Computers in Biology and Medicine 203 (2026) 111472

Contents lists available at ScienceDirect

Computers in Biology and Medicine

journal homepage: www.elsevier.com/locate/compbiomed

Reconstructing in-vitro and in-vivo signals and parameters in networks of elastic vessels using physics-informed neural networks

J. Orera*, J. Mairal, L. Sánchez-Fuster, J. Murillo

Aragón Institute of Engineering Research, University of Zaragoza, C. de Mariano Esquillor Gómez, S/N, Zaragoza, 50018, Spain

HIGHLIGHTS

- We combine Physics-Informed Neural Networks with 0D models of arterial networks.
- 0D-PINNs infer Young's modulus and wall stiffness from experimental waveforms.
- Aortic flow is inferred from a single in-vivo carotid pressure measurement.
- 0D-PINNs are able to jointly estimate the parameters of the vascular beds.
- Despite scarce data, 0D-PINNs can still achieve accurate reconstructions.

ARTICLE INFO

Keywords:
Networks of elastic vessels
Elastic arteries
0D models
Physics-informed neural networks (PINNs)
Signals and parameters reconstruction
Inverse problems

ABSTRACT

The reconstruction of waveforms and hidden parameters is crucial for the physical modeling of steady and transient flows in networks of elastic vessels (arteries), where many mechanical properties are not directly measurable. This work investigates the potential of Physics-Informed Neural Networks (PINNs) to address the challenge of reconstructing pressure and flow signals and inferring parameters from experimental data. We incorporate the zero-dimensional (0D) system of coupled differential equations that describe flow in elastic vessels into the neural network, which we call 0D-PINN. We evaluate our methodology with several test cases representing different dynamical systems, including an experimental mock arterial network with 37 silicone vessels replicating the human arterial system, as well as a clinical case based on in-vivo MRI data from a healthy adult's thoracic aorta. It is shown that coupling 0D models with Physics-Informed Neural Networks (PINNs) enables the recovery of parameters and waveforms from experimental in-vitro or in-vivo data.

1. Introduction

The numerical simulation of vessel networks is characterized by the difficulty of simulating transient states, where the propagation of pressure and flow waveforms depends on the geometrical and mechanical properties of the vessels as well as the boundary conditions of the problem. The modeling of vessel networks in complex and realistic cases is useful in a wide range of scenarios: from the application to the modeling of pulse wave propagation in a model of a human arterial network [1] to the reproduction of in-vivo measurements [2]. In both cases, reproducing an experimental model with a numerical solver [3] requires knowledge of the network's geometrical and material properties, the test fluid's rheological data, and the inflow and outflow boundary conditions.

Scientific machine learning, defined as the use of AI techniques in combination with physical modeling in science-related problems, is a field of growing interest [4–6]. Traditional machine learning models are usually trained using large amounts of data in order to learn behaviors without explicitly being taught. Data-driven models have achieved remarkable success making highly accurate predictions in text prediction, language learning and image recognition.

In biomedical applications [2,7,8], data scarcity remains a significant challenge, particularly due to the high costs and technical difficulties of the measurements. In these data-limited environments, data-driven models often struggle to identify hidden patterns and make reliable predictions. As a result, traditional solvers are commonly relied upon, but they face the limitation of not being able to incorporate the limited data available. These traditional solvers make use of physical laws and

* Corresponding author.
Email address: jorera@unizar.es (J. Orera).

https://doi.org/10.1016/j.compbiomed.2026.111472
Received 21 July 2025; Received in revised form 26 November 2025; Accepted 10 January 2026
Available online 22 January 2026

numerical methods codified in algorithms to approximate physical phenomena [9,10]. When data is scarce or when physics is complex and non-linear, combining physics-based traditional solvers and data-driven approaches has shown significant advantages [5,11–13].

This new hybrid paradigm has been coined Physics-Informed Machine Learning (PIML). PIML can be used to uncover hidden physical relations by reproducing unknown conservation or constitutive laws [14,15], by calibrating parameters or boundary conditions of the system (inverse problems) [16,17] or by accelerating traditional numerical methods at the predictive stage [18,19]. Likewise, physical modeling can enhance machine learning techniques by increasing the amount of data available for training [20], or by adding extra constraints derived from known physical laws [21,22].

One of the most promising subfields of PIML is Physics-Informed Neural Networks (PINNs). PINNs employ soft constraints to combine data-driven learning with physical laws. PINNs include physical laws as learning biases, normally in the form of a system of differential equations. In the absence of *big data*, this allows the model to be trained with a much smaller volume of data (virtually only initial and/or boundary conditions), because physics guides the learning process. While this idea was first introduced more than two decades ago [23,24], it has been greatly developed in the last few years [5,25–27].

In particular, Physics-Informed Neural Networks (PINNs) have demonstrated notable success in solving PDEs and ODEs in high-dimensional settings. However, for lower-dimensional problems (3D, 2D, or 1D), they are generally less efficient and accurate than the advanced numerical methods employed in CFD solvers. Nevertheless, when it comes to inverse problems —where the goal is not purely computational efficiency or achieving real-time machine precision but rather to uncover hidden patterns from noisy or partial data in an offline fashion— PINNs have made significant strides. This is particularly true compared to most traditional numerical methods, which primarily operate as forward solvers and thus depend on prior assumptions regarding material properties, initial conditions, and boundary conditions. In the established framework, any situation in which some data are known while input signals or geometric/constitutive parameters remain unknown can be cast as an *inverse problem*. The unknowns are usually time–invariant parameters—e.g., vessel length, stiffness, etc, or localized time-dependent waveforms such as the inlet flow rate, in the case of cardiovascular modeling. The inverse algorithm therefore exploits complete or partial measurements acquired over one cycle to infer the hidden parameters or to reconstruct the missing waveforms at specific vessels or vascular beds within the network.

Following a zero-dimensional (0D) modeling perspective, the relevant mechanical parameters of the elastic vessels can be aggregated into *lumped* parameters, and the average evolution of physical variables can be reduced to a system of coupled Ordinary Differential Equations (ODEs), depending only on time. In general, these are called lumped parameters, and in the field of hemodynamics they are called Windkessel models. While their predictions are often less accurate than 1D or 3D approaches, they remain particularly useful for evaluating wave morphology, propagation patterns, and allow for coupling with extensive 3D and 1D models [28]. Their simplicity translates into a high computational efficiency while preserving overall accuracy [29–31]. A common issue with Windkessel models is that their parameters must be calibrated, as they are usually unknown. The influence of uncertainties in the calibration of external boundary conditions was tackled in [32], where it is shown that more compliant walls lead to increased accuracy in the parametric inference. In [33], the impact of arterial wall stiffness on several hemodynamic factors is analyzed, including the wall shear stress distribution. In [34], a non-iterative approach for calibrating the Windkessel parameters is studied, departing from a steady CFD simulation of the patient's hemodynamics. In [35], a Bayesian calibration of the vascular bed lumped parameters is performed, and the algorithm is trained on synthetic data resulting from a 3D simulation of the proposed thoracic aortic network. Drawbacks of the proposed methodology are the computational costs of the 3D simulation as well as the need for this data to initialize the calibration.

Inverse solvers such as PINNs are able to automatically infer these parameters to agree with both conservation laws and available data. In the particular field of fluid mechanics, PINNs have emerged as powerful forward solvers able to handle both sub- and supersonic scenarios and assimilate boundary data [36,37]. In the specific field of hemodynamics, other works such as [38], demonstrate the promising potential of PINNs to solve inverse problems related to the recovery of vascular bed parameters. In that study, predictions of flow and pressure were compared against those from a Reduced-Order Unscented Kalman Filter. However, as in [38], the algorithm was trained solely on artificially-generated in-silico data rather than in-vivo measurements. Moreover, the availability of abundant in-silico training data is an idealized condition that is rarely met in practice. In contrast, part of our work relies on in-vivo data as well.

In [39], the physics-constrained coupled neural differential equation (PCNDE) framework is employed as a forward solver and evaluated against a conventional finite element method (FEM)-based 1D solver. Temporal periodicity is enforced via hard constraints using Fourier series fitting. Although PCNDE performs well in forward modeling tasks, it has not yet been formulated or validated for inverse problems. RoePINNs [12] address this gap by introducing an advanced discretization strategy for fluxes between cells or collocation points in the domain. This enables both forward and inverse formulations and enhances interpretability, as the model is grounded in a detailed and physically consistent formulation of the governing differential equations.

This work showcases the capabilities of these algorithms to effectively reconstruct hidden non-measurable parameters in complex scenarios where highly nonlinear flow and pressure waveforms are involved. We also show the advantages of incorporating the 0D model equations into the PINN framework. We focus on forward and inverse problems in 0D representations of networks of elastic vessels as well as networks of elastic arteries coupled to vascular beds. This type of problem can be ill-posed: with the available data, the solution for at least some of the unknown variables of the problem is not guaranteed to exist or may not be unique. To reconstruct a solution that complies with available data (usually pressure and/or flow measurements) and with the physics of the network, we use PINNs in combination with a 0D model of the network (hereafter denoted 0D-PINNs), while enforcing additional hypotheses of different natures (periodicity of the predicted waveform). In fact, enforcing the periodicity of the predicted waveforms eliminates the need to provide initial conditions (unknown in real clinical scenarios) to the 0D-PINN, a requirement in previous works [40].

One of the main caveats of PINNs is the training cost associated with the minimization of a complex multi-objective loss function. The computational efficiency of PINNs can potentially be enhanced through techniques such as extreme learning machines (ELM) [41], which achieve ultra-fast training by assigning random weights to the hidden layers and only solving analytically for the output layer weights; randomized neural networks [42], which use random projections to accelerate the optimization process; Fourier feature embeddings, which map input data into a higher-dimensional space and provide faster convergence [43,44], and other strategies and models designed to accelerate neural network training [27], such as the Monte Carlo eXtreme Theory of Function Connections (MC X-TFC) [45].

The present work includes important advances over similar works in the field [40] when modeling vessel networks, where the 1D equations for mass and momentum were used to define the residual loss. In [40], an adaptive grid search post-processing of the outputs of the neural network was used to provide an estimation of the Windkessel model parameters. On the contrary, in the present work, Windkessel model parameters are calibrated by the 0D-PINNs. In fact, this is not an unnecessary or optional step when modeling pressure and flow waveforms, as otherwise the system remains undetermined [2]. This aspect was misinterpreted

in [40], where the model was trained solely on velocity and area data. As a result, the inferred pressure values exhibited non-physiological fluctuations (ranging from 100 to 800 mmHg) in in-silico Y-shaped bifurcation geometries. Furthermore, the method yielded inaccurate flow waveform estimations in a healthy human aorta–carotid bifurcation, even when using a Discontinuous Galerkin solver coupled with a three-element Windkessel outflow model, whose parameters were identified through post-processing of the neural network outputs. It is important to recognize that the wave propagation phenomena considered here inherently require the specification of external boundary conditions (e.g., Windkessel models) as well as appropriate initial conditions. Machine learning approaches designed for time-dependent predictions cannot disregard these fundamental mathematical constraints [9].

The structure of the work is as follows: first, a single junction of elastic vessels (one parent and two daughter vessels) is analyzed, in order to test the potential of the 0D-PINN methodology in an in-silico case. Next, we extend the 0D-PINN methodology to a full network comprising and 37 elastic vessels (21 junctions) replicating a human arterial system, and we compare our results with experimental data measured in [1]. Finally, 0D-PINNs are used to reconstruct signals and parameters in the thoracic arterial network of a healthy adult patient, where we compare our results with in-vivo data obtained in [2]. The objective of this section is twofold: on one hand, to prove that it is possible to accurately infer central aortic flow rate with 0D-PINNs using a single pressure waveform recorded in a terminal vessel (in this case, the LLCA) where measurements are non-invasive, as well as to test the robustness of the 0D-PINN when fed with real clinical (in-vivo) data instead of synthetic (in-silico) data. The clinical significance of this study lies in the ability of 0D-PINNs to reconstruct an otherwise intrusive measurement, the flow at the aortic root, using a non-invasive tonometric measurement at the patient's neck.

2. Lumped parameter models of elastic vessels

This section presents the basic equations of the Windkessel (0D) approximation to a network of elastic vessels. Windkessel models do not take into account variations in space, because variables are averaged along vessels's length. If the vessel walls are considered to be thin and homogeneous, the stress $\tau = \epsilon E_{wall}$, with ϵ the wall strain and E_{wall} the wall Young modulus, can be linked to the internal pressure p [1]:

$$\tau = \frac{r(p - p_o)}{e}, \tag{1}$$

where r is the vessel radius, e denotes the vessel wall thickness, $\sigma = 0.5$ is the Poisson ratio and p_o is the external pressure. The strain ϵ is related to the change in vessel radius $\eta = r - r_o$ via Laplace's law [46]:

$$\epsilon = \frac{1}{1 - \sigma^2} \frac{\eta}{r_o}, \tag{2}$$

where r_o is the vessel radius at $p = p_o$. Combining (1) and (2) and approximating $1/r$ by $1/r_o$, we have:

$$p = p_o + \frac{4E_{wall}\, e}{3\,(r_o)^2}\, \eta. \tag{3}$$

By expressing η in terms of the cross-sectional area, we arrive at the elastic vessel law:

$$p = p_o + k\left(\sqrt{\alpha} - 1\right), \qquad \alpha = \frac{A}{A_o}. \tag{4}$$

where $A_o = \pi r_o^2$ and

$$k = \frac{4\sqrt{\pi}\, E_{wall}\, e}{3\sqrt{A_o}}. \tag{5}$$

is the vessel wall stiffness coefficient. This equation is used as a constitutive law to close the system of equations. It relates the pressure of a vessel to the changes in volume according to its elastic properties.

