication that the learning rate changes from $\eta = 10^{-3}$ to $\eta = 10^{-4}$ after the specified iteration number.

Table 2
Generation of synthetic training and validation data for the prototype Y-shaped bifurcation.

Arterial segment	Length (cm)	Peripheral resistance (10^{10} Pa s m^{-3})	Peripheral compliance (10^{-10} m^{3}Pa^{-1})	β (Pa/m)	Equilibrium cross-sectional area (m^2)
1	17.03	–	–	6.97e+07	1.36e−05
2	0.7	0.5251	0.3428	5.42e+08	1.81e−06
3	0.67	0.2702	0.6661	6.96e+07	1.36e−05

Physiological data used as simulation parameters for the discontinuous Galerkin method.

that viscous losses play a secondary role in the larger systemic arteries considered in this work.The proposed algorithms are implemented in Tensorflow v1.10 [44], and computations were performed in single precision arithmetic on a single NVIDIA Tesla P100 GPU card. The choice of single over double precision arithmetic will be justified in a subsequent section. All neural networks architectures and hyper-parameters employed in each example are summarized in table Table 1. The neural network models for the first and the third case was trained for, approximately, 1 h and 20 min and for the second case for approximately 7 h. All code and data accompanying this manuscript will be made publicly available at https://github.com/PredictiveIntelligenceLab/1DBloodFlowPINNs.

3.1. Flow through a prototype Y-shaped bifurcation

3.1.1. Arterial network topology and observed measurements

Here we consider a simple arterial network resembling a prototype carotid bifurcation. The network consists of 3 vessels with 1 bifurcation where each vessel has lengths $L_1 = 170.3$ mm, $L_2 = 7$ mm, $L_3 = 6.7$ mm, respectively, with one bifurcation point at $x_b = 170.3$ mm. Somewhere within these vessels we assume we have access to time-series data for the blood velocity and the wall displacement, as is shown in Fig. 3. Specifically, we assume $N_u = N_A = 413$ measurements at the inlet and outlets of the arterial network for $t \in [0, 3.3]$ s which represents approximately four cardiac cycles with a period of $T = 0.8$ s. We should emphasize that we do not assume any measurements for the pressure at any location in the arterial network.

3.1.2. Generation of synthetic training and validation data

For this example we created a set of synthetic data for the cross sectional area and velocity using an in-house Discontinuous Galerkin (DG) solver [23,30,61]. Out of the resulting waveforms we choose 4 cycles from the steady state solution as training data for the neural network model. For the DG simulation, we considered nominal values for the blood density $\rho = 1060$ Kg/m^3 and the blood viscosity $\nu = 3.5$ mPa s, respectively [62]. Moreover we provide the DG solver with the Windkessel and arterial wall parameters shown in Table 2.

3.1.3. Neural network approximation set-up

We parametrize the solution of this problem using three neural networks, one for each vessel. Each of these networks consists of 7 hidden layers with 100 neurons per layer, followed by hyperbolic tangent activation function. The neural networks are initialized using Xavier initialization [49]. This architecture has enough capacity to

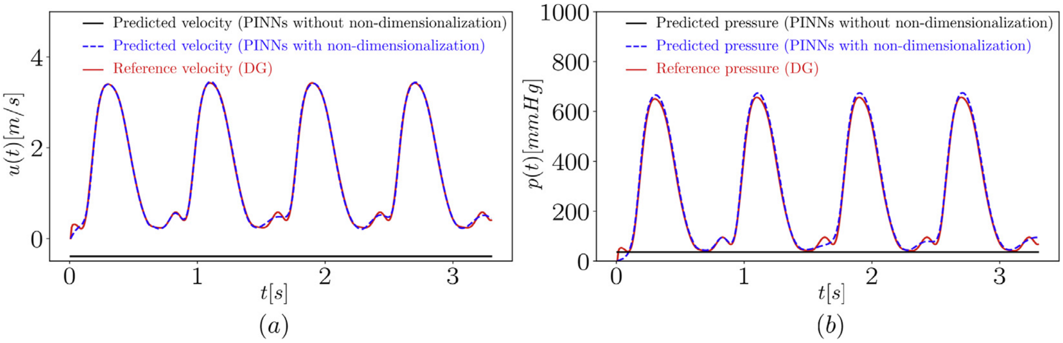

Fig. 5. *Flow through a prototype Y-shaped bifurcation:* (a) Comparison of predicted velocity wave between Discontinuous Galerkin (*red*), physics-informed neural networks with non-dimensionalization (*blue*) and physics informed neural networks without non-dimensionalization (*black*) at the middle point of vessel #1. (b) Comparison of predicted pressure wave between Discontinuous Galerkin (*red*), physics-informed neural networks with non-dimensionalization (*blue*) and physics-informed neural networks without non-dimensionalization (*black*) at the middle point of vessel #1. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

efficiently capture fine features in the propagating waveforms. A more detailed discussion on the neural network structure can be found in Section 3.1.5 where we present a systematic study that justifies the aforementioned choices in terms of balancing the trade-off of computational cost and predictive accuracy. The neural networks take the scaled inputs $[\hat{x}, \hat{t}]$ and predict the non-dimensional outputs $[\hat{A}, \hat{u}, \hat{p}]$, as discussed in Section 2.7. We are providing each neural network with velocity and cross-sectional area time series for one point at the inlet and the two outlets of the arterial network. Using this data we infer the solution at $N_r = 2000$, randomly chosen (using Latin hypercude sampling [42]) collocation points, inside each domain, and $N_b = 1024$ interface points. Moreover we randomly choose batches of $N_{batch} = 1024$ points as input to the Adam optimizer [47] and learning rate $\eta = 10^{-3}$ for the first 90,000 iterations. Consequently, we set $\eta = 10^{-4}$ for the following 40,000 for further minimizing the error and fine-tuning the neural network parameters.

3.1.4. Comparison of model predictions and reference solution

We first demonstrate the importance of the non-dimensionalization and normalization of the governing equations, as described in Section 2.7. In order to compute the unknown quantities we assume that we have access to the values of $A(t)$ and $u(t)$ for all times at one spatial point in each vessel, Fig. 3. We show in Fig. 5 that the original methodology introduced in [31,38,39] fails to overcome the difficulty of handling the different scales and thus the numerical prediction collapses. As we mentioned in Section 2.7, this phenomenon occurs due to the significant difference between the orders of magnitude of the model variables, i.e. $\beta \sim 10^8$ Pa/m, $\rho \sim 10^3$ kg/m^3, $A \sim 10^{-5}$ m^2 and $u \sim 10$ m/s, which casts great difficulty on the training of the neural network. We, also, show that the proposed normalization and non-dimensionalization tools significantly improve the model training capabilities and, thus, provide accurate predictions.

The results produced by our model for the middle points of each domain and randomly selected values of t are presented in Fig. 6. Evidently, there is a good match between the solution acquired from Discontinuous Galerkin solver [45] and the one predicted by the proposed physic-informed neural networks model. In Figs. 5 and 6 we observe that the secondary peaks are not well captured by the model predictions. We would like to point out that this discrepancy is not related to underlying physiological conditions, as for both computational methodologies the same propagation equations are used, but is mainly attributed to the representation power of the proposed algorithm. The neural network is probably not expressive enough to capture the small details of the wave-form. In our experience, that is a general trait of physics-informed neural network models which often present difficulties in capturing solutions that exhibit multi-scale features. Addressing these shortcomings is still an open area for research and the current best practice for addressing them involves tuning the neural network architecture and/or the optimization algorithm. The prediction error is further quantified in the systematic studies presented in Section 3.1.5.

To highlight the role of the continuity equation described in Section 2.5, we present a comparison between the left hand side and the right hand side of the continuity equations (16) and (17), (18) in Fig. 7. We can infer from