The equations of 0D models can be obtained by averaging 1D equations in space [47]. The mass and momentum conservation equations in a single vessel, bounded by two junctions, can be written as follows:

$$\begin{cases} C\frac{dp}{dt} = q^{in} - q^{out} \\ p - p^{out} = Rq^{out} + L\frac{dq}{dt} \end{cases}, \tag{6}$$

where q^{in}, q^{out} are the inlet and outlet flow rates, p the transmural pressure at the entrance of the vessel and p^{out} the transmural pressure at the end of the vessel. R, L and C are the aggregated or lumped parameters. Lumped parameters comprise the dynamics of the flow into spatially averaged parameters. Resistance R relates to the friction of the flow against the arterial wall, inertance L accounts for the conduit function (acceleration of the flow) and compliance C relates to the reservoir function (capacity to accumulate blood volume). Lumped parameters are defined as:

$$R = R_o\alpha^{-5/2}, \quad L = L_o\alpha^{-1}, \quad C = C_o\alpha^{1/2}, \tag{7}$$

with

$$R_o = \frac{\beta\pi\mu l^3}{V_o^2}, \quad L_o = \frac{\rho l^2}{V_o}, \quad C_o = \frac{V_o}{\rho c_o^2}, \quad c_o^2 = \frac{k}{2\rho}. \tag{8}$$

Thus, $V_o = A_o l$ is the reference volume of the vessel and $\alpha = \frac{V}{V_o}$ is the deformation rate of the volume V with respect to the diastolic volume. Additionally, l is the length of the vessel, ρ and μ are the density and viscosity of blood and β is a coefficient related to the velocity profile of blood inside the vessel. Here we choose $\beta = 9$ which represents a parabolic profile. Lastly, c_o is the reference Pulse Wave Velocity (PWV), and is mostly used in the clinical setting as an indicator of arterial stiffness.

The system of ODE time-dependent equations is analogous to that used to find the voltage and electric current in an electric network. Thus, it is usual to represent lumped models using circuit diagrams, as depicted in Fig. 1.

In this formulation, external boundary conditions (representing vascular beds) are represented by 3-element lumped-parameter models with compliance and resistance. Their parameters are equivalent to those of a compliant vessel, but now lumped parameters are constant in time and there is no inertance parameter. The most common external boundary model includes a reflection coefficient Z^b, a resistive element R^b and a compliant element C^b. The three-element Windkessel model is an enhancement of the original two-element Windkessel model [48], as it can reproduce wave reflection phenomena by introducing a characteristic impedance term Z^b [49]. Although extensions to a four-element Windkessel model have been proposed, the three-element version remains the most commonly used [49]. When coupling an external boundary (0D lumped-parameter three-element Windkessel model) with a vessel (for example, an artery), the system of governing equations reads

$$\begin{cases} C\frac{dp}{dt} = q^{in} - q^b \\ p - p^b = (R + Z)\, q^b + L\frac{dq^b}{dt} \\ p^b - p^{out} = R^b\left(q^b - C^b\frac{dq^b}{dt}\right) \end{cases}, \tag{9}$$

with q^b, p^b the flow rate and pressure at the vascular bed, p^{out} being the asymptotic vascular pressure at the external boundary and R^b, Z^b, C^b being its 0D lumped parameters. The final configuration is depicted in Fig. 1.

These basic pieces can be assembled together to create networks by assuming that the outflow pressure of the *parent* vessel is the same as the inflow pressure of the *daughter* vessel. An example illustration of one junction with three vessels is depicted in Fig. 2.

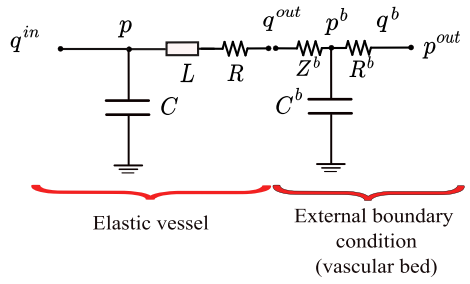


Fig. 1. Section 2. 4-element lumped-parameter model.

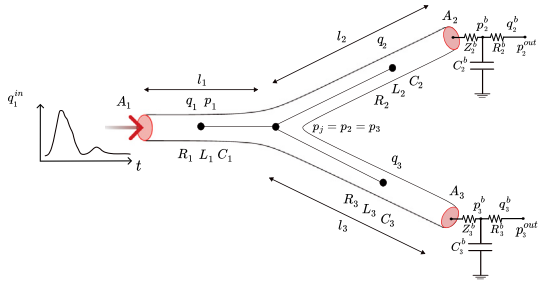


Fig. 2. Section 2. Representation of a single junction of elastic vessels.

Thus, any junction of n_p parents and n_d daughters can be described by a system of equations:

$$\begin{cases} \sum_i^{n_d} C_i \frac{dp_i}{dt} = \sum_j^{n_p} q_j - \sum_i^{n_d} q_i \\ p_i - p_j = R_i q_i + L_i \frac{dq_i}{dt} \quad \forall \quad i \in \mathcal{P} \\ p_J - p_i^b = \left(R_i + Z_i^b\right) q_i - L_i \frac{dq_i}{dt} \quad \forall \quad i \in \mathcal{D} \\ p_i^b - p_i^{out} = R_i^b q_i^b - R_i^b C_i^b \frac{dp_i^b}{dt} \quad \forall \quad i \in \mathcal{D} \end{cases}, \tag{10}$$

where q and p are the flow and pressure at the inlet node of each vessel and p_j is the pressure at the junction, assumed to be common for all vessels, n_p is the number of parent vessels and n_d is the number of daughter vessels. In a complete vascular network comprising n_j junctions, n_v vessels, and n_s external boundary conditions, the system of equations in (10) is enforced at every junction. The final two relations are included only when the daughter segments are terminal vessels; otherwise, they are omitted.

3. 0D-PINN models

PINNs can be used as forward solvers, replicating the functioning of a numerical solver, but they can also incorporate additional data available (initial, boundary or domain data) to reconstruct signals and parameters, as inverse solvers. In this manuscript we will apply different PINNs to the following test cases:

Test Case A involves a vessel bifurcation (as shown in Fig. 2), designed to showcase the construction and operation of the 0D-PINN framework under periodic boundary conditions, wherein the lumped elements are used to define a purely resistive external boundary condition (the compliance is null).

Test Case B extends the methodology to a full-body mock arterial circulatory system constructed in the laboratory [1] with 37 vessels and 21 junctions. Again, the lumped elements used to define the external boundary conditions are resistances.

Test Case C presents a clinical scenario in which 0D-PINNs are applied to predict and infer hemodynamic variables in the thoracic aorta of a healthy adult. The model is validated against in-vivo 4D flow MRI measurements [2].

Table 1
Summary of forward problems studied.

Category	Forward Test Case A	Forward Test Case B	Forward Test Case C
Description	Single junction of elastic vessels	Experimental mock arterial network built in [1]	Human thoracic aorta using 4D-MRI [2]
Objective	Validate 0D-PINN for wave prediction	Predict flow and pressure in experimental set-up	Predict flow pressure waveforms in all vessels
Training Data	Inlet flow + initial conditions vs periodicity constraints	Inlet flow + initial conditions vs periodicity constraints	Aortic root flow rate + periodicity constraints
Ground Truth Data	0D-Picard integration	Flow and pressure in-vitro measurements	Flow and pressure in-vivo measurements
Output	Flow and pressure waveforms at parent and daughter vessels	Flow and pressure waveforms in all vessels	Flow and pressure waveforms in aorta and additional branches

Table 2
Summary of inverse problems studied.

Category	Inverse Test Case A	Inverse Test Case B	Inverse Test Case C
Description	Single junction of elastic vessels	Experimental mock arterial network built in [1]	Human thoracic aorta using 4D-MRI [2]
Objective	Infer lumped resistance parameters at external boundary conditions	Reconstruct flow and pressure + vessels stiffness coefficients	Reconstruct aortic root flow rate + Windkessel resistance parameters at vascular beds
Training Data	Luminal area + initial conditions vs periodicity constraints	Cross sectional area + initial conditions vs periodicity constraints	Left common carotid artery pressure + initial conditions
Ground Truth Data	Lumped resistance parameters used in 0D-Picard integration	Flow and pressure in-vitro measurements	Flow and pressure in-vivo measurements
Output	Flow and pressure waveforms in parent and daughter vessels + inferred lumped resistance parameters	Flow and pressure waveforms in all vessels + vessels stiffness coefficients	Aortic root flow waveform + Windkessel resistance parameters at vascular beds

A summary of the forward problems addressed in these three cases is provided in Table 1, while the corresponding inverse problems are outlined in Table 2.

3.1. Methods

In this section, we elaborate on the general formulation of 0D-PINNs for networks of elastic arteries — that is, the integration of the 0D coupled system of equations and the construction of the loss function used in the neural network optimization process. We present the equations defining the different loss components of the loss function of the PINN for an arbitrarily large arterial network, addressing both forward and inverse problems. In the subsequent subsections corresponding to Test Cases A, B, and C, we specify the details of each arterial configuration and the type of data available – in-silico, in-vivo, or in-vitro, depending on the case.

Signals that we assume are available as data and are used to train the algorithm are marked with a *tilde*. The PINN usually requires their fitting, interpolation or extrapolation at the collocation points. If the *tilde* does not appear, it means the variable is being predicted as an output of the model.

3.1.1. Forward problem

In the forward problem, the 0D-PINN acts as a forward solver, similar to any traditional numerical CFD solver. In a network composed of n_j junctions comprising n_p parent vessels and n_d daughter vessels and n_s external boundary conditions, the ODE loss $\mathcal{L}_{ODE}$, also known as *collocation* loss, associated with the minimization of the ODE residual reads:

$$\begin{aligned}\mathcal{L}_{ODE} = &\frac{1}{n_{coll}}\sum_{c=1}^{n_{coll}}\left(\sum_{j=1}^{n_j}\sum_{d=1}^{n_d} l_d\left(C_d\frac{dp_p}{dt}\Big|_{t_c} - \sum_{p=1}^{n_p} q_p + \sum_{d=1}^{n_d} q_d(t_c)\right)\right)^2 \\ &+ l_1\frac{1}{n_{coll}}\sum_{c=1}^{n_{coll}}\left(p_1(t_c) - p_{j=1}(t_c^i) - R_1\tilde{q}_1^{in}(t_c) - L_1\frac{d\tilde{q}_1^{in}}{dt}\Big|_{t_c}\right)^2 \\ &+ \frac{1}{n_{coll}}\sum_{c=1}^{n_{coll}}\left(\sum_{j=1}^{n_j}\sum_{d=1}^{n_d} l_d\left(p_j(t_c) - p^*(t_c) - (R_d + Z_d^b)q_d(t_c) - L_d\frac{dq_d}{dt}\Big|_{t_c}\right)\right)^2 \\ &+ \frac{1}{n_{coll}}\sum_{c=1}^{n_{coll}}\left(\sum_{s=1}^{n_s} l_s\left(p_s^b - p_s^{out} - R_s^b q_s^b(t_c) + R_s^b C_s^b\frac{dp_s^b}{dt}\Big|_{t_c}\right)\right)^2\end{aligned} \tag{11}$$

where the first term is the sum of conservation equations for all junctions, the second term is the momentum equation concerning the network's inlet root vessel, the third term is the momentum equation in the rest of the vessels and the last term is the external boundary condition at the terminal vessels. In (11), t_c are the collocation points in the time dimension, l_i are constant coefficients (also known as Lagrange multipliers [26]) that account for the relative importance of each of the addends of the loss function, $p^* = p_d^b$ for terminal daughter vessels, and for vessels between junctions $p^* = p_d^{j+1}$ and $Z_d^b = 0$.

The fitting of initial conditions is achieved through the loss $\mathcal{L}_o$ when they are available:

$$\mathcal{L}_o = \frac{1}{n_{coll}}\sum_{i=1}^{n_{coll}}\left(\sum_{d=1}^{n_d} l_d\left(q_d(t_o) - \tilde{q}_{o,d}\right)\right)^2 + \frac{1}{n_{coll}}\sum_{i=1}^{n_{coll}}\left(\sum_{d=1}^{n_d} l_d\left(p_d(t_o) - \tilde{p}_{o,d}\right)\right)^2 \tag{12}$$

If the solution is periodic, as an alternative to the loss in (12), a constraint to enforce periodicity $\mathcal{L}_{cycle}$ can be defined, namely

$$\mathcal{L}_{cycle} = \sum_{d=1}^{n_d} l_d\left(q_d(t_o) - q_d(t_o+T)\right)^2 + \sum_{d=1}^{n_d} l_d\left(p_d(t_o) - p_d(t_o+T)\right)^2 \tag{13}$$

with T the cycle duration, without providing the actual value of q or p at time t_o. Finally, the fitting of available boundary conditions, in this case the flow rate at the root vessel $\tilde{q}_1^{in}$, is expressed through the $\mathcal{L}_{bc}$ as

$$\mathcal{L}_{bc} = l_1\frac{1}{N_{bc}}\sum_{i=1}^{N_{bc}}\left(q_1(t_i) - \tilde{q}_1^{in}(t_i)\right)^2. \tag{14}$$

with N_{bc} the number of data points sampled at time instants t_{bc} at the inlet boundary condition.

The global loss function of the 0D-PINN, $\mathcal{L}_{0D-PINN}$, is constructed as a weighted sum of the described individual losses, and has the following form for the forward setting:

$$\mathcal{L}_{0D-PINN}^F = \mathcal{L}_{ODE} + \mathcal{L}_{bc} + \mathcal{L}_o + \mathcal{L}_{cycle}. \tag{15}$$

Predicted variables $\{q_1, \dots, q_j, p_1, p_{n_j}\} = \mathcal{PINN}(t;\boldsymbol{\Theta})$ are evaluated at the collocation points t_c, parameterized with $\boldsymbol{\Theta} = \{\boldsymbol{w}, \boldsymbol{b}\}$ the set of weights $\boldsymbol{w}$ and biases $\boldsymbol{b}$ of the neural network. A diagram of the training step of a forward 0D-PINN is depicted in Fig. 3. In the figure, note that time t fed into the PINN can represent either training or test data. All derivatives are evaluated at the collocation points t_c.

Important extra considerations regarding the choice of predicted variables must be accounted for at the stage of building the PINN. As a general rule, redundant variables tend to slow down and complicate the optimization process, since they introduce unnecessary additional terms into the loss function. Therefore, the model should output only essential variables. For instance, simultaneously predicting p_j, p_1 and p_2 in the junction in Fig. 2 would be redundant, since $p_j = p_1 = p_2$, with p_j being the pressure at the junction and p_1, p_2 the pressures of the two daughters v_1 and v_2 respectively. Enforcing this equality as a loss, would therefore make optimization more complex. Thus, only a single output variable for the three pressures is used.

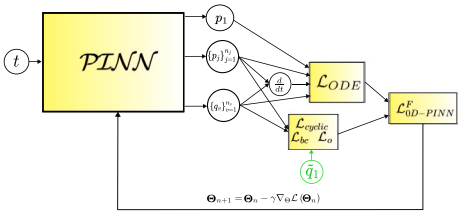


Fig. 3. Section 3.2.1. Test case A (forward). Diagram of the forward 0D-PINN training step for a network of n_j junctions, with input, output variables and different terms of the loss function. Inlet boundary condition flow rate data $\tilde{q}_1$ is shown in green.