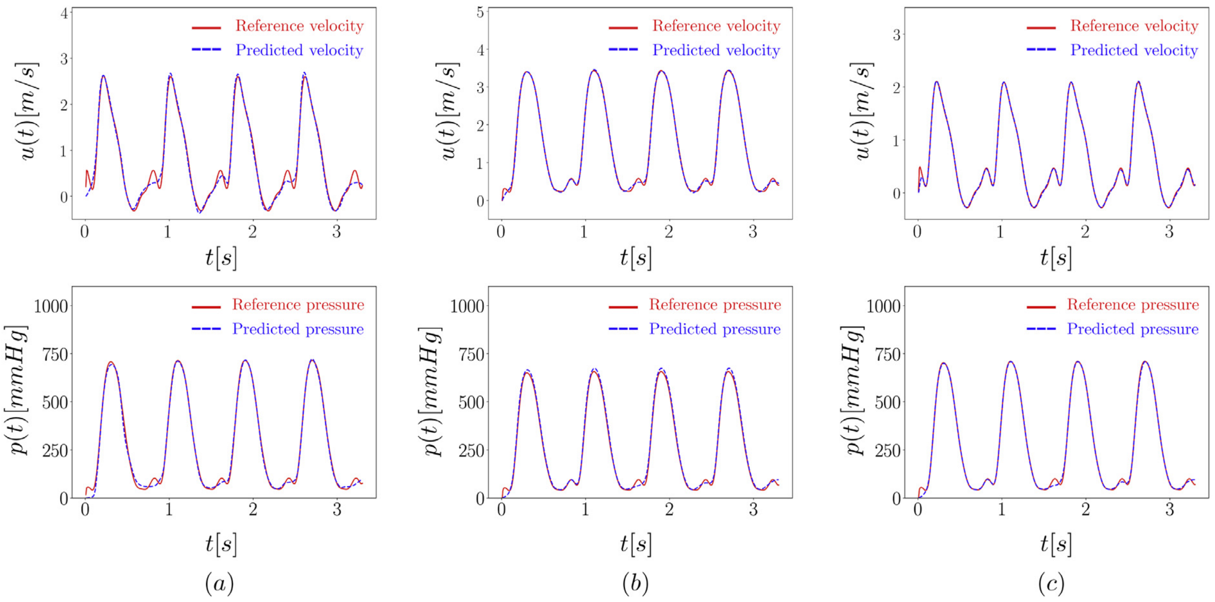

Fig. 6. *Flow through a prototype Y-shaped bifurcation:* Comparison of the physics-informed neural network model predictions and the discontinuous Galerkin reference solution. The numbering of the vessels starts from left to right, with the far left figure corresponding to the father (vessel #1) and the middle and right one to the daughter (vessels #2 and #3, respectively), as shown in Fig. 3. The top row represents the predicted velocity and the bottom row the predicted pressure for each vessel.

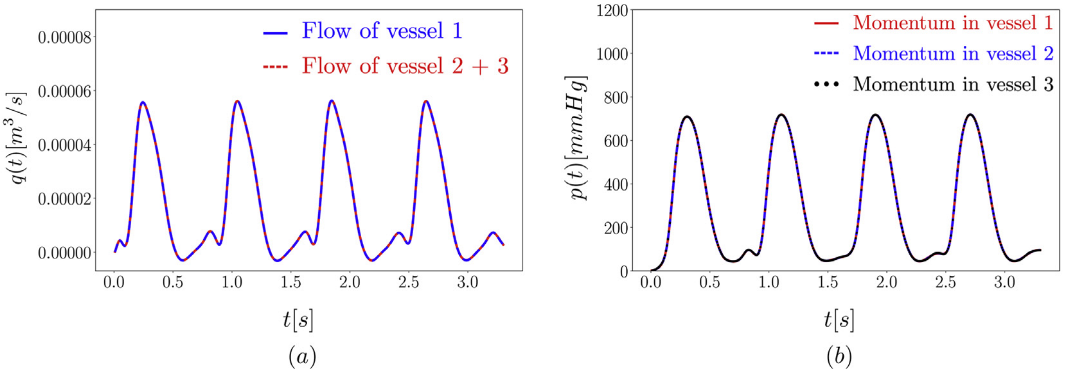

Fig. 7. *Flow through a prototype Y-shaped bifurcation:* Comparison of conserved quantities at the bifurcation point after training the proposed physics-informed neural networks. (a) Conservation of mass (see Eq. (16)). (b) Conservation of momentum (see Eqs. (17) and (18)) for the bifurcation point.

the figure that the mass in vessel #1 is equal to the total mass in vessel #2 and vessel #3. Moreover, the momentum at the father and daughter vessels at the bifurcation point is equal to each other. Thus, the conservation laws are well preserved in our model.

3.1.5. Systematic studies

We present a series of comprehensive systematic studies to test the sensitivity of the proposed methods and quantify their robustness with respect to different hyper-parameter settings together with a comparison between single and double precision arithmetic in order to justify our default precision choice.. All numerical experiments are performed for the canonical Y-shaped bifurcation topology presented in the previous section. To test the sensitivity of the proposed methods with respect to the initialization of the neural networks, we consider a data set comprising of $N_u = N_A = 413$, $N_r = 2000$, $N_b = 1024$, $N_{batch} = 1024$ training, collocation, interface points and batch size,

Table 3
Systematic study on the neural network initialization for single precision arithmetic.

Relative $\mathcal{L}_2$ error									
3.42e−02	4.99e−02	5.65e−02	3.92e−02	1.07e−01	5.52e−02	5.34e−02	3.54e−02	6.41e−02	4.57e−02
4.62e−02	4.04e−02	2.94e−02	3.78e−02	3.48e−02	1.32e−01	6.01e−02	3.00e−02	4.78e−02	4.63e−02
3.68e−02	4.48e−02	4.37e−02	5.81e−02	1.23e−01	5.80e−02	4.06e−02	2.33e−02	3.02e−02	3.96e−02
5.70e−02	3.61e−02	3.85e−02	3.83e−02	4.82e−02	4.85e−02	5.46e−02	6.01e−02	3.84e−02	7.43e−02
5.82e−02	5.09e−02	4.13e−02	6.66e−02	1.31e−01	7.31e−02	6.09e−02	5.59e−02	3.50e−02	3.47e−02

Relative $\mathbb{L}_2$ prediction error for different neural network initialization using different randomized seeds. The prediction errors are obtained by comparing the predicted value of pressure with the reference one in parent vessel #1 at point $x = 100\,\text{mm}$ for 2000 randomly selected temporal values.

Table 4
Systematic study on the neural network initialization for double precision arithmetic.

Relative $\mathcal{L}_2$ error									
4.02e−02	1.43e−01	2.21e−02	3.24e−02	3.13e−02	4.77e−02	2.57e−02	4.87e−02	5.66e−02	2.23e−02
2.81e−02	3.19e−02	6.22e−02	3.82e−02	3.31e−02	4.87e−02	3.13e−02	3.82e−02	4.62e−02	2.85e−02
3.37e−02	3.37e−02	3.26e−02	3.32e−02	3.47e−02	3.62e−02	3.05e−02	3.49e−02	3.40e−02	2.90e−02
3.20e−02	3.26e−02	3.58e−02	2.89e−02	5.01e−02	3.99e−02	3.54e−02	6.88e−02	3.35e−02	5.24e−02
2.84e−02	7.04e−02	6.08e−02	6.30e−02	9.21e−02	5.36e−02	5.43e−02	3.43e−02	3.85e−02	3.11e−02

Relative $\mathbb{L}_2$ prediction error for different neural network initialization using different randomized seeds. The prediction errors are obtained by comparing the predicted value of pressure with the reference one in parent vessel #1 at point $x = 100\,\text{mm}$ for 2000 randomly selected temporal values.

respectively, and fix the architecture for our neural networks to consist of seven hidden layers containing hundred neurons each followed by a hyperbolic tangent activation function. By doing so, we create an ensemble of fifty cases all starting from a Xavier initialization [49] for all network weights (with a randomized seed), and a zero initialization for all bias parameters. In Tables 3 and 4 we report the relative error between the predicted mean solution and the known Discontinuous Galerkin solution [45] for all 50 randomized trials at point $x = 100$ mm in vessel #1 for 2000 randomly selected temporal values for the single and double precision arithmetic case, respectively. Evidently, our results are robust with respect to the neural network initialization as in all cases the stochastic gradient descent [47] training procedure converged to solutions that are close to each other. We summarize the result by reporting the mean and the standard deviation of the relative $\mathbb{L}_2$ error defined as:

$$\hat{\mathcal{L}}_2 = \frac{\sqrt{\sum_{i=1}^{N_f}(p^i_{prediction} - p^i_{measurements})}}{\sqrt{\sum_{i=1}^{N_f}(p^i_{measurements})}}, \tag{21}$$

for single:

$$\hat{\mathcal{L}}_2 \in [\mu_L - \sigma_L, \mu_L + \sigma_L] = [5.29 \times 10^{-2} - 2.38 \times 10^{-2}, 5.29 \times 10^{-2} + 2.38 \times 10^{-2}].$$

and double precision arithmetic:

$$\hat{\mathcal{L}}_2 \in [\mu_L - \sigma_L, \mu_L + \sigma_L] = [4.25 \times 10^{-2} - 1.99 \times 10^{-2}, 4.25 \times 10^{-2} + 1.99 \times 10^{-2}].$$

It is evident that double precision leads to more accurate results. However, the computational time required for the analysis is significantly longer (2 h and 30 min instead of 1 h and 20 min, approximately). Granted that the single precision provides predictions of adequate accuracy, we choose to perform the calculations in single precision to shorten the wall-clock time of the analysis. A potential middle ground to be explored in future work aiming to balance accuracy and computational cost is to employ mixed precision arithmetic: store the network parameters and evaluate its forward pass in single precision, but evaluate the loss functions in double precision.

Subsequently, we test the sensitivity of our model with respect to the architecture of the neural networks. In this case, we fix the number of noise-free training data as $N_u = N_A = 413$, $N_r = 2000$ and $N_b = 1024$ for each vessel. In all cases, we used a hyperbolic tangent activation function. We initialize our neural network weights using Xavier initialization [49]. In Table 5 we report the relative $\mathbb{L}_2$ prediction error for different fully connected neural network

Table 5
Systematic study on the neural network architecture.

N_g	N_n			
	20	50	100	200
1	6.14e−01	8.41e−01	7.04e−01	1.07e+00
3	4.09e−01	3.08e−01	3.68e−01	2.39e−01
5	5.18e−01	2.35e−01	1.03e−01	2.49e−01
7	4.42e−01	3.88e−02	3.73e−02	3.56e−02

Relative $\mathbb{L}_2$ prediction error for different fully connected neural network architectures with different number of hidden layers N_g, and neurons N_n in each layer. The prediction errors are obtained by comparing the predicted value of pressure with the reference one in parent vessel #1 at point $x = 100\,\text{mm}$ for 2000 randomly selected temporal values.

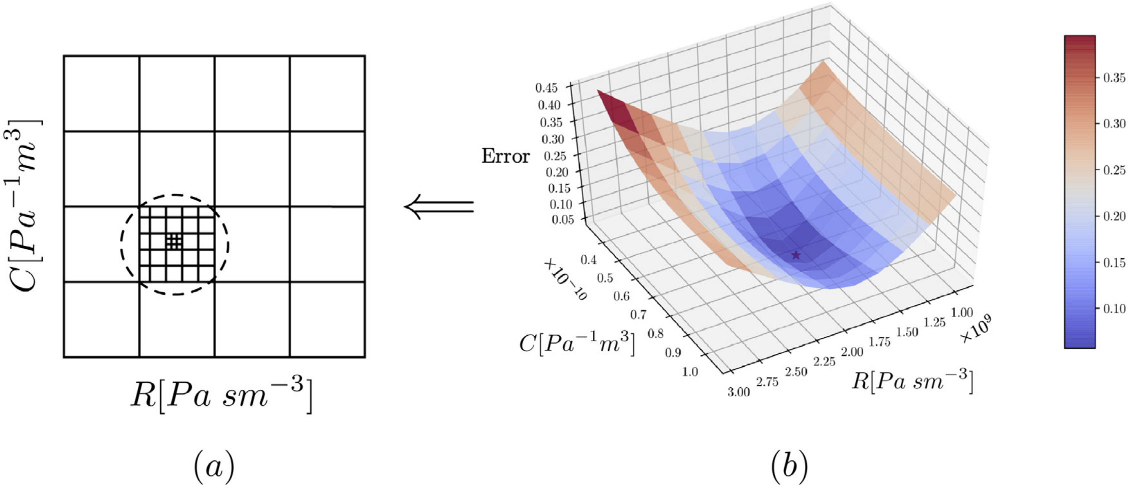

Fig. 8. *RCR identification by adaptive grid search for a Y-shaped bifurcation:* (a) A sketch that illustrates the adaptive grid search. (b) The surface illustrating the values of the loss with respect to parameters R and C. The red star denotes the point in the parameter space where the loss function achieves the minimum value.

architectures (i.e. different number of layers N_g and number of nodes N_n in each layer). The general trend suggests that as the neural network capacity is increased, we obtain more accurate predictions, which is indicating that our physics-informed constraint on the PDEs residual can effectively regularize the training process and safe-guard against over-fitting. For the rest of the paper we will utilize an architecture defined by seven layers and a hundred neuron per layer for its good performance in balancing the accuracy and computational intensity.

3.1.6. Windkessel parameter identification by adaptive grid search

As described in Section 2.8, our method provides an effective mechanism for calibrating Windkessel model parameters as a simple post-processing step. Specifically, using the predicted flow and pressure at the outflow points, we start by defining a coarse mesh in the space of Windkessel model parameters, and use an ODE solver with inputs R, C and the inferred flow to calculate the pressure wave form. After we calculate the pressure waveform for every grid point, we find the one that produces the smallest relative error when compared with the predicted one and refine the mesh near its vicinity. By repeating this procedure for a number of times, in this case five, we acquire a set of R and C that produce the minimum relative error when compared with the target one. In Fig. 8 we present the relative error as a function of the parameters R and C for the case of vessel #3 as a visual example of how the method of parameters identification works. For details about the predicted R, C and the corresponding relative loss, Eq. (20), we refer to Table 6.

Table 6
Windkessel parameter identification for a Y-shaped bifurcation.

# vessel	Param		
	R_{Pred}	C_{Pred}	Relative error of P waveforms
#2	7.58e+08 $Pasm^{-3}$	4.89e−11 $Pa^{-1}m^3$	3.36e−02
#3	1.84e+09 $Pasm^{-3}$	7.36e−11 $Pa^{-1}m^3$	3.67e−02

Resistance and compliance discovered by adaptive grid search for the three vessels, one bifurcation case and the $\mathcal{L}_2$ relative error between the pressure waveforms predicted by the physics-informed neural networks and the one reconstructed using Eq. (19) with the identified R and C parameters.

Table 7
Flow through an idealized pelvic arterial network.

Arterial segment	Length (cm)	Peripheral resistance (10^{10} Pa s m^{-3})	Peripheral compliance (10^{-10} m^3Pa^{-1})	β (Pa/m)	Equilibrium cross-sectional area (m^2)
1	1.0682	–	–	2.65e+07	2.14e−05
2	6.66638	–	–	2.60e+07	2.21e−05
3	6.99352	–	–	2.63e+07	2.17e−05
4	14.7735	0.3133	16.62	2.82e+07	1.97e−05
5	14.9503	0.1654	31.49	2.71e+07	2.08e−05
6	13.6421	0.1682	30.96	2.67e+07	2.12e−05
7	13.4384	0.2086	2.092	2.87e+07	1.92e−05

Physiological data used as simulation parameters for the discontinuous Galerkin method.

3.2. *Flow through an idealized pelvic arterial network*

3.2.1. *Network topology and observed measurements*

Our goal here is to demonstrate the effectiveness of the proposed methodology for a slightly more complicated problem, where the available training data lies inside the arterial domain, i.e., no available data is assumed at the inflow or the outflow boundaries. This illustrates a favorable property of our approach versus traditional simulation tools that necessitate the precise prescription of boundary conditions, especially for cases in which measurements are not accessible on the boundaries. To this end, we consider a network consisting of seven arteries with three bifurcations, where every single artery splits into two. This topology resembles an idealized female pelvic geometry, starting from the descending aorta, splitting down to the uterine arteries. We set the vessel lengths to be $L_1 = 10.68$ mm, $L_2 = 66.66$ mm, $L_3 = 69.94$ mm, $L_4 = 147.74$ mm, $L_5 = 149.50$ mm, $L_6 = 136.42$ mm and $L_7 = 134.38$ mm, as shown in Fig. 4. Each vessel is considered to have its own local coordinate system defined in the interval $x \in [0, L^j]$, where L^j is the length of the respective vessel. The arterial topology includes three bifurcations at the end points of domains #1, #2 and #3, as depicted in Fig. 4. For this geometry we assume we obtained measurements for the cross-sectional area and velocity over time at the middle points of domains #1, #4, #5, #6 and #7 and we want to predict the pressure at the boundary points. No information is provided for domains #2 and #3, and no information about the pressure in any vessel. In order to show that in this case the model can also propagate the information via the boundaries, we aim to infer the pressure at two points in domains #2 and #3 namely $x_2 = 3$ mm and $x_3 = 6$ mm.

3.2.2. *Generation of synthetic training and validation data*

In this example we generated a set of synthetic data for cross-sectional area and velocity using an in-house Discontinuous Galerkin solver [23,30,61]. Out of the resulting waveforms, we chose 3 steady state cycles as training data. For the DG simulation, we choose the blood density to be equal to 1060 Kg/m^3 and the viscosity 3.5 mPa s [62]. Moreover we provide the DG solver with the Windkessel and structural parameters shown in Table 7.

3.2.3. *Neural network approximation set-up*

We parametrize the solution of each artery by a neural network, seven in total, consisting of seven hidden layer and one hundred neurons followed by a hyperbolic tangent activation function, each. The neural network takes the

Table 8
Windkessel parameter identification for an idealized pelvic arterial network.