Another important consideration is that neural networks often perform better when input and output variables, as well as their derivatives, are of the same order of magnitude. A common strategy is to scale them around 10^0. By multiplying the inputs and outputs by these scaling factors, the variables can be brought close to unity in a straightforward manner. For example, if the flow rate q is order 10^3, the neural network can be trained to predict $10^3 q$, so that the scaled q stays close to 1. In the following, this approach is taken, hence the equations do not need to be explicitly nondimensionalized.

3.1.2. Inverse Problem

One of the most relevant advantages of PINNs is that the formulation of the inverse problem is almost identical to that of the forward problem. The difference lies in treating the unknown signals χ and/or parameters $\boldsymbol{p}$—those to be reconstructed or inferred—as trainable variables, while adding additional loss terms that account for the neural network's fitting to the available data and other possible constraints. In the inverse problem, an extra loss term, $\mathcal{L}_D$, related to the available dataset D, is included to perform the fitting of a specific signal χ for which some data D is available, and reads:

$$\mathcal{L}_D = \sum_{d=1}^{n_v}\sum_{j=1}^{N_\chi}\sum_{i=1}^{n_\chi} l_d\left(\chi_j(t_i) - \tilde{\chi}_j(t_i)\right)^2$$

with n_v the number of vessels in the network, N_χ the number of signals χ available as measurement data per vessel v_i, sampled at discrete times t_i. The entire set of trainable parameters is now $\boldsymbol{\Theta}^+ = \{\boldsymbol{w}, \boldsymbol{b}, \boldsymbol{p}\}$. A diagram of the training step of an inverse 0D-PINN is depicted in Fig. 5. The global loss function for the inverse setting now is

$$\mathcal{L}_{0D-PINN}^I = \mathcal{L}_{ODE} + \mathcal{L}_{bc} + \mathcal{L}_o + \mathcal{L}_{cycle} + \mathcal{L}_D. \tag{16}$$

3.2. Test case A

In Test case A, we show the flow computation in a junction with 3 elastic vessels, one parent vessel and two daughter vessels, using in-silico data generated with a numerical solver (Picard time integration scheme). We feed the parent vessel with an inlet flow that forks into the two daughter vessels, which are terminal vessels and are coupled with an external boundary conditions modeled with a single resistance (compliance in the external boundary condition is neglected). The available in-silico dataset includes the inlet flow rate curve in time $\tilde{q}_1^{in}$ at the parent vessel, labelled v_1. Lumped parameter constants for the vessels

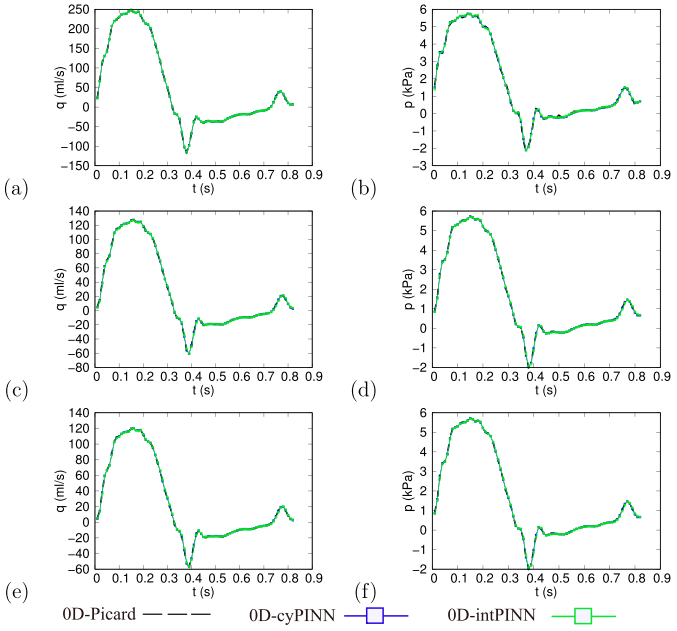


Fig. 4. Section 3.2.1. Test case A (forward). Flow rate and pressure are shown for vessels v_1 (parent), v_2 and v_3 (daughters): 0D-intPINN (-■-), 0D-cyPINN (-■-) and 0D-Picard (– –).

Table 3
Test case A (forward). Comparison of relative errors ϵ (%) in pressure and flow rate between the 0D-intPINN and 0D-cyPINN predictions against the 0D-Picard reference.

Vessel	0D-intPINN		0D-cyPINN	
	ϵ_p(%)	ϵ_q (%)	ϵ_p(%)	ϵ_q (%)
v_1	0.8604	0.1174	1.0039	0.2598
v_2	0.2021	0.2593	0.4865	0.5313
v_3	0.2021	0.2408	0.4865	0.5258

$R_{o,v}, L_{o,v}, C_{o,v}$ and external boundary conditions Z_v^b, R_v^b, C_v^b, located at the outlet of each of the terminal vessels, are also known in the forward setting, but are not marked with a tilde, to keep the notation uncluttered.

3.2.1. Forward problem

In the forward problem the 0D-PINN predicts the flow and pressure curves in the three vessels of the network. Fig. 4 shows a comparison of the 0D-PINN and 0D-Picard time integration scheme, *0D-Picard* for short, as the reference solution for the flow, pressure and volume waveforms in two different scenarios.

If the 0D-PINN is fed with initial conditions (12), then we denote the algorithm as 0D-intPINN. If no information about the value of the initial condition is provided and periodicity across different cardiac cycles is enforced (13), then we denote the algorithm as 0D-cyPINN.

Results in Fig. 4 show that both the 0D-PINN fed with initial condition data (0D-intPINN) as well as the 0D-PINN constrained by periodicity conditions (0D-cyPINN) are in very close agreement with the 0D-Picard reference solution.

Error metrics defined Appendix A are used to evaluate the discrepancy between the 0D-PINN and the 0D-Picard reference solution. Table 3 summarizes the error in the flow rate and pressure using the 0D-intPINN and the 0D-cyPINN. Errors for the flow rate and pressure lie below 1.1%. We observe slightly higher errors for the 0D-cyPINN compared to the 0D-intPINN across all vessels, although the differences remain minimal in both pressure and flow rate predictions.

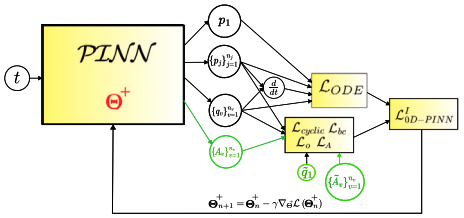


Fig. 5. Section 3.2.2. Test case A (inverse). Diagram of the inverse 0D-PINN training step with input, output variables and different terms of the loss function. Inlet flow rate data and area data are shown in green, while learnable parameters are shown in red.

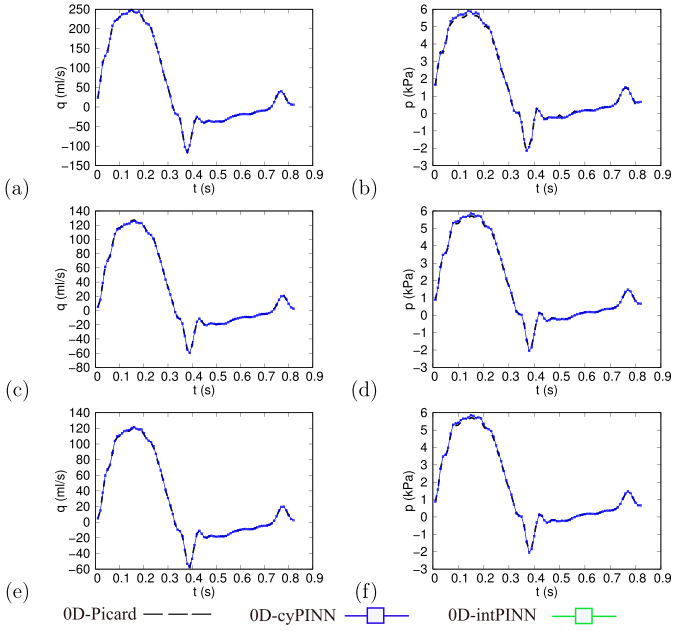


Fig. 6. Section 3.2.2. Test case A (inverse). Parent vessel v_1: 0D-cyPINN (-■-) predictions versus 0D-Picard time integration algorithm reference solution (– –). Flow rate and pressure are plotted for each of the three vessel of the junction.

3.2.2. Inverse problem

In the current configuration of one isolated junction, we have an inverse problem where R_2^b, R_3^b are trainable parameters, initialized at the beginning of the training and changing over the iterations, together with the weights and biases of the neural network. Loss term $\mathcal{L}_A$ accounts for the fitting of the vessel area in the three vessels of the junction and reads:

$$\mathcal{L}_A = \sum_{i=1}^{N_a} \sum_{v=1}^{n_v} l_v \left(A_v(t_i) - \tilde{A}_v(t_i) \right)^2$$

with N_a the number of data points of the area waveforms of each vessel $\tilde{A}_v$ sampled at time instants t_i.

The entire set of trainable parameters is now $\boldsymbol{\Theta}^+ = \{\boldsymbol{w}, \boldsymbol{b}, R_2^b, R_3^b\}$. A diagram of the training step of an inverse 0D-PINN is depicted in Fig. 5. Parallel to the inference of the resistance parameters, the flow, area and pressure curves of each vessel must be recovered, in a similar way as in the forward problem. The inference of resistance parameters improves as the prediction of the flow and pressure curves at the different vessels improves. In this case, the global loss function reads

$$\mathcal{L}^I_{0D-PINN} = \mathcal{L}_{ODE} + \mathcal{L}_{bc} + \mathcal{L}_o + \mathcal{L}_{cycle} + \mathcal{L}_A. \tag{17}$$

Fig. 6 shows a comparison of the 0D-PINN and 0D-Picard predictions for the flow and pressure waveforms. The resistance parameters R_1^b and

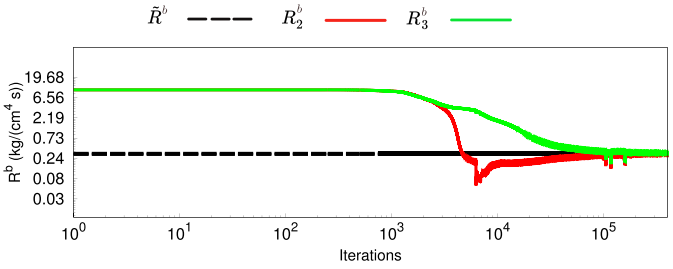


Fig. 7. Section 3.2.2. Test case A (inverse). Evolution of the inverse 0D-PINN inferred Windkessel resistance parameters R_v^b over the training iterations in Test case B.

Table 4
Test case A (inverse). Error ϵ (%) between 0D-cyPINN and 0D-Picard predictions of flow rate, pressure, volume and resistance in Test case A, inverse problem.

Vessel	$\epsilon(p)$(%) (%)	$\epsilon(q)$ (%)	e_R (%)
v_1	1.247098	0.044760	–
v_2	0.838141	0.351988	3.899253
v_3	0.838141	0.375021	1.711910

R_2^b of the two external boundary conditions in the isolated junction must converge to ground truth values $\tilde{R}^b$, used to generate the in-silico area waveforms. Their convergence is plotted in Fig. 7. It is observed that the convergence is achieved after approximately 10^5 iterations of the algorithm.

Table 4 summarizes the error for the flow rate and pressure reconstruction as well as for the resistance inference. Errors for the prediction of the flow rate and pressure waves lie below 1.3%, while for the inference of the resistances, they lie below 4%.

Values for the parameters of the vessels and hyperparameters of the 0D-PINN in Test case A are shown in Appendix B.

3.3. Test case B

In Test case B, we apply the 0D-PINN methodology to the experimental circulatory system built in [1] using silicone vessels. The network was composed of $n_j = 21$ junctions interconnecting $n_v = 37$ silicone elastic vessels, terminating in $n_s = 16$ peripheral resistance vessels mimicking vascular beds. Vessels are silicone vessels resembling the geometrical and physiological/mechanical properties of a real human arterial circulatory system. Although the mechanical properties of silicone vessels in the network differ significantly from those of real arteries, this mock network enabled for the computational validation of a cardiovascular simulator using experimental measurements [1].

As detailed in [1], the network of elastic vessels was built using custom-made aluminum molds. A 65%–35% water–glycerol solution with a density of 1050 kg/m^3 and a viscosity: of 2.5 mPa·s at 30 °C was employed to have properties close to blood. Based on tensile tests, a constant Young's modulus of 1.2 MPa was adopted. While the setup only approximates systemic human circulation, it successfully reproduces physiologically relevant pulse waveforms in aortic vessels. In terminal branches, however, the waveforms are less realistic, since vascular beds are isolated resistances with no compliance elements.

Further details of the geometrical, mechanical and physiological properties of the network can be found in [1]. A representation of the network with the corresponding numbering of the junctions and vessels can be found in Fig. 8. The name of each artery can be found in Table 2 of the reference work [1]. Since Test case B features a network of compliant vessels that replicate a human arterial tree, we will refer to the vessels and external boundary conditions as *vessels* and *vascular beds*, respectively.

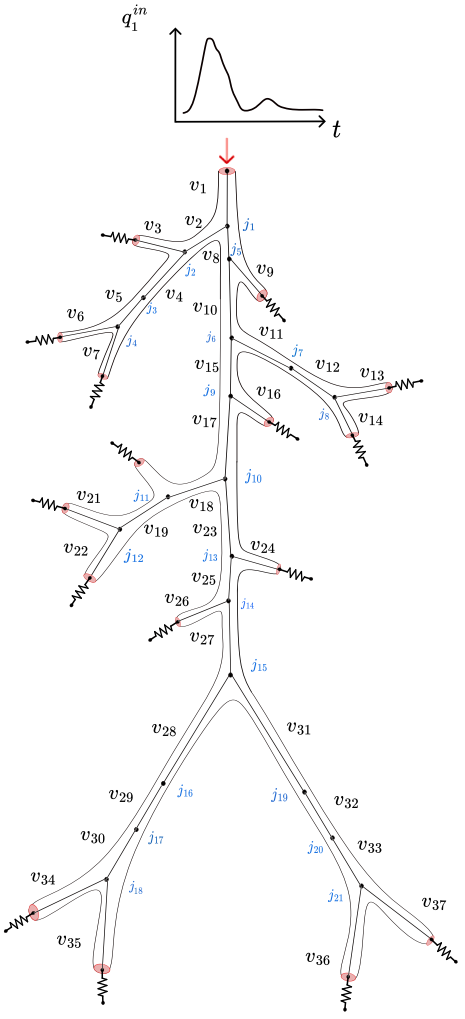


Fig. 8. Section 3.3. Test case B. Representation of the mock arterial network with $n_j = 21$ junctions, $n_v = 37$ elastic vessels and $n_s = 16$ peripheral resistances.