# vessel	Param		
	R_{Pred}	C_{Pred}	Relative error of P waveforms
#4	1.36e+09 Pasm^{-3}	3.78e−09 Pa^{-1}m^3	7.92e−03
#5	1.64e+09 Pasm^{-3}	2.22e−10 Pa^{-1}m^3	3.29e−02
#6	1.22e+09 Pasm^{-3}	2.96e−09 Pa^{-1}m^3	8.26e−03
#7	2.69e+09 Pasm^{-3}	1.84e−09 Pa^{-1}m^3	2.56e−02

Resistance and compliance discovered by the adaptive grid search for the seven vessels, three bifurcations case and the $\mathcal{L}_2$ relative error between the pressure waveforms of the ordinary differential equation solver using the identified parameters and the model prediction.

scaled inputs $[\hat{x}, \hat{t}]$ and predicts the non-dimensional outputs $[\hat{A}, \hat{u}, \hat{p}]$ for each domain. For domains #2, #3 we provide only the initial conditions, and no other information on A u and p, and train the model to be able to predict these values inside these domains by employing information propagated by the conservation laws on the boundaries. We are randomly, using Latin-hypercube sampling [42], select $N_r = 2000$ collocation points in each domain and have $N_A = N_u = 874$ measurements of u and A, sampled at the middle points of domains #1, #4, #5, #6 and #7. For domains #2 and #3 we provide $[\hat{x}_{init}, \hat{t}_{init}]$ as inputs, where the $init$ subscript denotes the coordinates and the time of the initial conditions. In this case $\hat{t}_{init}$ is a constant array having zero values and $\hat{x}_{init}$ a constant array comprised of values of equispaced points inside the domain. We choose $N_{batch} = 1024$ randomly selected points at each training step and the same number of interface points $N_b = 1024$ for imposing the continuity conditions. The model is trained employing the Adam optimizer [47] with $\eta = 10^{-3}$ learning rate for the first 280,000 iterations and $\eta = 10^{-4}$ for the consequent 40,000. We observe from the results presented in Fig. 9 that there exists some discrepancy between the predicted values for vessels #2 and #3 for which no data is provided.

3.2.4. Numerical comparison and Windkessel parameter identification

In this case, we aim to calibrate a three-element Windkessel model parameter by employing our algorithm to predict the flow at the outflows of the arterial network (red points in Fig. 4) and then utilize these values, as explained in Section 2.8, to solve Eq. (19). We start by creating a coarse mesh for R and C and solve the ODE on the grid points using the predicted flow. As long as, we calculate the pressure for all grid points, we choose the one for which the relative $\mathcal{L}_2$ error between the ODE solution and the model prediction exhibits the smallest value. Then, we generate an identical mesh around its vicinity. As soon as we calculate the waveforms and their respective relative error for the new grid points, we refine the mesh again. We repeat this process for a number of iterations, five in our case, to acquire the optimal values of resistance and compliance. For details on the predicted R, C and the corresponding relative loss we refer the reader to Table 8. In Fig. 9 the numerical results of our method are presented, in comparison to the solution produced by the discontinuous Galerkin method and the discontinuous Galerkin method using the discovered Windkessel parameters.

3.3. Flow through the aorta/carotid bifurcation of a healthy human subject

For the last example we will test the effectiveness of the proposed method on a realistic clinical case involving measurements acquired in the aorta/carotid bifurcation of a healthy volunteer.

3.3.1. Geometry and problem setup

In this case we consider an arterial geometry consisting of 4 vessels containing 1 bifurcation. An MRI image of this geometry is illustrated in Fig. 10. By utilizing medical image processing and blood flow velocity measuring techniques (see Section 3.3.2), we acquire measurements of area and velocity at the Aorta #1, Aorta #2, Aorta #3, Aorta #4 and carotid points. Our goal is to leverage the data at the carotid, Aorta #1 and Aorta #4 points to predict the area, velocity and pressure anywhere inside the arterial network. More precisely, we will test the predictions of our model against the measurements at Aorta #3. Furthermore, we would like to utilize the predicted pressure at the carotid point and Aorta #4 to discover R and C parameters, as discussed in Section 2.8. For vessel #3 we provide the network with just the initial conditions of velocity and area. We adopt a 4 vessels geometry instead of 3, as

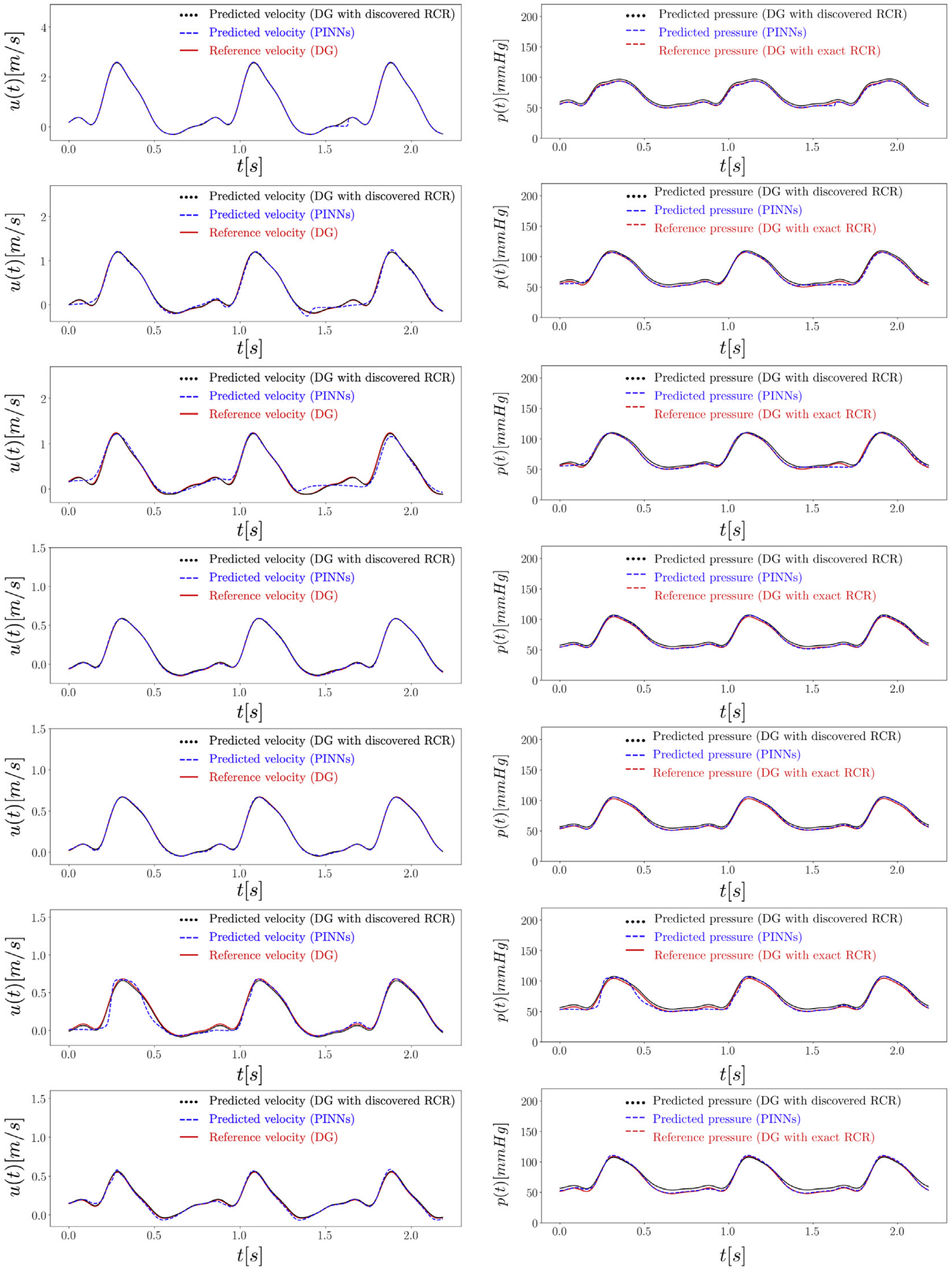

Fig. 9. *Numerical comparison for an idealized pelvic arterial network:* On the left the comparison between the predicted velocity and on the right the predicted pressure of the three methods is presented. The comparison is between our model (*blue*), the reference discontinuous Galerkin solution (*red*) and the discontinuous Galerkin solution using the discovered Windkessel parameters (*black*). On the right, the same case is presented. The vessels are numbered form top to bottom, starting with #1 at the top to #7 at the bottom. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

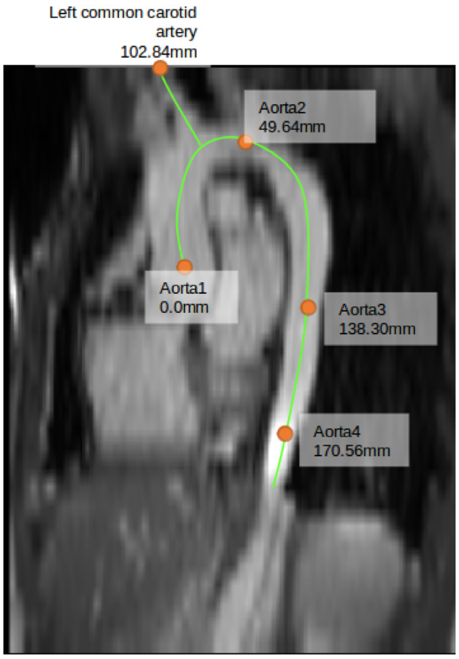

Fig. 10. *Flow through the aorta/carotid bifurcation of a healthy human subject:* Positions of acquired 4D flow MRI measurements in the aorta/carotid bifurcation of a healthy volunteer. Path length measurements are presented in reference to "Aorta1". Measurements collected at locations ["Aorta1", "Aorta4", "Left common carotid artery"] were used to train the physics-informed neural network model, while measurements at locations ["Aorta2", "Aorta3"] were used to validate its predictions.

the equilibrium cross-sectional area is not constant at the vessel ends. Thus, by considering 4 vessels and A_0 to be a linear function, we increase the accuracy of the model. For this case, we choose the neural network architecture and parameters to be the same as in Section 3.1.

3.3.2. Clinical data acquisition

The measurements used in this section were based on acquired MR images from a healthy female volunteer (age = 27 years, weight = 51 kg, height = 160 cm) using a 1.5T Avanto scanner (Siemens Healthcare, Erlangen, Germany). The patient was lying head-first, supine on the table. For the 3-lead electrocardiogram (ECG) monitoring, a neck coil, two body array coils on the chest and abdomen, and a spine array coil were used. The MRI protocol consisted of a Balanced Steady-State Free Precession (bSSFP) localizer of the neck through abdomen, followed by 2D Cine images and prospectively ECG-gated 2D Phase contrast images prescribed at four locations along the aorta and one location in the left common carotid artery, as shown in Fig. 10. Imaging parameters are provided in Table 9. The vessel area at each location was quantified by segmenting the 2D Cine images [15]. The velocity at each location was quantified by manual segmentation of the phase difference images (ImageJ 1.48 v;) 1.48v;imagej.nih.gov/ij, extracting the mean intensity (mi) inside the contour at each time frame, and computing the following equation: u = (2048-mi)/2048*venc, where venc = velocity encoding parameter. The venc values for aorta1, 2, 3, 4, and the carotid artery were 200, 150, 150, 150, and 150 cm/s, respectively. The arc lengths between consecutive planes were computed by segmentation of the bSSFP localizer images. The centerline extraction, and path length parameterization was acquired by using Seg3D 2.3.0 http://sci.utah.edu/cibc-software/seg3d.html [16] and VMTK [14]. Finally, we utilized a periodic kernel Gaussian Process regression scheme [63] to smooth out the input waveforms before we provide them as training data to the physics-informed neural networks. We chose to employ a periodic kernel Gaussian process smoothing over Fourier smoothing for the following reasons. First, Gaussian process regression naturally returns uncertainty estimates which can be used to assess the goodness of fit and could be subsequently used for uncertainty propagation tasks. Second, the wave-forms we obtain from the clinic may not necessarily be on a regular grid (i.e., the time intervals might not be equi-spaced), and additional interpolation is

Table 9
Flow through the aorta/carotid bifurcation of a healthy human subject.

	2D Cine	2D Phase Contrast
Flip angle (degrees)	55	25
Repetition time/echo time (ms)	51.74/1.7	10.35/6.92
Bandwidth (Hz/pixel)	765	390
Voxel size (mm3)	0.65 x 0.65 x 8	0.49 x 0.49 x 5
Temporal resolution (ms)	29.4–34.8	20.7
# Cardiac phases	30 (retrospectively-gated)	37–43 (prospectively-gated)

MRI parameters for 2D Cine images and 2D Phase contrast images used in the clinical data acquisition process.

Table 10
Flow through the aorta/carotid bifurcation of a healthy human subject.

# vessel	Param		
	R_{pred}	C_{pred}	Relative error of P waveforms
Carotid	2.09e+09 $Pasm^{-3}$	2.76e−10 $Pa^{-1}m^3$	3.98e−02
Aorta #4	1.48e+08 $Pasm^{-3}$	9.23e−09 $Pa^{-1}m^3$	7.35e−02

Windkessel model parameters discovered by adaptive grid search, and the $\mathcal{L}_2$ relative error in the pressure wave predicted by the neural networks versus the wave computed by evaluating the 3-element Windkessel model using the identified parameters.

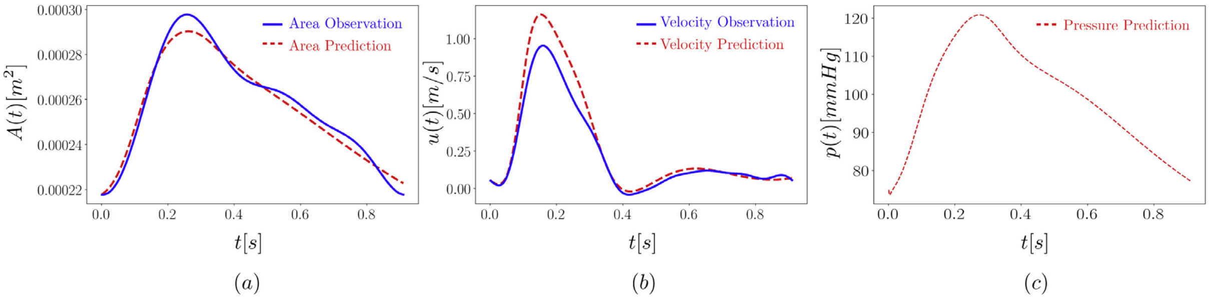

Fig. 11. *Flow through the aorta/carotid bifurcation of a healthy human subject:* (a) Comparison of the neural network model predictions against the clinical cross-sectional area measurements at the test point Aorta #3. (b) Comparison of neural network model predictions against the clinical cross-sectional velocity measurements at the test point Aorta #3. (c) Predicted pressure wave at the test point Aorta #3. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

needed before Fourier smoothing can be concisely applied. Last, in the Fourier case we have to choose the number of modes in such way that the trade-off between approximation quality and noise reduction is optimal. In Gaussian process regression this trade-off between data fit and model complexity is automatically balanced by maximizing the marginal likelihood of the model.

3.3.3. Numerical results

For this case we will compare the clinically measured data and the model prediction for the Aorta #3 point. The numerical results are presented in Fig. 11. Comparing the clinical measurements of area and velocity at Aorta #3 to our model predictions, we show that there is good agreement. Further more, the predicted pressure at this point is within a range consistent with the values reported in literature for a healthy patient (i.e., ranging between 80–120 mmHg). We present in Fig. 12 each component of the loss function, Eq. (9), as a function of the number of training iterations, to exhibit the training efficiency of the method.

As discussed in Section 2.8, we can also calibrate the R and C parameters of three-element Windkessel models for each of the outlets using the predicted pressure and flow rate at the carotid and Aorta #4 points. For more details we refer the reader to Table 10.

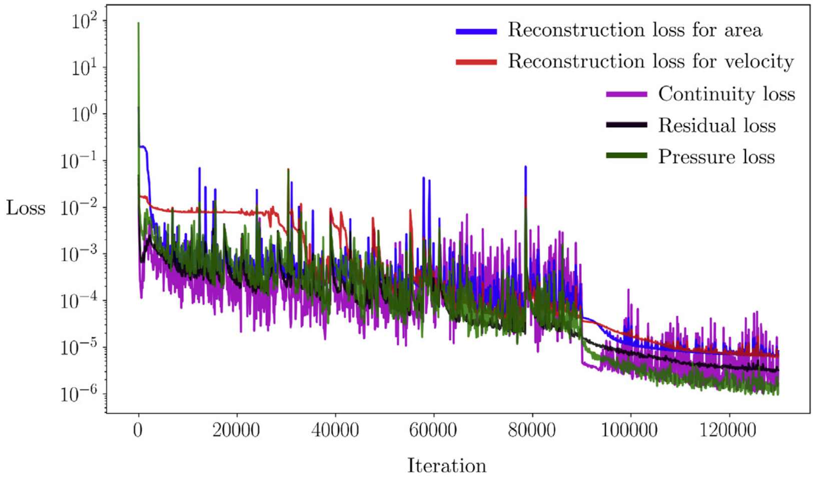

Fig. 12. *Flow through the aorta/carotid bifurcation of a healthy human subject:* Values of the loss functions versus number of stochastic gradient descent iterations during the training of the physics-informed neural networks. The magenta colored line denotes the continuity loss (see Eqs. (16)–(18)). The blue colored line denotes the reconstruction loss of the area measurements. The red colored line denotes the reconstruction loss of velocity measurements. The black colored line denotes the residual loss (see Eq. (15)). The green colored line denotes the pressure loss, meaning how close is the pressure predicted by the network with the one produced by inserting the predicted area to the pressure–area relation (see Eq. (2)).

Table 11
Flow through the aorta/carotid bifurcation of a healthy human subject.

Arterial segment	Length (cm)	Peripheral resistance (10^{10} Pa s m^{-3})	Peripheral compliance (10^{-10} m^3 Pa^{-1})	β (Pa/m)	Equilibrium cross-sectional area (m^2)
1	4.964	–	–	−6.47581e+06*x + 2.47267e+06	6.91e−04*x + 2.29e−04
2	5.32	0.2285	2.759	1.37380e+08*x + 2.15121e+06	−4.46e−03*x + 2.64e−04
3	8.866	–	–	7.32369e+06*x + 2.15121e+06	−5.18e−04*x + 2.64e−04
4	3.2259	0.01667	92.32	−8.42729e+06*x + 2.80053e+06	7.26e−04*x + 2.18e−04

Physiological data used as simulation parameters for the Discontinuous Galerkin method for the aorta/carotid bifurcation case. Based on the 4D flow MRI measurements, we have assumed a linear tapering across the vessels' length (i.e. the equilibrium cross-sectional area and the parameter β are linear functions of the local spatial coordinate of each artery).