3.3.1. Forward problem

In the forward setting, the flow rate and pressure are computed in each vessel of the network with the 0D-Picard time integration numerical scheme as well as using the 0D-intPINN and the 0D-cyPINN, and compared against the *in-vitro* measurements taken by the authors in [1] at the middle point of each vessel, at 11 of the 70 locations of the experimental mock circulatory system. The predictions of the flow and pressure waveforms in time compared with the *in-vitro* data are shown in Figs. 9 and 10. While for the sake of brevity only the more representative locations have been shown, the subsequent results are applicable to the rest of the locations (vessels) of the experimental network.

Fig. 9 shows the predictions at three vessels located in the upper trunk of the experimental network (v_1: Ascending aorta (AA), v_7: Right ulnar artery (RUA), v_{10}: Aortic arch II). Both 0D-intPINN and 0D-cyPINN show good agreement with the experimental measurements for pressure, and less agreement for the measurements with flow. This can be explained by considering that pressure variations in time are small along the vessels, while flow variations in time can be large depending on the measurement site. Therefore, the 0D model, where all variables are considered constant in the vessel is able to reproduce pressure waveforms.

Fig. 10 shows the predictions at three vessels located in the lower trunk of the experimental network (v_{27}: Abdominal aorta II, v_{29}:

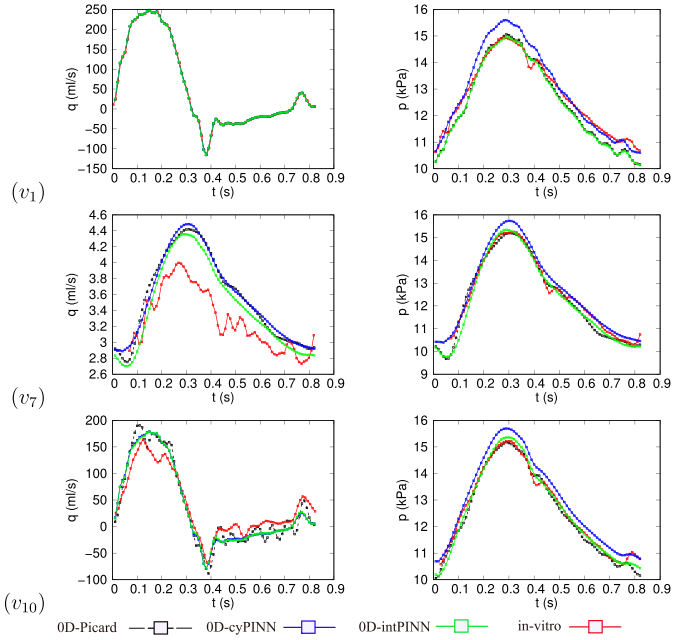


Fig. 9. Section 3.3. Test case B (forward). Flow rate and pressure predictions for each selected vessel, with v_1: AA, v_7: Right ulnar, v_{10}: Aortic arch II. 0D-intPINN (-■-), 0D-cyPINN (-■-), 0D-Picard solver (-■-) predictions and *in-vitro* data (-■-) are shown.

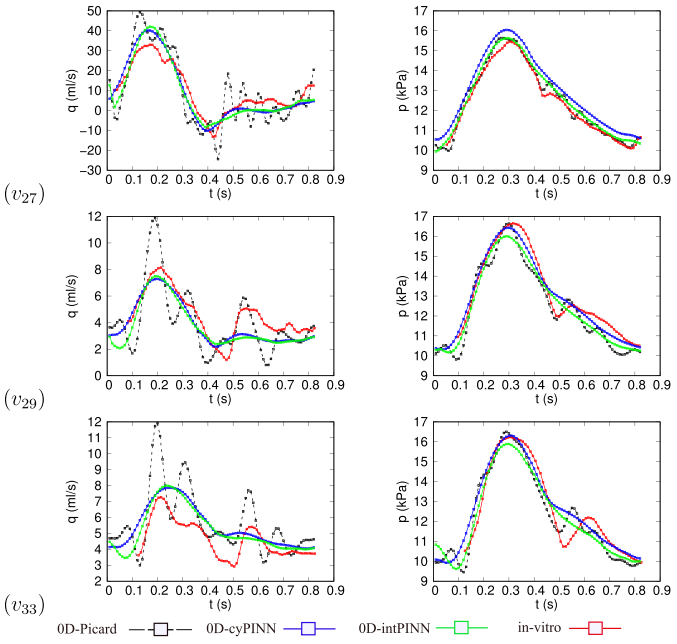


Fig. 10. Section 3.3. Test case B (forward). Flow rate and pressure predictions for each selected vessel, with v_{27}: Abdominal aorta II, v_{29}: R. iliac-femoral II, v_{33}: L. iliac-femoral III. 0D-intPINN (-■-), 0D-cyPINN (-■-), 0D-Picard solver (-■-) predictions and *in-vitro* data (-■-) are shown.

R. iliac-femoral II, v_{33}: L. iliac-femoral III). The 0D-Picard produces higher-frequency oscillations in the flow signal compared to the in-vitro data for flow at the midpoint of the vessel. In contrast, we observe that the 0D-PINN attenuates these high-frequency oscillations. This behavior is expected, since the standard 0D-PINN employs a basic multilayer perceptron (MLP) architecture with hyperbolic tangent (tanh) activation functions, which naturally smooth out high-frequency components. Interestingly, this inherent property of the 0D-PINN proves beneficial,

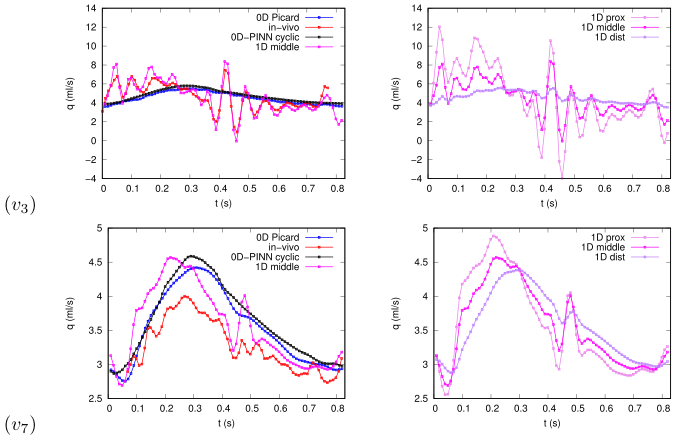


Fig. 11. Section 3.3. Test case B (forward). Left: flow-rate predictions for vessels v_3 (upper) and v_7 (lower) (0D-cyPINN -■-, 0D-Picard (-■-), in-vitro (-■-), 1D solver prediction at middle point (-■-)). Right: 1D solver predictions at middle point/proximal point (start of artery)/distal point (end of artery) for vessels v_3 (upper) and v_7 (lower) (-■-, -■-, -■-).

as it reproduces the in-vivo data more closely than the highly oscillatory 0D-Picard model.

To explain why the 0D-Picard model produces these oscillations, Fig. 11 (left) shows the flow waveforms from the 0D-Picard and 0D-cyPINN models compared with the 1D reference solution at the midpoint of the Right Carotid Artery (RCA) (v_3) and Right Ulnar Artery (RUA) (v_7), obtained using a 1D finite-volume scheme as described in [3]. We observe that the flow waveform prediction of the 1D solver at the midpoint agrees with the in-vivo data at the same location (measurements were made at the vessel's midpoint), since spatial variations are taken into account in the 1D formulation. Fig. 11 (right) compares the 1D model solutions at the proximal, middle, and distal positions (start, middle and end of the vessel) along the same artery. We observe that the flow waveform varies significantly depending on the location within the vessel. Consequently, the 0D-Picard model prediction, which averages these three spatial waveform discrepancies, exhibits a different shape, resulting from these variations of the actual flow wave along the artery. For the RCA (v_3, Fig. 11 - upper row), the 0D-Picard solution (left) shows no oscillations because it effectively represents an average between the strongly oscillatory behavior observed at the proximal and middle locations and the nearly flat waveform at the distal location. In the case of the RUA (v_7, Fig. 11 - lower row), the 0D-Picard solution (left) overestimates the flow. Once again, this behavior can be attributed to the fact that the 0D-Picard solution represents a spatially averaged quantity along the vessel, and thus fails to reproduce the flow reduction evident in the in-vivo signal and captured by the 1D model toward the distal end of the artery.

Error metrics for the pressure signal and mean pressure obtained using the 0D-Picard integration model and the 0D-cyPINN are shown in Table 5, providing the errors for the set of vessels resembling the aorta and the set of vessels in the first and second generations after the aorta. Errors in the pressure signal are low for both 0D-Picard and 0D-PINNs, below 7% in all cases. The mean error in pressure is below 5%. For the sake of brevity, not all vessels are shown in Figs. 9 and 10. We display only those vessels that best illustrate the oscillatory behavior or potential caveats of the method. Nevertheless, we also observe very good agreement in the remaining vessels. In particular, the 0D–Picard solution shows an excellent fit with the *in-vivo* data in the thoracic aorta I (v_{15}) (see Fig. 12).

3.3.2. *Inverse problem*

In this setting, the stiffness coefficients k_v of the n_v vessels are unknown and must be inferred. For the inference of k_v, two approaches are

Table 5
Section 3.3.2. Test case B. Relative errors ϵ (%) in pressure, average pressure, and Young's modulus for Test Case B across 0D-Picard and 0D-cyPINN models, for forward and inverse problems.

Location	Forward				Inverse		
	0D-Picard		0D-cyPINN		0D-cyPINN		
	$\epsilon(p)$ (%)	$\epsilon(\bar{p})$ (%)	$\epsilon(p)$ (%)	$\epsilon(\bar{p})$ (%)	$\epsilon(p)$ (%)	$\epsilon(\bar{p})$ (%)	$\epsilon(E)$ (%)
Aortic vessels	1.67	0.79	3.44	3.41	2.83	2.62	2.71
First gen.	3.23	1.79	3.37	2.35	2.80	1.81	2.68
Second gen.	3.79	1.70	3.55	2.26	2.80	1.51	2.68
v_1	2.54	1.92	1.96	1.76	1.93	1.33	2.71
v_3	2.58	1.97	2.18	1.60	1.75	1.23	2.68
v_7	2.11	1.26	1.88	1.38	1.79	0.92	2.68
v_{10}	1.50	1.12	2.47	2.50	2.13	1.91	2.66
v_{11}	3.86	3.74	0.96	0.19	1.10	0.38	3.21
v_{14}	2.48	0.80	2.64	1.93	2.27	1.23	3.56
v_{15}	1.48	0.03	3.56	3.64	3.02	2.82	3.91
v_{17}	0.95	0.14	3.49	3.50	2.65	2.58	0.27
v_{20}	2.72	0.47	4.41	3.92	3.49	3.03	0.23
v_{24}	2.19	1.23	5.02	4.80	3.94	3.87	2.69
v_{27}	1.73	1.06	4.75	4.75	3.75	3.75	2.64
v_{29}	4.81	4.02	2.74	0.85	2.53	0.80	3.25
v_{33}	4.53	1.15	3.90	1.15	3.18	0.63	5.28
v_{34}	6.78	3.03	6.11	3.47	4.35	2.38	4.48

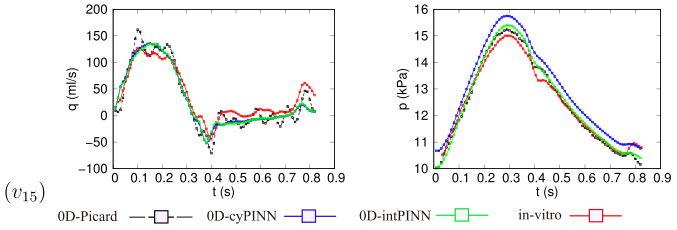


Fig. 12. Section 3.3. Test case B (forward). Flow rate and pressure predictions for Thoracic aorta I (v_{15}). 0D-intPINN (-■-), 0D-cyPINN (-■-), 0D-Picard solver (-■-) predictions and *in-vitro* data (-■-) are shown.

possible. The first assumes, according to [1], that all vessels are made of the same material and that a single Young modulus E can be inferred for all the vessels in the entire network, with the final values of k_v are defined in combination with the vessel area and wall thickness. The entire set of trainable parameters is in this case $\boldsymbol{\Theta}^+ = \{\boldsymbol{w}, \boldsymbol{b}, E\}$. In the second approach, we assume that the stiffness coefficients are independent of each other, and the entire set of trainable parameters is now $\boldsymbol{\Theta}^+ = \left\{\boldsymbol{w}, \boldsymbol{b}, k_{v1}, \ldots, k_{n_v}\right\}$.

In turn, we assume additional area waveform data $\{A_v(t)\}$ in all vessels is available through measurements across the entire cardiac cycle for both approaches. Since in the experimental work performed in [1] *in-vitro* vessel area signals were not reproduced in all vessels, we used an in-silico area dataset $\{\tilde{A}_v(t)\}_{v=1}^{n_v}$ obtained with the 0D-Picard numerical scheme as the training dataset. Fig. 13 displays the vessel area waveform for the vessels conducting the flow in the aorta to the Right Anterior Yibial (RAT) (v_1, v_2, v_8, v_{10}, v_{15}, v_{17}, v_{23}, v_{25}, v_{27}, v_{28}, v_{29} and v_{34}). As we show later, 0D-PINN is able to handle area signals of different orders of magnitude.

The evolution of E over the course of training iterations is shown in Fig. 14. This approach yields an estimate of E that closely matches the actual Young's modulus of silicone in the experiment, with a final error of only 3.16% after training.

The stiffness coefficients k_v, vary across different vessels according to Eq. (4) in the range 1–5 MPa. These stiffness parameters are trainable and are expected to converge toward their true (in-silico) values over the training iterations. Although the initial values of $\{k_v\}_{v=1}^{n_v}$ at the beginning of the training process can be arbitrary, closer initial estimates can reduce the overall training duration by accelerating convergence and improving the reconstruction of both the flow rate and pressure waveforms.

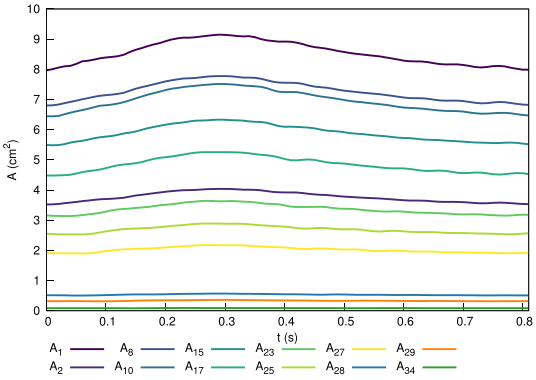


Fig. 13. Section 3.3.2. Test case B (inverse). Vessel area waveform for vessels v_1, v_2, v_8, v_{10}, v_{15}, v_{17}, v_{23}, v_{25}, v_{27}, v_{28}, v_{29} and v_{34}.

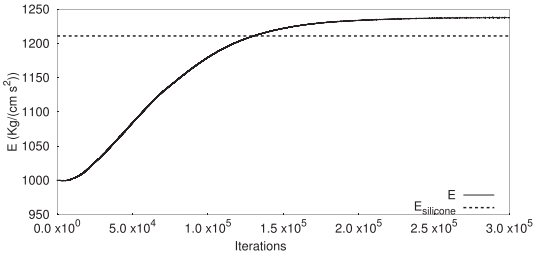


Fig. 14. Section 3.3.2. Test case B (inverse). Evolution of the estimated Young's modulus E during training, assuming identical material properties for all vessels in the network.

To demonstrate that convergence is achievable regardless of the initial values of the stiffness coefficients, three distinct initialization strategies are proposed:

1. I1: For each vessel, k_v is set to either $\tilde{k}_v + 20\%$ or $\tilde{k}_v - 20\%$, where the choice between + and − is made randomly.
2. I2: For each vessel, the value is sampled from a uniform distribution: $k_v \sim \mathcal{U}(0.85\,\tilde{k}_v,\ 1.15\,\tilde{k}_v)$.
3. I3: Initialize all vessels with the same stiffness parameter, specifically setting $k = k_1 = \ldots = k_{n_v}$.

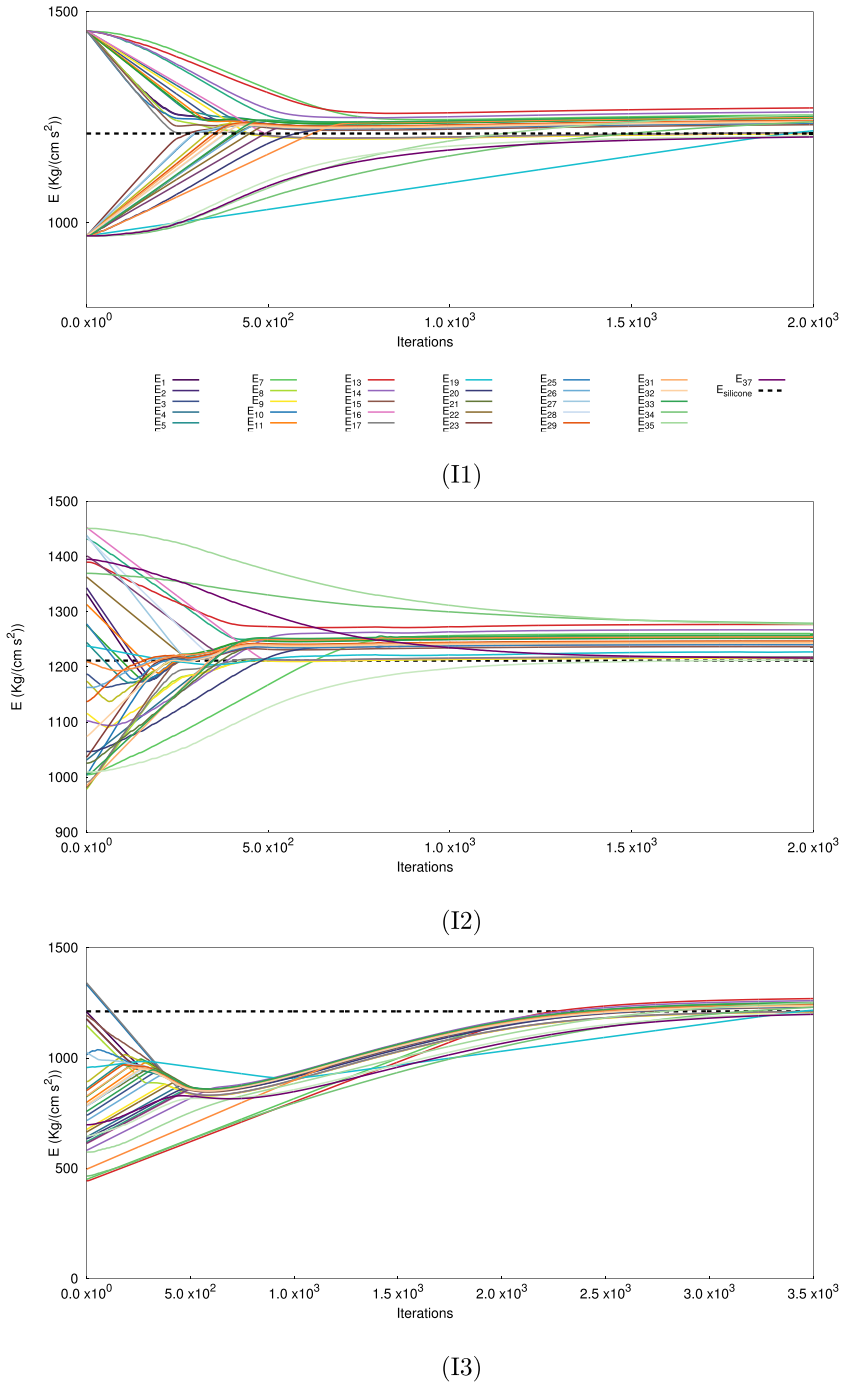


Fig. 15. Section 3.3.2. Test case B (inverse). Evolution of the inferred Young modulus by the inverse 0D-cyPINN versus the training iterations, using initializations I1 (upper), I2 (middle) and I3 (lower).

Given the wide range of variation in the stiffness coefficients, in order to simplify the convergence analysis and the accuracy of the predictions, we explore the results in terms of the individual Young modulus obtained for each vessel, which must be similar to the experimental one, $E_{silicone} = 1.2$ MPa. In this way, deviations produced by the vessel geometry are not considered. For initializations I1, I2 and I3, convergence of the inferred Young modulus E toward the real value of silicone $E_{silicone}$ is shown in Fig. 15. Convergence is achieved, and the value at the last training iteration is used to calculate subsequent error metrics.

Fig. 16 shows the error e_E for the reconstruction of E in all vessels v_i of the experimental network, using the different proposed initialization for E. We observe that all the errors lie below 6%. The reconstruction results for flow rate and pressure waves with the 0D-cyPINN (enforcing periodicity constraints) and 0D-intPINN (training with initial conditions) using initialization I1 are shown in Figs. 17 and 18. The results obtained using initialization I2 and I3 are nearly identical and are therefore omitted for brevity. The flow and pressure are computed in each vessel with the 0D-Picard time integration scheme and the 0D-PINN, and compared against the *in-vitro* measurements published by the authors [1] in 6 of the 70 locations of the experimental mock circulatory system.

We observe a close agreement between the inverse 0D-cyPINN and 0D-intPINN predictions, with the 0D-Picard solutions. Unlike in the

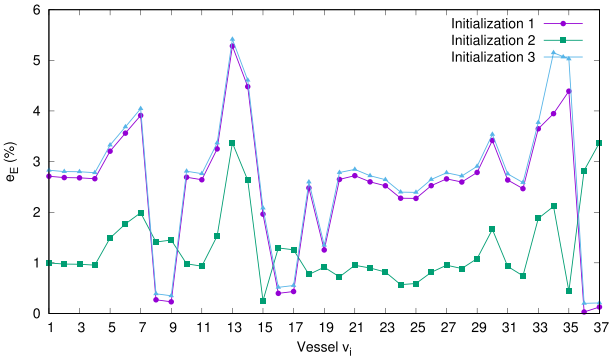


Fig. 16. Section 3.3.2. Test case B (inverse). Error e_E between the inverse 0D-cyPINN Young modulus E reconstruction and $E_{silicone}$ for each vessel v_i, using initializations I1, I2 and I3.

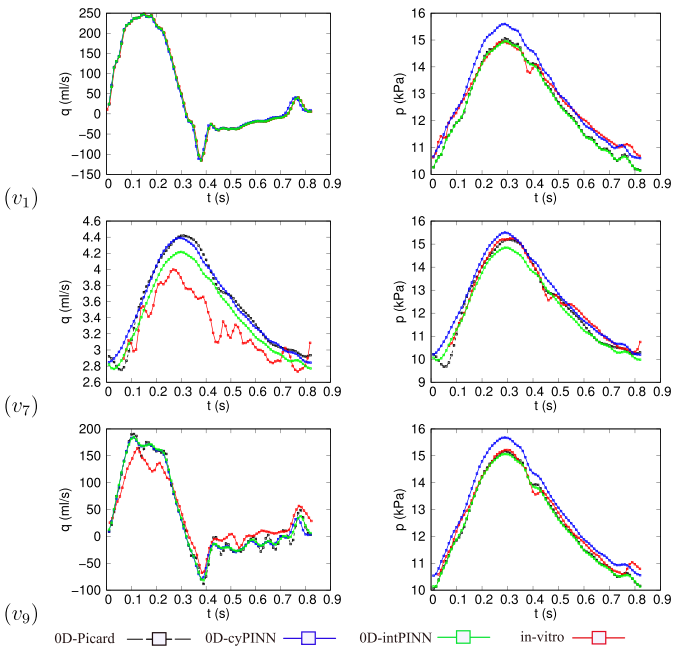


Fig. 17. Section 3.3.2. Test case B (inverse). Flow rate and pressure predictions for each selected vessel, with v_1: AA, v_7: Right ulnar, v_{10}: Aortic Arch II. 0D-intPINN (-■-), 0D-cyPINN (-■-), 0D-Picard solver (-■-) predictions and *in-vitro* data (-■-) are shown.

forward setting, the inverse 0D-cyPINN and 0D-intPINN are able to recover the oscillatory patterns introduced by the 0D-Picard method. This is because the PINNs in the inverse setting are trained using luminal area data $A_v(t)_{v=1}^{n_v}$, which inherently contain these oscillatory features. As a result, the inverse process enables the reconstruction of similar oscillations in the pressure and flow waveforms. As observed before, this behavior is beneficial in most regions of the network, as real physiological data also exhibit such oscillations. However, at specific locations, such as v_{29}, the forward PINN introduced a damping effect that aligned more accurately with in-vitro measurements than the highly oscillatory 0D-Picard solution.

Errors between the 0D-cyPINN predictions and the in-vitro reference data in this inverse case are reported in Table 5, using initialization I1 for wall stiffness. Errors for initializations I2 and I3 are very similar to I1, and are not shown for brevity. Errors in the signal pressure are slightly lower than in the forward case, below 5% in all cases, while mean errors for pressure are similar. Errors e_E between the inferred value of E with respect to the real value are reported to be below 6% in all cases.

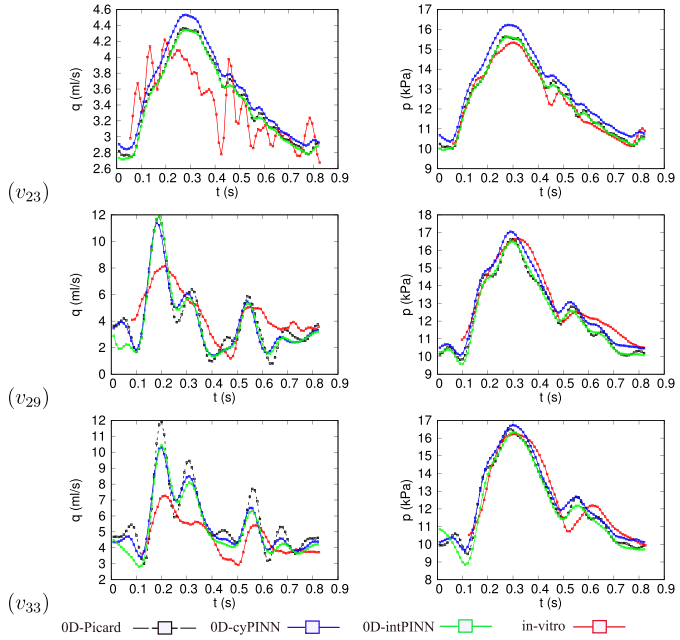


Fig. 18. Section 3.3.2. Test case B (inverse). Inverse predictions for each selected vessel, with v_{27}: Abdominal aorta II, v_{29}: R. iliac-femoral II, v_{33}: L. iliac-femoral III. 0D-intPINN (-■-), 0D-cyPINN (-■-), 0D-Picard solver (-■-) predictions and *in-vitro* data (-■-) are shown.

In Table 5 we evaluate the same metrics now comparing 0D-cyPINN predictions with respect to experimental in-vitro measurements for flow rate and pressure, using initialization I1. Results show that, with respect to the in-vitro measurements, the 0D-cyPINN reconstruction of flow and pressure has lower errors than the 0D-Picard.

3.4. Test case C

We study the application of 0D-PINNs to the aortic thoracic network analyzed in [2]. The authors in [2] collected MRI measurements of a healthy adult's thoracic aortic network, they segmented and accurately characterized the geometry of the different vessels, including: Ascending Aorta (AA), upper cerebral branches (Brachiocephalic Artery (BCA), Left Common Carotid Artery (LCCA), Left Subclavian Artery (LSA)) and the mid-lower thoracic Descending Aorta (DA). Pressure measurements were recorded by the authors in [2] using applanation tonometry (SphygmorCor system). Flow measurements were recorded by the authors in [2] using gated PC-MRI with flash angiography.

Fig. 19 shows a representation of the network under study. The thoracic aortic network is modeled as a 0D network comprising $n_j = 3$ junctions, $n_v = 7$ elastic vessels and $n_s = 4$ vascular beds. Arteries in the thoracic network are denoted v_i with $i = 1, .., 7$ while terminal vascular beds are denoted v_i^b with $i = 8, \dots, 11$.