3.3.4. Discontinuous Galerkin simulation using the identified Windkessel model parameters

Using the discovered three-element Windkessel model parameters we can employ a conventional Discontinuous Galerkin solver to infer the velocity within the arterial network, and compare these with results against the reference measurements and the neural network model predictions. For the DG simulation, we choose the blood density to be equal to 1060 Kg/m^3 and the viscosity 3.5 mPa s [62]. Moreover, we provide the DG solver with the Windkessel and structural parameters introduced in Table 11. The results are presented in Fig. 13.

In Section 2.1 we argued that the presented one dimensional reduced order model [23] constitutes an accurate approximation of the underlying fluid dynamics, but here we do actually observe some discrepancy between the model predictions and the clinically acquired data, see Fig. 13 for Aorta #4 and the carotid outlets. The accuracy of the one-dimensional model has been previously validated by Reymond et al. for a both nominal as well as patient-specific geometries [6,35]. In both cases the authors considered arterial trees consisting of many systemic arteries including the Left Common Carotid Artery and the Thoracic Aorta. In the case of patient specific validation, they employed an average of 4 measurements of the local diameter at each cross-section and also averaged over 5–15 cardiac cycles to obtain a smoothed flow waveform. Moreover, for modeling the pulse wave propagation they utilized a form of incompressible Navier–Stokes equations that takes into consideration the wall viscous effects [6], as well as a non-linear viscoelastic pressure relation for which they tuned the viscoelastic parameters based on values appearing in the literature. Even in this more detailed case, where more complex models that can better capture

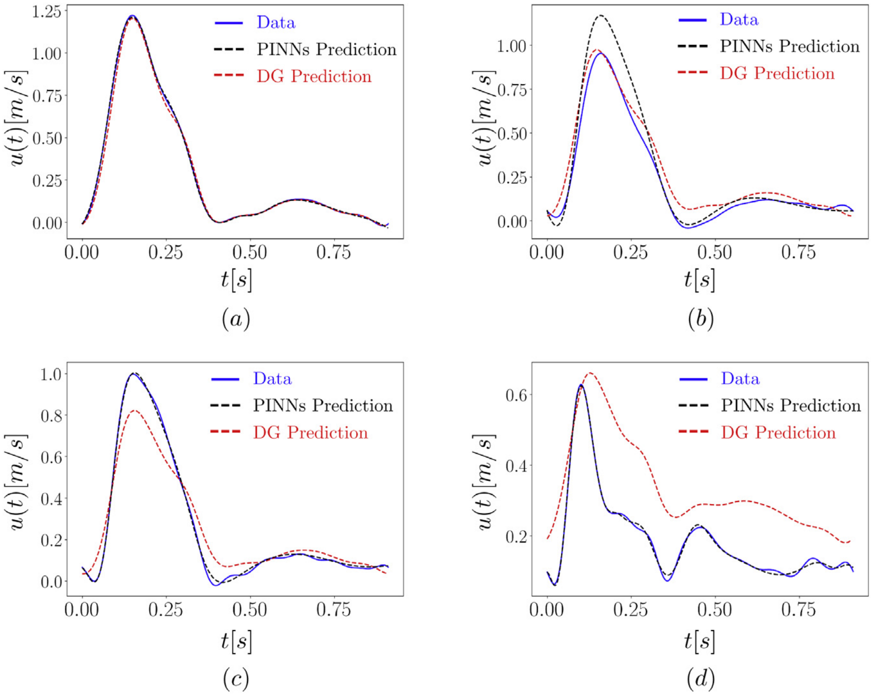

Fig. 13. *Flow through the aorta/carotid bifurcation of a healthy human subject:* Comparison of the clinically acquired waveforms of blood velocity versus the predictions of the proposed physics-informed neural networks, and of a conventional Discontinuous Galerkin solver with a three-element Windkessel model outflow condition using the R and C parameters identified by post processing the neural network outputs. (a) Aorta #1. (b) Aorta #3. (c) Aorta #4. (d) Left Common Carotid.

the complexity of the real world are utilized, there existed some discrepancy between the predicted values and the measured ones [6,35]. In our case, we do not pre-process the data in such a detailed manner, nor we employ such an expressive model that requires a large amount of information related to geometry, elastic properties etc. It can be seen from Fig. 13 that the Discontinuous Galerkin simulation that utilizes the Windkessel parameters discovered by the procedure described in Section 2.8, can capture the magnitude and the wave peak timing in a favorable manner. For the case of the Left Common Carotid Artery we detect a larger amount of discrepancy and a time shift, but this may be attributed to back-propagating elastic waves that the model in this form cannot capture. Overall, considering the simplicity of the one-dimensional pulsatile flow model and the amount of provided information, it still produces favorable results. The prediction of the neural network on the other hand is very accurate at the points that measurements are provided, but slightly over-estimates the velocity at the point that the model is not trained. This discrepancy can be attributed to the simplicity of the underlying physics-informed constraints, as well as the noise corruption in the velocity and cross-sectional area measurements used to train the neural network model. Overall, for the limited amount of measurements and the noise level they contain, the performance of the deep learning model still has margins for improvement.

4. Discussion

Advances in physics-informed machine learning provide the connecting link for integrating theoretical models and real-world data. This work is just one example of this general paradigm applied to the modeling and simulation of cardiovascular flows. Here, we have demonstrated how one-dimensional models of blood flow model can by seamlessly synthesized with clinical data to construct deep neural networks that can predict quantities which cannot be reliably measured in a non-invasive manner (e.g., blood pressure), by complementing the governing laws of fluid flow in compliant arteries with scattered measurements obtained by medical imaging techniques. To facilitate the efficient training of such physics-informed networks, we put forth proper non-dimensionalization and normalization techniques, as well as formulate a composite training objective that enables the consistent propagation of flow information across a network of systemic arteries. The proposed methodology yielded favorable results across a collection of synthetic and realistic examples, and showed good agreement with state-of-the-art numerical solvers using Discontinuous Galerkin discretizations. Moreover, we showed how a simple evaluation of the trained neural networks can provide a low-cost post-processing procedure for calibrating Windkessel model parameters for cases where traditional simulations are sought.

In comparison with conventional pure physics-based computational models, the advantages of our data-driven approach can be mainly attributed to the fact that they by-pass the tyranny of mesh generation, and they do not require the precise prescription of initial or boundary conditions, or even constitutive laws [31,40]. One can view the proposed techniques as a tool for PDE-constrained filtering of scattered noisy data in which the underlying physics can be only partially known, as opposed to interpreting these algorithms as numerical PDE solvers in the classical sense. We argue that this can potentially bring additional flexibility and eventually reduce the time required to obtain reliable predictions in a realistic clinical setting.

Deploying the proposed tool in the clinic would necessitate two main directions for improving our present model. First, given measurements for a new patient we need to reduce the problem setup time (i.e., geometry and 4D flow MRI data extraction), as well as the training time of the physics-informed neural network, from several hours to a few minutes in order to make our predictions practically useful to a clinician. The latter can be potentially achieved by leveraging ideas from transfer learning, in which pre-trained models can be used as an initial starting point for training when data for a new patient arrives. In such cases, the neural networks need not be trained from scratch (i.e., from a random initialization), but only be fine-tuned to adapt to the new data. The second direction of improvement is related to enhancing the accuracy and the robustness of the neural network predictions, as well as to endowing those predictions will reliable uncertainty estimates. This would include designing more stable and robust neural network architectures, further understanding the approximation power of physics-informed neural networks, and adopting probabilistic formulations that can enable uncertainty quantification. In all cases, extensive validation studies will be required to assess the potential of physics-informed neural networks as a reliable clinical tool.

Although the direct incorporation of clinical measurements is one of the main strengths of the proposed methodology, but also one of its weaknesses. Such measurements have often very coarse resolution and may be heavily corrupted by noise. This setting introduces challenges in encouraging the neural networks to simultaneously fit the data and satisfy the underlying conservation laws, as these requirements may contradict each other in the presence of noise. Although, this did not pose a significant challenge in the reported clinical case it may become an issue for studying smaller arterial networks for which the resolution of 4D flow MRI techniques may not suffice to produce readings with low signal-to-noise ratio. Moreover, the use of a mean square error loss tacitly assumes a Gaussian likelihood model for the data corresponding to an isotropic Gaussian noise model which may not be adequate to reflect the noise statistics in the real data signal which are likely non-Gaussian and heteroscedastic. Therefore, a use of a more robust likelihood/loss function or a probabilistic formulation similar to the one recently put forth in [33] could be useful. An alternative approach would be implement a physics based de-noising algorithm similar to the one put forth by *Rudy et al.* [64] in order to de-noise the measured data such that they satisfy the Navier–Stokes equation in a more accurate manner. We could also implement a more detailed form of the Navier–Stokes equation, taking in to account more detailed effects that better reflect the underlying physics (e.g., non-Newtonian rheology, wall visco-elasticity, etc.). On the data front, we could get more samples from the same patient and more patients and calibrate the inputs to have smoother measurements.