The parameters of the four 0D Windkessel models at the vascular beds are computed using the formulation in [2,50,51], where the resistance and compliance of the entire model network R_T and C_T respectively are calculated as

$$R_T = \frac{\bar{p}_5 - p_{out}}{\bar{q}_1}, \quad C_T R_T = \tau, \quad C_T = C_c + C_p, \tag{18}$$

with $\bar{p}_5$ the mean blood pressure of the ensemble averaged carotid waveform, τ is the diastolic pressure decay, C_T the compliance of the entire model network, C_p the total peripheral compliance and C_c the sum of the compliance of each 0D model segment. Vascular resistance and

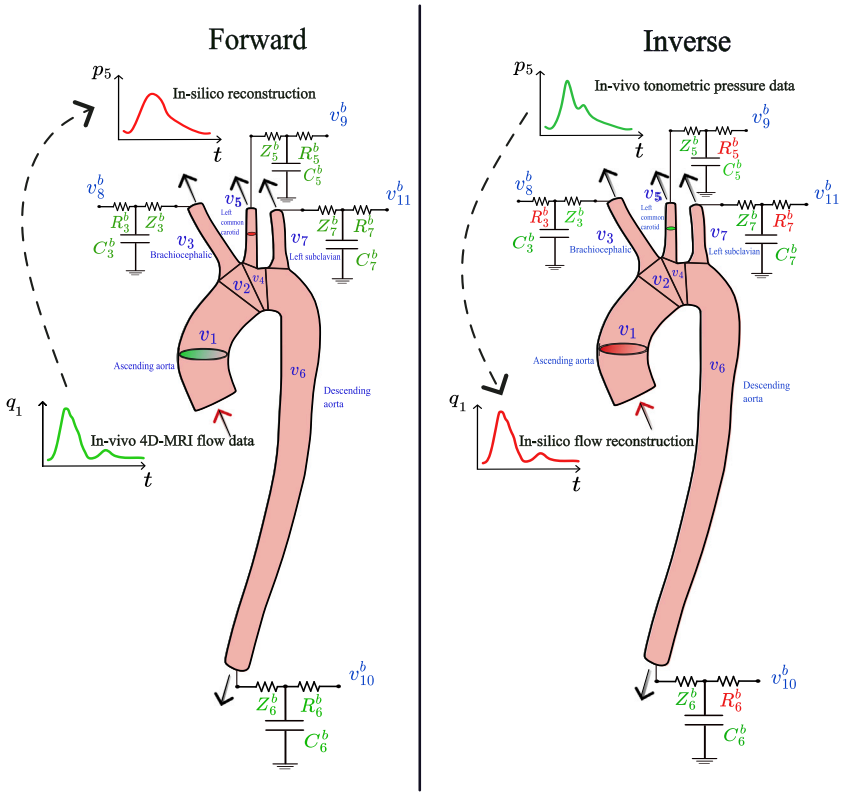


Fig. 19. Section 3.4. Test Case C. (Left) Representation of the thoracic aortic network, forward problem. (Right) Representation of the thoracic aortic network, inverse problem. Vessel labels are show in blue. Variables shown in red are unknown in the inverse problem and must be inferred, while known variables or signals are shown in green.

compliance at vascular bed s are calculated as:

$$R_s^b = R_T / f_s^b, \quad C_s^b = \frac{R_T}{R_s^b - Z_o^b} C_p, \quad f_s^b = \bar{q}_s^{b,out} / \bar{q}_s^{in}. \tag{19}$$

with $\bar{q}_s^{in}$ and $\bar{q}_s^{b,out}$ the aortic inlet average flow and the vascular bed outlet average flow, respectively.

In this figure, vessel labels are shown in blue. Parameters and signals shown in red are unknown and must be inferred, while those in green are known. A comparison between the known and unknown variables in the forward and inverse problems is depicted.

In the forward problem, the inlet flow rate wave at the aortic root as well as the vascular bed parameters are known, hence the 0D-PINN infers the rest of the waves in the rest of the vessels. In the inverse setting, the inlet flow rate wave at the aortic root and the Windkessel resistance parameters at the beds and flow fraction (R_s^b f_s^b) in each vascular bed, are unknown. In this setting, the stiffness coefficient k_v is known in all vessels, both in the forward and inverse problems. While in Test case B, the area waveform in time in all vessels was used to train the inverse PINNs, in this case, only the LCCA pressure waveform is used to train the algorithm. Notice that even though the most relevant variable to be reconstructed is the aortic root flow q_1, the flow and pressure waveforms in the rest of the vessels are also computed as a sub-product of the 0D-PINN computation.

3.4.1. Forward problem

The forward problem consists of predicting the flow and pressure waves in all the vessels using the inlet flow rate at the AA, assuming we also know the Windkessel parameters of the four vascular beds of the network. As demonstrated in previous test cases, the 0D-PINN's forward problem prediction offers valuable insights into its learning dynamics and provides a useful framework for hyperparameter tuning before addressing the inverse problem. The global loss function is the same as used in previous testcases.

In the study performed in [2], measurements are only available in the AA (v_1), (v_5) and 4 locations of the DA (v_6). Since we considered a single vessel to model the DA, we use an average of the flow signals at the different locations, to compare with the 0D-PINN reconstruction. In Fig. 20, the 0D-PINN predictions are compared with both the 0D-Picard predictions and in-vivo MRI measurements taken from the ascending and DA. Overall, the 0D-PINN and 0D-Picard predictions show strong agreement. However, notable discrepancies are observed when comparing with the in-vivo MRI data, particularly in the predicted pressure at the LCCA (v_5) and the flow rate at the DA (v_6). This is a consequence of the simplifications introduced by the 0D-model. Table 6 shows the errors of the 0D-PINN predictions with respect to the MRI flow data at the locations where flow or pressure was measured. It is observed that the error in flow in vessel v_1, the AA is not exactly null, despite the fact that it is a known signal, which is a general drawback of all neural networks. The prediction for pressure in the LCCA, as well as the prediction of the flow in the DA, are accurate and errors lie below 5%, showing that despite the various simplifications assumed in the 0D model, the 0D-PINN algorithm is able to reconstruct information that closely matches in-vivo data.

3.4.2. Inverse problem

The global loss function in this test case is modified by fitting the pressure waveform at the LCCA (v_5):

$$\mathcal{L}^I_{0D-PINN} = \mathcal{L}_{ODE} + \mathcal{L}_{bc} + \mathcal{L}_o + \mathcal{L}_{cycle} + \mathcal{L}_{p_5}, \quad \mathcal{L}_{p_5} = \sum_{i=1}^{N_p} l_p \left(p_5(t_i) - \tilde{p}_5(t_i)\right)^2. \tag{20}$$

with N_p being the number of samples in the carotid pressure signal. The set of trainable parameters in this case is $\boldsymbol{\Theta}^+ = \{\boldsymbol{w}, \boldsymbol{b}, R_3^b, R_5^b, R_6^b, R_7^b\}$.

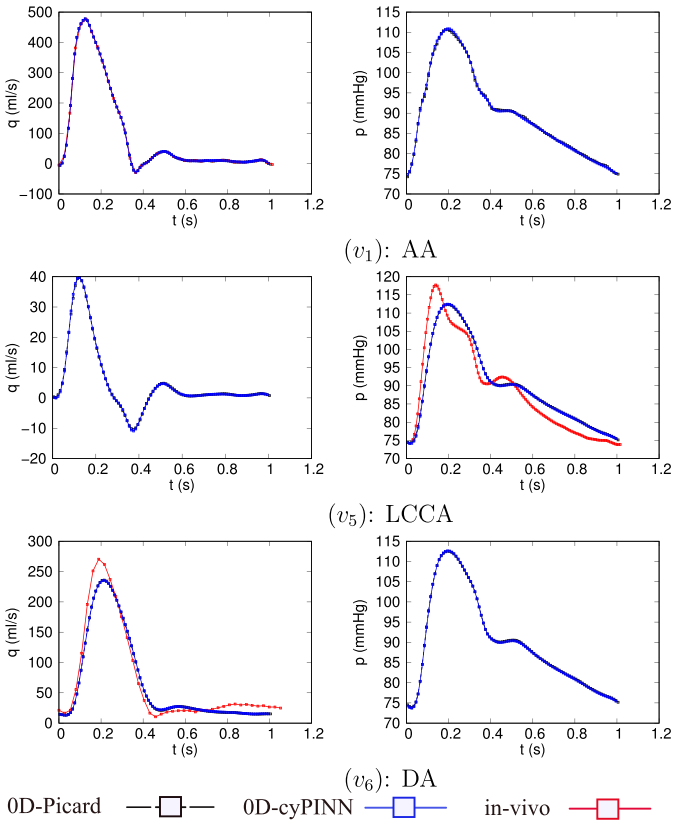


Fig. 20. Section 3.4.1. Test case C (forward). 0D-Picard vs 0D-PINN reconstructions for vessels (v_1), (v_5), and (v_6). In-vivo data are shown where available.

Table 6
Section 3.4.2. Test case C (inverse). Left: Forward error $\epsilon(\%)$ between 0D-PINN and in-vivo MRI measurements. Right: Inverse error ϵ between 0D-PINN trained with LCCA pressure and in-vivo MRI measurements. Windkessel resistance inference error e_R is also shown. (-) indicates that no in-vivo MRI measurement is available for the error computation.

Vessel	Forward		Inverse				
	$\epsilon(q)$ (%)	$\epsilon(p)$ (%)	$\epsilon(q)$ (%)	$\epsilon(p)$ (%)	$\epsilon(\bar{q})$ (%)	$\epsilon(\bar{p})$ (%)	e_R (%)
v_1	0.120	–	**9.83**	–	1.08	–	–
v_5	–	3.92	–	3.95	–	1.01	–
v_6	4.84	–	6.83	–	1.16	–	–
v_8^b	–	–	–	–	–	–	0.39
v_9^b	–	–	–	–	–	–	0.26
v_{10}^b	–	–	–	–	–	–	0.39
v_{11}^b	–	–	–	–	–	–	0.42

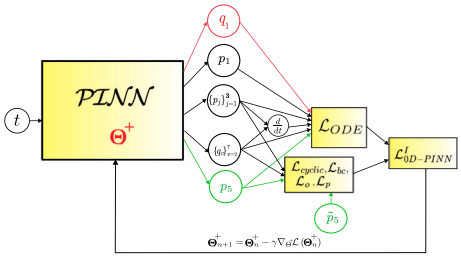


Fig. 21. Section 3.4.2. Test case C (inverse). Diagram of the inverse 0D-PINN training step for the thoracic aortic network, with input, output variables and different terms of the loss function. Reconstructed inlet flow waveform q_1 and Windkessel resistances $\{R_s^b\}_{s=3,5,6,7}$ are shown in red, while peripheral pressure wave data $\tilde{p}_5$ measured in the LLCA is shown in green.

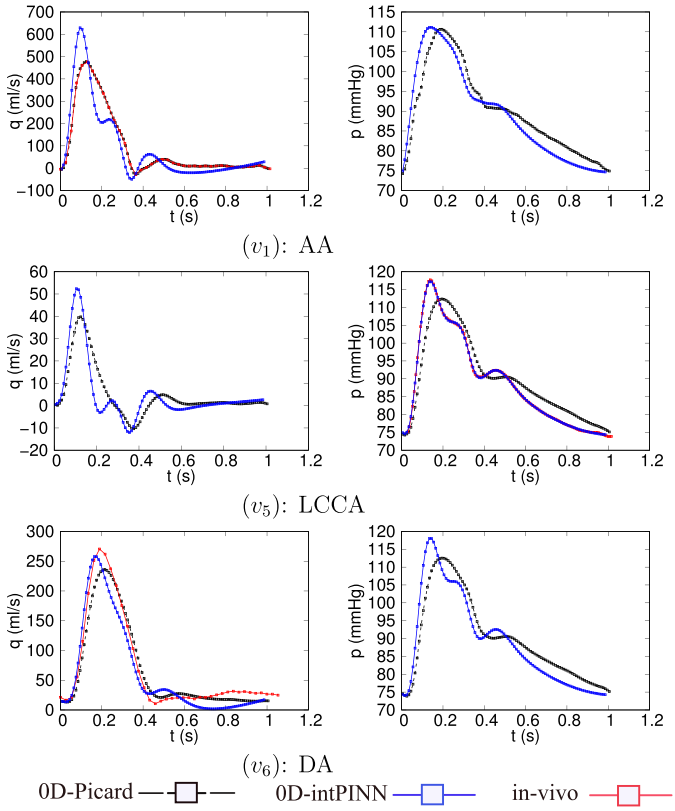


Fig. 22. Section 3.4.2. Test case C (inverse). 0D-Picard vs 0D-PINN reconstructions for vessels (v_1), (v_5), and (v_6). In-vivo data are shown where available.

Fig. 21 shows the 0D-PINN representation of the proposed inverse problem. Fig. 22 shows the 0D-intPINN reconstruction of the flow rate and pressure over time for the different vessels in the thoracic aortic network, trained with in-vivo data. There are discrepancies between the waves reconstructed by the 0D-PINN and the 0D-Picard solution, because the 0D-PINN was trained on real pressure data, whereas the 0D-Picard had only information about the inlet flow. When comparing the 0D-PINN reconstruction of flow and pressure signals with the in-vivo measurements (in red), we observe that the reconstructed flow waveform at the aortic inlet (v_1) exhibits a higher peak and noticeable oscillations. This behavior can be attributed to the shape of the pressure waveform at the carotid artery (v_5), which also contains such oscillatory features. Although these oscillations in pressure are physiological, the simplified nature of the 0D model leads to a less realistic flow signal compared to the measured one. This limitation stems from the mathematical model itself, rather than from the PINN approach. In contrast, at the DA (v_6), the 0D-PINN reconstruction shows good agreement with the in-vivo measurement.

Fig. 23 shows the convergence of $\{R_s^b\}_{s=3,5,6,7}$ with the iterations for two different initial values, (1) $\{R_s^b\}_{s=3,5,6,7} = 10\frac{kg}{cm^4\ s}$ and (2) $\{R_s^b\}_{s=3,5,6,7} = 5\frac{kg}{cm^4\ s}$. It is observed that the inferred resistances tend to their real values $\{\tilde{R}_s^b\}_{s=3,5,6,7}$. The inference of these parameters and the reconstruction of the flow rate and pressure waveforms at the the rest of the vessels of the network are coupled, that is, the optimization of waveforms and parameters is performed jointly, despite the fraction of flow leaving each vascular bed not being known in the inverse case.

It is particularly noteworthy that, despite the relatively significant reconstruction errors found in the prediction of the flow and pressure waveforms, the Windkessel resistances are inferred with high accuracy (errors below 2%). This accuracy in the prediction cannot be explained

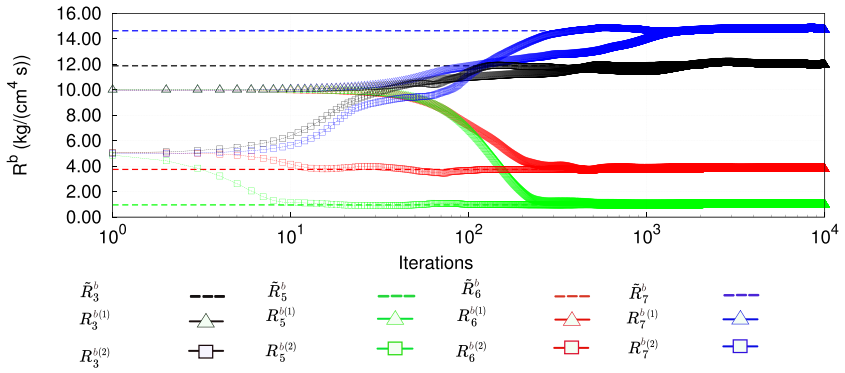


Fig. 23. Section 3.4.2. Test case C (inverse). Evolution of the Windkessel resistance across iterations for initializations I1 and I2.

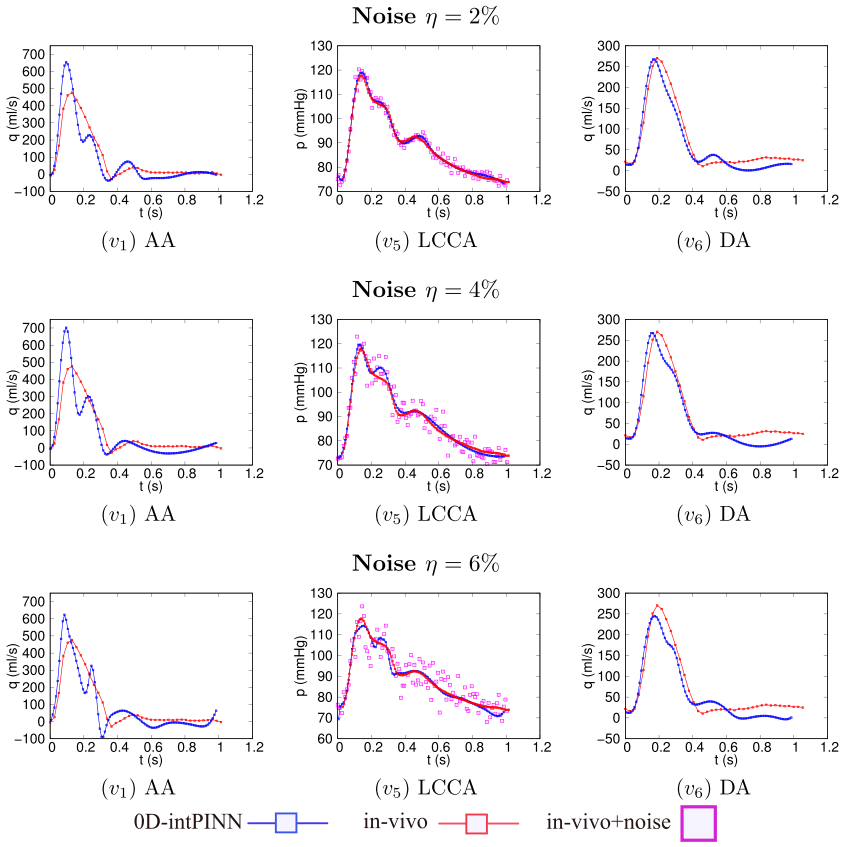


Fig. 24. Section 3.4.3. Test case C (inverse). 0D-PINN reconstruction for vessels v_1, v_5 and v_6, across Gaussian noise levels ($\eta = 2, 4$ and 6%) for $\tilde{p}_{\text{LCCA}}(t)$.

by the fact that the compliance of the peripheral beds, C_s^b, is given, as it involves the fraction of the flow through each terminal vessel, as shown in (19). Peripheral compliance modulates the pulse pressure, that is, the difference between systolic and diastolic pressure. The total resistance, is the parameter that regulates the mean pressure, which depends on the distribution of flow in the vascular beds. The advantage of this algorithm is that it is able to provide a coherent value of total resistance, inferring also an accurate value of average flow in the main aorta, by simply providing it with information on pressure in the LCCA.

Table 6 shows the errors of the 0D-intPINN predictions with respect to the in-vivo measurements for the different vessels. We observe close agreement between the measurements and the 0D-PINN trained with *in-vivo* pressure data. Errors $\epsilon(q)$ and $\epsilon(p)$ lie below 10%, while averaged errors $\epsilon(\bar{q})$ and $\epsilon(\bar{p})$ are below 3%. For the flow rate reconstruction at the aortic root, we obtain an error of 9.83%. For the vascular beds Windkessel resistance inference at the terminal vessels, we obtain an errors below 0.5%. The error in the inference of the total peripheral resistance $R_p = R_3^b + R_5^b + R_6^b + R_7^b$ is 0.34%.

3.4.3. *Influence of noise and uncertainty quantification in the reconstructed signals*

The quantification of the influence of noise in the algorithm becomes relevant since in most clinical applications noise in the measurements is present. We analyse how different levels of noise affect the solution of Test case C - inverse. Following the approach in [45], we perform an analysis of the uncertainty, which is divided into aleatoric and epistemic uncertainty, and compute an estimate of the total uncertainty in the reconstruction of the flow rate in the thoracic aorta.

Fig. 24 shows the 0D-PINN predictions without regularization for three different levels of Gaussian noise added to the measured carotid

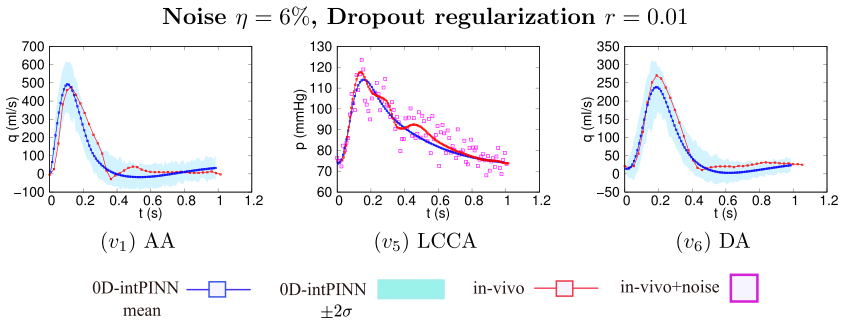


Fig. 25. Section 3.4.3. Test case C (inverse). 0D-PINN reconstruction for vessels v_1, v_5 and v_6 containing in-vivo data, using dropout regularization when noise is $\eta = 6\%$. Blue band represents $\pm 2\sigma$, with σ the standard deviation.

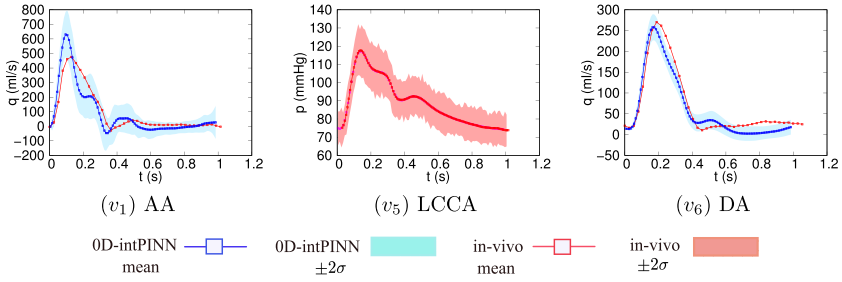


Fig. 26. Section 3.4.3. Test case C (inverse). 0D-PINN reconstruction for vessels v_1, v_5, and v_6 containing in-vivo data with $\eta = 6\%$ noise for $\tilde{p}_{\text{LCCA}}(t)$. Uncertainty boundaries for aleatoric error.

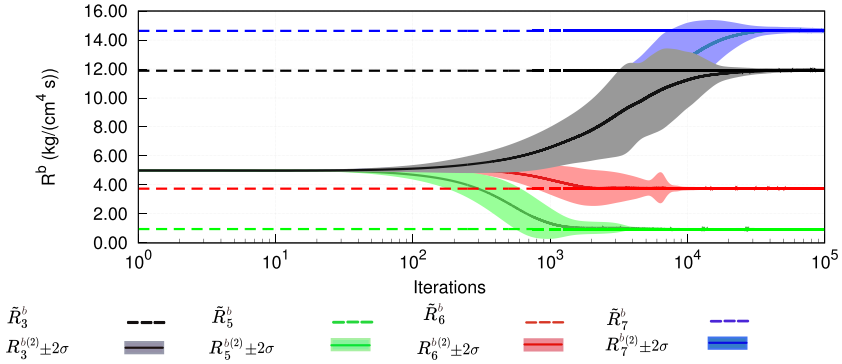
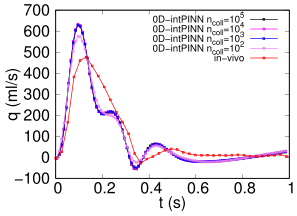
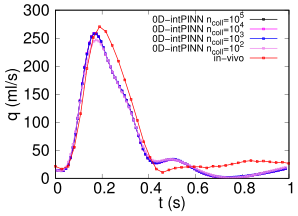


Fig. 27. Section 3.4.2. Test case C (inverse). Evolution of the Windkessel resistance and uncertainty boundaries under noisy measurements across iterations for initialization I2.

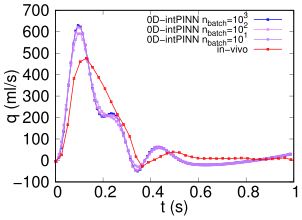


0D-cyPINN reconstructed flow in AA (v_1) for values of n_{coll}.

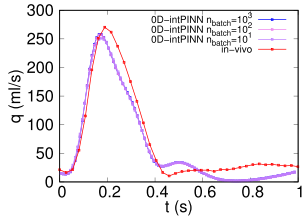


0D-cyPINN reconstructed flow in DA (v_6) for values of n_{coll}.

Fig. 28. Section 3.4.4. 0D-intPINN aortic flow reconstructions against in-vivo data, for vessels (v_1) and (v_5) across n_{coll}.



(v_1).



(v_6).

Fig. 29. Section 3.4.4. 0D-intPINN aortic AA flow reconstructions for vessels (v_1) (left) and (v_5) (right) against in-vivo data, across $n_{cobatchll}$.

pressure (2%, 4%, and 6%). The noisy signal $\tilde{p}_{LCCA}(t)$ was generated by applying multiplicative Gaussian noise to the reference data $p_{LCCA}(t)$ as

$$\tilde{p}_{LCCA}(t) = p_{LCCA}(t)\left(1 + \frac{\eta}{100}\xi(t)\right), \tag{21}$$

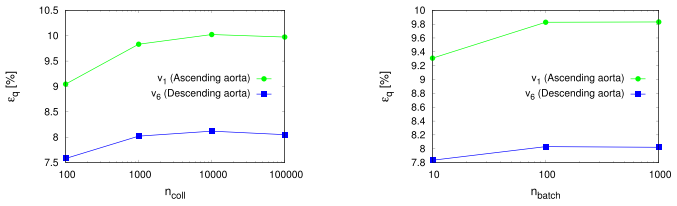
where η denotes the relative noise amplitude as a percentage (%) and $\xi(t) \sim \mathcal{N}(0,1)$ represents a standard normal random variable. In Fig. 24 we observe that as the measurement noise increases, the shape of the reconstructed flow waveform in the aorta deteriorates, while its magnitude remains largely unaffected. The difference in magnitude with respect to the in-vivo signal arises from the inherent approximation of the 0D model, as explained in Section 3.4.2.

Additionally, we apply dropout regularization to the data loss [52]. Regularization is expected to produce a solution that is less prone to overfitting the noise and therefore closer to the true mean solution. In this case, we regularize $\mathcal{L}_D$, which is associated with fitting the noisy LCCA signal, thereby mitigating overfitting.

The results after regularization for a noise level of 6% are shown in Fig. 25. Reconstructed flow in v_1 and v_6 is plotted together with uncertainty boundaries corresponding to two standard deviations. These uncertainty boundaries were obtained by repeatedly performing Monte Carlo (MC) Dropout tests with dropout rate $r = 0.01$ [52], in which the model is evaluated multiple times with dropout layers kept active during inference. The dropout rate r may seem very low, but since the neural network has 10 layers with 10 neurons each — a total of 1219 parameters — even small values of r correspond to pruning quite a large number of neurons. This stochastic sampling of the network allows the estimation of predictive uncertainty: the mean prediction represents the expected reconstructed flow, while the variance across the different realizations of the noise reflects model uncertainty. Fig. 25 shows that the reconstructed flow with uncertainty boundaries corresponding to two standard deviations ($\pm 2\sigma$) in v_1 and v_6 (left and right) for a noise level of $\eta = 6\%$ in $\tilde{p}_{LCCA}(t)$ (middle). These uncertainty boundaries, obtained with an MC Dropout with $N = 200$ repetitions, represent an estimate of the epistemic uncertainty, that is, the variability in the model's predictions due to uncertainty in its learned parameters.

When fitting the signal contaminated with $\eta = 6\%$, the regularization flattens the highest peak occurring at approximately $t = 0.17$ s, as the algorithm adjusts a smoother, less noisy signal to the noisy data points. This modified shape is then propagated downstream to the AA, resulting in a smaller peak in the reconstructed AA flow (Fig. 25, left). Since regularization reduces the amplitude of the highest peak in $\tilde{p}_{\mathrm{LCCA}}(t)$ (Fig. 25, middle), it produces an AA flow reconstruction that is much closer to the real flow peak (Fig. 25, left). On the other hand, the smoothing effect of regularization prevents the capture of the secondary wave observed in the in-vivo $\tilde{p}_{\mathrm{LCCA}}(t)$ signal. Consequently, this secondary wave is also absent in the AA flow reconstruction. Uncertainty boundaries (as shown in Fig. 25) reach a maximum standard deviation of $\sigma = \pm 77.5$ ml for the flow in AA and $\sigma = \pm 49.5$ ml for the flow in DA.

To assess the aleatory uncertainty of the reconstructed flow, we add Gaussian noise of 6% to the measured pressure signal, simulating realistic measurement noise. We then repeatedly evaluate and train the model through multiple Monte Carlo (MC) iterations ($N = 100$) to obtain the resulting distribution of the reconstructed flow at the aortic root. Fig. 26 (middle) presents the LCCA pressure together with its noise-induced uncertainty boundaries, whereas Fig. 26 (left and right) illustrates the reconstructed flow signals in the AA and DA, along with their respective uncertainty limits of two standard deviations. The uncertainty boundaries for the reconstruction of the vascular Windkessel bed resistances R^b are shown in Fig. 27. We observe that despite the noise, Windkessel parameters are also accurately inferred after 10^4 epochs. Uncertainty boundaries (as shown in Fig. 26) reach a maximum standard deviation of $\sigma = \pm 87$ ml for the flow in AA and $\sigma = \pm 16$ ml for the flow in DA.