Finally, we should emphasize the need for scalable and reliable methods for uncertainty quantification, especially for over-parametrized deep learning models. Clinical applications like the one studied in this work often involve scarce data and high-consequence decisions, therefore posing the need for robust prediction models. Specific to this work, the most straightforward way to quantify uncertainty in the model predictions is to consider ensemble averaging of bootstrapped neural networks [65,66]. Alternatively, Bayesian approaches based on Markov Chain Monte Carlo sampling [67] or generative models can be employed [68]. As examples, we refer the reader to the recent works of [33,69], where an example of employing probabilistic neural networks to perform uncertainty quantification for PDE-governed systems is provided. Despite some progress, these works should be considered as first steps in the direction of incorporating uncertainty quantification in deep learning. This is an important aspect in data-driven modeling pipelines (especially in absence of theoretical bounds and guarantees), and defines an open area for future research.

Acknowledgments

The authors would like to thank Veronica Aramendia-Vidaurreta and Brianna Moon for assisting in the acquisition of carotid/aorta MRI data. G.K., Y.Y. and P.P. would like to acknowledge support from the US Department of Energy under the Advanced Scientific Computing Research program (grant DE-SC0019116), the Defense Advanced Research Projects Agency under the Physics of Artificial Intelligence program (grant HR00111890034). E.H. received support from the Eunice Kennedy Shriver National Institute of Child Health and Human Development (U01-HD087180), National Institute of Biomedical Imaging and Bio-engineering (T32-EB009384) and National Science Foundation (DGE-1321851).

References

[1] G.A. Rose, H. Blackburn, R. Gillum, R. Prineas, et al., Cardiovascular survey methods, second ed., 56, WHO, Geneva, Switzerland, 1982.

[2] A.C. Kak, M. Slaney, G. Wang, Principles of computerized tomographic imaging, Med. Phys. 29 (1) (2002) 107.

[3] M. Haris, A. Singh, K. Cai, F. Kogan, J. McGarvey, C. DeBrosse, G.A. Zsido, W.R. Witschey, K. Koomalsingh, J.J. Pilla, et al., A technique for in vivo mapping of myocardial creatine kinase metabolism, Nature Med. 20 (2) (2014) 209.

[4] L. Formaggia, A. Quarteroni, A. Veneziani, Cardiovascular Mathematics: Modeling and simulation of the circulatory system, 1, Springer Science & Business Media, 2010.

[5] C.A. Figueroa, I.E. Vignon-Clementel, K.E. Jansen, T.J. Hughes, C.A. Taylor, A coupled momentum method for modeling blood flow in three-dimensional deformable arteries, Comput. Methods Appl. Mech. Engrg. 195 (41–43) (2006) 5685–5706.

[6] P. Reymond, F. Merenda, F. Perren, D. Rufenacht, N. Stergiopulos, Validation of a one-dimensional model of the systemic arterial tree, Amer. J. Physiol. Heart Circ. Physiol. 297 (1) (2009) H208–H222.

[7] L. Grinberg, E. Cheever, T. Anor, J.R. Madsen, G. Karniadakis, Modeling blood flow circulation in intracranial arterial networks: A comparative 3D/1D simulation study, Ann. Biomed. Eng. 39 (1) (2011) 297–309.

[8] A. Stefanovska, Physics of the human cardiovascular system, Contemp. Phys. 40 (1) (1999) 31–55.

[9] Y. Ma, J. Choi, A. Hourlier-Fargette, Y. Xue, H.U. Chung, J.Y. Lee, X. Wang, Z. Xie, D. Kang, H. Wang, et al., Relation between blood pressure and pulse wave velocity for human arteries, Proc. Natl. Acad. Sci. 115 (44) (2018) 11144–11149.

[10] M. O'Rourke, Mechanical principles in arterial disease, Hypertension 26 (1) (1995) 2–9.

[11] C. Chan, C. Poon, R.C. Wong, Y. Zhang, A hybrid body sensor network for continuous and long-term measurement of arterial blood pressure, in: 2007 4th IEEE/EMBS International Summer School and Symposium on Medical Devices and Biosensors, IEEE, 2007, pp. 121–123.

[12] R. Kett-White, P.J. Hutchinson, P.G. Al-Rawi, A.K. Gupta, J.D. Pickard, P.J. Kirkpatrick, Adverse cerebral events detected after subarachnoid hemorrhage using brain oxygen and microdialysis probes, Neurosurgery 50 (6) (2002) 1213–1222.

[13] W.A. Edelstein, J.M. Hutchison, G. Johnson, T. Redpath, Spin warp NMR imaging and applications to human whole-body imaging, Phys. Med. Biol. 25 (4) (1980) 751.

[14] L. Antiga, M. Piccinelli, L. Botti, B. Ene-Iordache, A. Remuzzi, D.A. Steinman, An image-based modeling framework for patient-specific computational hemodynamics, Med. Biol. Eng. Comput. 46 (11) (2008) 1097.

[15] P.A. Yushkevich, J. Piven, H.C. Hazlett, R.G. Smith, S. Ho, J.C. Gee, G. Gerig, User-guided 3D active contour segmentation of anatomical structures: Significantly improved efficiency and reliability, Neuroimage 31 (3) (2006) 1116–1128.

[16] CIBC, Seg3D: Volumetric Image Segmentation and Visualization, Scientific Computing and Imaging Institute (SCI), 2016, Download from: http://www.seg3d.org.

[17] R. Aaslid, T.-M. Markwalder, H. Nornes, Noninvasive transcranial doppler ultrasound recording of flow velocity in basal cerebral arteries, J. Neurosurg. 57 (6) (1982) 769–774.

[18] G.F. Mitchell, M.A. van Buchem, S. Sigurdsson, J.D. Gotal, M.K. Jonsdottir, Ó. Kjartansson, M. Garcia, T. Aspelund, T.B. Harris, V. Gudnason, et al., Arterial stiffness, pressure and flow pulsatility and brain structure and function: The Age, Gene/Environment Susceptibility–Reykjavik study, Brain 134 (11) (2011) 3398–3407.

[19] M. Markl, A. Frydrychowicz, S. Kozerke, M. Hope, O. Wieben, 4D flow MRI, J. Magn. Reson. Imaging 36 (5) (2012) 1015–1036.

[20] S. Plein, T.N. Bloomer, J.P. Ridgway, T.R. Jones, G.J. Bainbridge, M.U. Sivananthan, Steady-state free precession magnetic resonance imaging of the heart: Comparison with segmented k-space gradient-echo imaging, J. Magn. Reson. Imaging Official J. Int. Soc. Magn. Reson. Med. 14 (3) (2001) 230–236.

[21] A.T. Barker, R. Jalinous, I.L. Freeston, Non-invasive magnetic stimulation of human motor cortex, Lancet 325 (8437) (1985) 1106–1107.

[22] S. Urquiza, P. Blanco, M. Vénere, R. Feijóo, Multidimensional modelling for the carotid artery blood flow, Comput. Methods Appl. Mech. Engrg. 195 (33–36) (2006) 4002–4017.

[23] S. Sherwin, V. Franke, J. Peiró, K. Parker, One-dimensional modelling of a vascular network in space-time variables, J. Eng. Math. 47 (3–4) (2003) 217–250.

[24] K.S. Matthys, J. Alastruey, J. Peiró, A.W. Khir, P. Segers, P.R. Verdonck, K.H. Parker, S.J. Sherwin, Pulse wave propagation in a model human arterial network: Assessment of 1-D numerical simulations against in vitro measurements, J. Biomech. 40 (15) (2007) 3476–3486.

[25] J.A. Revie, D.J. Stevenson, J.G. Chase, C.E. Hann, B.C. Lambermont, A. Ghuysen, P. Kolh, G.M. Shaw, S. Heldmann, T. Desaive, Validation of subject-specific cardiovascular system models from porcine measurements, Comput. Methods Programs Biomed. 109 (2) (2013) 197–210.

[26] P. Chen, A. Quarteroni, G. Rozza, Simulation-based uncertainty quantification of human arterial network hemodynamics, Int. J. Numer. Methods Biomed. Eng. 29 (6) (2013) 698–721.

[27] N. Westerhof, J.-W. Lankhaar, B.E. Westerhof, The arterial Windkessel, Med. Biol. Eng. Comput. 47 (2) (2009) 131–141, URL https://doi.org/10.1007/s11517-008-0359-2.

[28] L. Grinberg, G.E. Karniadakis, Outflow boundary conditions for arterial networks with multiple outlets, Ann. Biomed. Eng. 36 (9) (2008) 1496–1514.

[29] M.S. Olufsen, Structured tree outflow condition for blood flow in larger systemic arteries, Amer. J. Physiol. Heart Circ. Physiol. 276 (1) (1999) H257–H268.

[30] P. Perdikaris, L. Grinberg, G.E. Karniadakis, An effective fractal-tree closure model for simulating blood flow in large arterial networks, Ann. Biomed. Eng. 43 (6) (2015) 1432–1442, URL https://doi.org/10.1007/s10439-014-1221-3.

[31] M. Raissi, P. Perdikaris, G. Karniadakis, Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations, J. Comput. Phys. 378 (2019) 686–707.

[32] M. Raissi, P. Perdikaris, G.E. Karniadakis, Machine learning of linear differential equations using gaussian processes, J. Comput. Phys. 348 (2017) 683–693.

[33] Y. Yang, P. Perdikaris, Adversarial uncertainty quantification in physics-informed neural networks, J. Comput. Phys. 394 (2019) 136–152.

[34] Y. Zhu, N. Zabaras, Bayesian deep convolutional encoder–decoder networks for surrogate modeling and uncertainty quantification, J. Comput. Phys. 366 (2018) 415–447.

[35] P. Reymond, Y. Bohraus, F. Perren, F. Lazeyras, N. Stergiopulos, Validation of a patient-specific one-dimensional model of the systemic arterial tree, Amer. J. Physiol. Heart Circ. Physiol. 301 (3) (2011) H1173–H1182.

[36] S. Sherwin, L. Formaggia, J. Peiro, V. Franke, Computational modelling of 1D blood flow with variable mechanical properties and its application to the simulation of wave propagation in the human arterial system, Internat. J. Numer. Methods Fluids 43 (6–7) (2003) 673–700.

[37] D. Lamponi, One dimensional and multiscale models for blood flow circulation, Tech. Rep., EPFL, 2004.

[38] M. Raissi, P. Perdikaris, G.E. Karniadakis, Physics informed deep learning (part I): Data-driven solutions of nonlinear partial differential equations, 2017, arXiv preprint arXiv:1711.10561.

[39] M. Raissi, P. Perdikaris, G.E. Karniadakis, Physics informed deep learning (part II): Data-driven discovery of nonlinear partial differential equations, 2017, arXiv preprint arXiv:1711.10566.

[40] A.M. Tartakovsky, C.O. Marrero, D. Tartakovsky, D. Barajas-Solano, Learning parameters and constitutive relationships with physics informed deep neural networks, 2018, arXiv preprint arXiv:1808.03398.

[41] N. Xiao, J. Alastruey, C. Alberto Figueroa, A systematic comparison between 1-D and 3-D hemodynamics in compliant arterial models, Int. J. Numer. Methods Biomed. Eng. 30 (2) (2014) 204–231.

[42] M. Stein, Large sample properties of simulations using latin hypercube sampling, Technometrics 29 (2) (1987) 143–151.

[43] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, A. Lerer, Automatic differentiation in pytorch, 2017.

[44] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, et al., Tensorflow: A system for large-scale machine learning, in: OSDI, Vol. 16, 2016, pp. 265–283.

[45] B. Cockburn, G.E. Karniadakis, C.-W. Shu, Discontinuous Galerkin methods: Theory, computation and applications, Vol. 11, Springer Science & Business Media, 2012.

[46] S. Ruder, An overview of gradient descent optimization algorithms, 2016, arXiv preprint arXiv:1609.04747.

[47] D.P. Kingma, J. Ba, Adam: A method for stochastic optimization, 2014, arXiv preprint arXiv:1412.6980.

[48] J. Duchi, E. Hazan, Y. Singer, Adaptive subgradient methods for online learning and stochastic optimization, J. Mach. Learn. Res. 12 (Jul) (2011) 2121–2159.

[49] X. Glorot, Y. Bengio, Understanding the difficulty of training deep feedforward neural networks, in: Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, 2010, pp. 249–256.

[50] J. Famiglietti, E. Wood, Multiscale modeling of spatially variable water and energy balance processes, Water Resour. Res. 30 (11) (1994) 3061–3078.

[51] Y.A. LeCun, L. Bottou, G.B. Orr, K.-R. Müller, Efficient backprop, in: Neural Networks: Tricks of the Trade, Springer, 2012, pp. 9–48.

[52] Z. Duanmu, W. Chen, H. Gao, X. Yang, X. Luo, N.A. Hill, A one-dimensional hemodynamic model of the coronary arterial tree, Front. Physiol. 10 (2019).

[53] C. Audebert, P. Bucur, E. Vibert, J.-F. Gerbeau, I. Vignon-Clementel, Closed-loop cardiovascular system model and partial hepatectomy simulation, 2015.

[54] J. Alastruey, S. Moore, K. Parker, T. David, J. Peiró, S. Sherwin, Reduced modelling of blood flow in the cerebral circulation: Coupling 1-D, 0-D and cerebral auto-regulation models, Int. J. Numer. Methods Fluids 56 (8) (2008) 1061–1067.

[55] M. Ismail, W.A. Wall, M.W. Gee, Adjoint-based inverse analysis of windkessel parameters for patient-specific vascular models, J. Comput. Phys. 244 (2013) 113–130.

[56] R.L. Spilker, C.A. Taylor, Tuning multidomain hemodynamic simulations to match physiological measurements, Ann. Biomed. Eng. 38 (8) (2010) 2635–2648, URL https://doi.org/10.1007/s10439-010-0011-9.

[57] R.L. Spilker, J.A. Feinstein, D.W. Parker, V.M. Reddy, C.A. Taylor, Morphometry-based impedance boundary conditions for patient-specific modeling of blood flow in pulmonary arteries, Ann. Biomed. Eng. 35 (4) (2007) 546–559, URL https://doi.org/10.1007/s10439-006-9240-3.

[58] Y.-C. Yu, J.R. Boston, M.A. Simaan, J.F. Antaki, Estimation of systemic vascular bed parameters for artificial heart control, IEEE Trans. Automat. Control 43 (6) (1998) 765–778.

[59] S. Pant, B. Fabrèges, J.-F. Gerbeau, I. Vignon-Clementel, A methodological paradigm for patient-specific multi-scale CFD simulations: From clinical measurements to parameter estimates for individual analysis, Int. J. Numer. Methods Biomed. Eng. 30 (12) (2014) 1614–1648.

[60] D.E. Schiavazzi, A. Baretta, G. Pennati, T.-Y. Hsia, A.L. Marsden, Patient-specific parameter estimation in single-ventricle lumped circulation models under uncertainty, Int. J. Numer. Methods Biomed. Eng. 33 (3) (2017) e02799.

[61] P. Perdikaris, G.E. Karniadakis, Fractional-order viscoelasticity in one-dimensional blood flow models, Ann. Biomed. Eng. 42 (5) (2014) 1012–1023, URL https://doi.org/10.1007/s10439-014-0970-3.

[62] F.E. Fossan, J. Mariscal-Harana, J. Alastruey, L.R. Hellevik, Optimization of topological complexity for one-dimensional arterial blood flow models, J. R. Soc. Interface 15 (149) (2018) 20180546.

[63] C.E. Rasmussen, in: O. Bousquet, U. von Luxburg, G. Rätsch (Eds.), Gaussian Processes in Machine Learning, ML Summer Schools 2003, Canberra, Australia, February 2 - 14, 2003, TÜBingen, Germany, August 4 - 16, 2003, Revised Lectures, in: Advanced Lectures on Machine Learning, Springer Berlin Heidelberg, Berlin, Heidelberg, 2004, pp. 63–71, URL https://doi.org/10.1007/978-3-540-28650-9_4.

[64] S.H. Rudy, J. Kutz, S.L. Brunton, Deep learning of dynamics and signal-noise decomposition with time-stepping constraints, 2018, arXiv preprint arXiv:1808.02578.

[65] B. Lakshminarayanan, A. Pritzel, C. Blundell, Simple and scalable predictive uncertainty estimation using deep ensembles, in: Advances in Neural Information Processing Systems, 2017, pp. 6402–6413.

[66] I. Osband, J. Aslanides, A. Cassirer, Randomized prior functions for deep reinforcement learning, in: Advances in Neural Information Processing Systems, 2018, pp. 8617–8629.

[67] A. Gelman, J.B. Carlin, H.S. Stern, D.B. Dunson, A. Vehtari, D.B. Rubin, Bayesian data analysis, Chapman and Hall/CRC, 2013.

[68] Y. Yang, P. Perdikaris, Physics-informed deep generative models, 2018, arXiv preprint arXiv:1812.03511.

[69] Y. Zhu, N. Zabaras, P.-S. Koutsourelakis, P. Perdikaris, Physics-constrained deep learning for high-dimensional surrogate modeling and uncertainty quantification without labeled data, 2019, arXiv preprint arXiv:1901.06314.