Evolution of ϵ_q with n_{coll}. Evolution of ϵ_q with n_{batch}.

Fig. 30. Section 3.4.4. Relative percentage error for the 0D-intPINN aortic flow reconstruction against in-vivo data, for vessels (v_1) and (v_5) across n_{coll} and. n_{batch}. Error definitions for ϵ_q and ϵ_p can be found in Appendix A.

Uncertainty boundaries (as shown in Fig. 27) reach a final standard deviation (at the end of the training process) of $\sigma = \pm 0.0087\ \frac{kg}{cm^4\ s}$ for $R_3^{b(2)}$, $\sigma = \pm 0.094\ \frac{kg}{cm^4\ s}$ for $R_5^{b(2)}$, $\sigma = \pm 0.048\ \frac{kg}{cm^4\ s}$ for $R_6^{b(2)}$ and $\sigma = \pm 0.0048\ \frac{kg}{cm^4\ s}$ for $R_7^{b(2)}$.

3.4.4. Convergence tests

To evaluate the algorithm under different hyperparameter configurations and study its convergence behavior, we varied the two relevant hyperparameters influencing network convergence during training: the number of collocation points (n_{coll}) and the batch size (n_{batch}). To showcase the convergence, we use Test case C - inverse, where the flow at the AA is reconstructed.

Convergence with the number of collocation points n_{coll}. Fig. 28 shows the 0D-intPINN flow reconstruction for four values of $n_{coll} = \{10^2, 10^3, 10^4, 10^5\}$ in vessels v_1 and v_6, where in-vivo data is available.

Convergence with collocation minibatch size n_{batch}. Fig. 29 shows the flow reconstruction for four values of $n_{batch} = \{10^1, 10^2, 10^3, 10^4\}$ in vessels v_1 and v_6, where in-vivo data is available.

Fig. 30 (left) illustrates the evolution of the error with respect to n_{coll}, while Fig. 30 (right) shows its variation with n_{batch}. We observe that the error increases as n_{coll} decreases, particularly in the AA (v_1), whereas it remains nearly constant in the DA (v_6), except for very low values of $n_{coll} = 10$. This behavior is consistent with the fact that reducing n_{coll} lowers the spatial and temporal resolution of the PINN across the computational domain, thereby decreasing the accuracy of the reconstruction. On the other hand, a slight increase in the flow reconstruction error relative to the in-vivo aortic flow is observed when decreasing n_{batch}, especially for the AA (v_1), while for the DA (v_6) the error remains largely unaffected, again except for very small n_{batch} values. Relative percentage error definitions for ϵ_q and ϵ_p can be found in Appendix A.

4. Conclusions and future work

In this work, a novel application of Physics-Informed Neural Networks (PINNs) is presented, combining 0D modeling of networks of elastic vessels (arteries) with Physics-Informed Neural Networks (PINNs). The resulting algorithm has been coined 0D-PINNs. Its novelty resides in its simplicity to incorporate a system of arbitrarily many coupled ODE equations describing the highly nonlinear dynamics of transient flow in networks of elastic vessels connected with arterial junctions and vascular beds, for both forward and inverse problems. Additional constrains including the wall constitutive model for the fluid-structure interaction as well as the periodicity across different cardiac cycles (suppressing the need for data on initial conditions) are embedded in the 0D-PINN structure, in order to account for the basic principles of pulse wave propagation.

A thorough analysis of the different design choices involved in the construction of the 0D-PINN is performed, accounting for the physical relevance of the predicted variables and the choice of hyperparameters.

First, a simplified illustrative example with one isolated junction of an elastic vessel is presented, in order to showcase the construction and

functioning of the 0D-PINNs, using the vanilla formulation with some additional constraints regarding the fluid-wall interaction model and the periodicity across different cardiac cycles. Results for the prediction of the different waves and the inference of the lumped resistances show a high level of accuracy (ϵ <4%), when compared with the 0D-Picard time integration numerical scheme. It is also shown that enforcing periodicity yields a similar level of accuracy compared to feeding initial conditions at the beginning of the wave cycle.

Next, this methodology is extended to an experimental whole-body mock arterial circulatory system built in [1] in the laboratory. Available data and experimental results in [1] are used to train an extended 0D-PINN algorithm, in both forward and inverse settings, demonstrating that the mechanical properties of the mock silicone arteries, in particular the silicone Young modulus, can be accurately inferred (ϵ <6%) by feeding the algorithm exclusively with luminal area data over time at the different vessels of the network. As a byproduct, flow and pressure waveforms are jointly reconstructed along with the Young modulus. Results show potential for the reconstruction of waves and hidden parameters in whole body arterial networks.

Finally, a clinical scenario concerning the prediction of non-measurable signals and parameters inside the human thoracic cavity has been presented, including the reconstruction of the flow and pressure waves in the ascending aorta, aortic arch, descending aorta and upper cerebral branches. 0D-PINNs have demonstrated high accuracy (ϵ <5%) in reproducing in-silico 0D-Picard time integration numerical predictions. For the inverse setting, 0D-PINNs have been able to not only reconstruct intrathoracic aortic root flow rate from non-invasive LCCA in-vivo pressure data, but also additional parameters without any prior assumptions about any signal other than the measured LCCA pressure. This has been highlighted as one of the most significant advantages of 0D-PINNs compared with traditional numerical solvers, which are usually unable to tackle complex inverse problems in such a straightforward fashion. The clinical relevance of the latter study resides in its potential application to real clinical scenarios such as surgical procedures or interventions where the reconstruction and monitoring of the ventricular outflow signal are relevant for the stability of the patient's cardiovascular system.

Regarding the proposed inverse problems in the healthy human aortic network, forthcoming work will refine the 0D-PINN formulation so that it captures additional hemodynamic effects—especially wave reflections arising downstream of the terminal vessels of the thoracic aorta—and may also transition to a one-dimensional PINN (1D-PINN), in order to improve the reconstruction accuracy of the aortic root flow waveform from real clinical data. A further priority is to calibrate the distal vascular bed compliance C_s^b; achieving this will require supplementary measurements of downstream flow fractions or the incorporation of additional physical constraints in the loss function, related to the exponential decay of the pressure waveform, because sensitivity analysis reveals that the model is markedly more responsive to variations in compliance than to other parameters.

While 3D and 1D models provide high resolution, 0D models offer the competitive advantage of extremely low computational cost. When combined with inverse PINN modeling techniques, they form a powerful and efficient computational tool. It is also remarkable that the 0D model, when used in combination with PINNs, retains the fundamental features necessary to recover parameters or waveforms from experimental in-vitro or in-vivo data, despite its simplifications.

CRediT authorship contribution statement

J. Orera: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Formal analysis, Data curation, Conceptualization. **J. Mairal:** Writing – review & editing, Writing – original draft, Validation, Methodology, Formal analysis, Conceptualization. **L. Sánchez-Fuster:** – Writing – review & editing, Validation, Software, Methodology, Conceptualization. **J. Murillo:** Writing – review & editing, Writing – original draft, Validation, Supervision, Resources, Project administration, Methodology, Funding acquisition, Formal analysis, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work was supported by project PID2023-150074NB-I00 funded by the MICIU/AEI/10.13039/501100011033/ and FEDER, UE.

Appendix A. Error metrics

We define the error metric ϵ_p for a predicted pressure signal $p(t)$ with respect to a *ground truth* reference data signal $\tilde{p}(t)$ as a time-averaged root mean square difference, normalized by the reference:

$$\epsilon_p = \frac{1}{N}\sum_i^N \sqrt{\left(\frac{p(t_i)-\tilde{p}(t_i)}{\tilde{p}(t_i)}\right)^2}, \tag{A.1}$$

with N being the number of samples in the cardiac cycle ground truth data. For the flow rate, we define the error metric ϵ_q for a predicted flow rate signal $q(t)$ with respect to a *ground truth* reference data signal $\tilde{q}(t)$ as a time-averaged root mean square difference:

$$\epsilon_q = \frac{1}{N}\sum_i^N \sqrt{\left(\frac{q(t_i)-\tilde{q}(t_i)}{\max_i \tilde{q}(t_i)}\right)^2}, \tag{A.2}$$

is normalized by the maximum value of the reference to avoid division by values close to zero. In addition, we define the error in the mean of the signal, denoted $\epsilon(\bar{p})$ and $\epsilon(\bar{q})$, as

$$\epsilon(\bar{p}) = \frac{\sqrt{(\bar{p}-\bar{\tilde{p}})^2}}{\bar{\tilde{p}}}, \quad \epsilon(\bar{q}) = \frac{\sqrt{(\bar{q}-\bar{\tilde{q}})^2}}{\bar{\tilde{q}}} \tag{A.3}$$

where the temporal mean is computed as:

$$\bar{p} = \frac{1}{N}\sum_{t=1}^N p(t). \tag{A.4}$$

To quantify the error in the inferred value of the resistance parameter R_v^b, we define

$$e_R = \frac{|R^b - \tilde{R}^b|}{\tilde{R}^b}. \tag{A.5}$$

The error e_E is defined as the relative difference between the predicted modulus E and the Young modulus of a material E_{wall}, computed as:

$$e_E = \frac{|E - E_{wall}|}{E_{wall}}. \tag{A.6}$$

Appendix B. Parameters and hyperparameters for the proposed test cases and computational efficiency of 0D-PINNs against 0D-picard

Parameters for Test cases A and C are shown in Table B.7. Parameters for Test case B can be extracted from Table 2 in [1]. Hyperparameters for Test cases A, B and C are shown in Table B.8. n_e denotes the total number of epochs used to train the models in the different test cases. The learning rate ℓ, determines the size of the steps the model takes during optimization. The size of the minibatches, n_{batch}, indicates the number

of samples used in each training iteration. The number of neurons per layer, m, defines the complexity of each layer in the neural network. The number of layers, l, indicates how *deep* the neural network is.

Table B.9 presents the computational times of the 0D-PINN and 0D-Picard algorithms for the different test cases, under both forward and inverse configurations. The 0D-Picard method outperforms the 0D-PINN by approximately three orders of magnitude in the forward problem.

Table B.7
Parameter values used in Test Cases A and C.

Parameters	Test Case A	Test Case C
Cardiac cycle (s)		
t_o	0.0	0.0
t_f	0.8264	1.0150
Cross-sectional radius (cm)		
r_1	1.0	1.2675
r_2	1.0	1.205
r_3	1.0	0.6767
r_4	–	1.1267
r_5	–	0.555
r_6	–	0.92
r_7	–	0.61
Vessel length L_i (cm)		
L_1	5	6.58
L_2	5	0.57
L_3	5	4.2
L_4	–	1.33
L_5	–	3.23
L_6	–	25.08
L_7	–	3.31
Resistance R^b (kg/(cm $\cdot s^4$))		
R_1^b	0	0
R_2^b	0.2266554	0
R_3^b	0.24024304	3.74289
R_4^b	–	0
R_5^b	–	14.63059
R_6^b	–	0.95852
R_7^b	–	11.88281
Outlet pressure p_{out} (kg/(cm $\cdot s^2$))		
p_{out_1}	0	0
p_{out_2}	3.2	0
p_{out_3}	3.2	43.89
p_{out_4}	–	0
p_{out_5}	–	43.89
p_{out_6}	–	43.89
p_{out_7}	–	43.89
Pulse Wave Velocity c_0 (cm/s)		
$c_{o,1}$	800	456
$c_{o,2}$	800	456
$c_{o,3}$	800	456
$c_{o,4}$	–	456
$c_{o,5}$	–	456
$c_{o,6}$	–	456
$c_{o,7}$	–	456

Table B.8
List of key hyperparameters for Test cases A, B and C.

Hyperparameter	Test case A	Test case B	Test case C
l_1 - l_{11} (forward and inverse)	1	–	–
l_1 - l_{118} (forward)	–	1	–
l_1 - l_{119} (inverse)	–	1	–
l_{119} - l_{155} (inverse)	–	10^2	–
l_1 - l_{38} (forward and inverse)	–	–	1
n_{epochs}	200,000	300,000	200,000
n_{coll}	1000	1000	1000
Batch size	200	200	200
Number of neurons (m)	10	50	10
Number of layers (l)	10	10	10
Activation function	*tanh*	*tanh*	*tanh*
Learning rate (α)	0.001	0.001	0.001

Table B.9
Compute-time comparison for 0D-PINN (forward & inverse; training and solving) and 0D-Picard (forward solving) across Test cases A, B, and C.

Method / Test-case (time in minutes)	Forward			Inverse		
	A	B	C	A	B	C
0D-PINN (train)	29.17	1375	420	450	2158	535
0D-Picard (solve)	1.37	4.17	1.33	–	–	–

In this study, we report the performance of the vanilla 0D-PINN (plain implementation), as our focus lies primarily on the methodological framework rather than computational optimization. Future work will aim to extend 0D-PINNs to real-time applications involving the inverse reconstruction problems proposed here, which will require a more detailed exploration of advanced training acceleration techniques [27]. All machine learning models in this work were trained using a single NVIDIA GeForce RTX 3090 GPU.

Hyperparameters for test cases A, B and C are shown in Table B.8.

References

[1] 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 (2007) 3476–3486.

[2] J. Alastruey, N. Xiao, H. Fok, T. Schaeffter, C.A. Figueroa, On the impact of modelling assumptions in multi-scale, subject-specific models of aortic haemodynamics, J. R. Soc. Interface 13 (2016) 20160073.

[3] J. Murillo, P. García-Navarro, A roe type energy balanced solver for 1d arterial blood flow and transport, Comput. Fluids 117 (2015) 149–167.

[4] J. Willard, X. Jia, S. Xu, M. Steinbach, V. Kumar, Integrating physics-based modeling with machine learning: A survey, arXiv preprint arXiv:2003.04919, 2020 1 1–34.

[5] G.E. Karniadakis, I.G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, L. Yang, Physics-informed machine learning, Nat. Rev. Phys. 3 (2021) 422–440.

[6] A. Quarteroni, P. Gervasio, F. Regazzoni, Combining physics-based and data-driven models: advancing the frontiers of research with Scientific Machine Learning, arXiv preprint arXiv:2501.18708, 2025.

[7] P. Segers, J. Kips, B. Trachet, A. Swillens, S. Vermeersch, D. Mahieu, E. Rietzschel, M. De Buyzere, L. Van Bortel, Limitations and pitfalls of non-invasive measurement of arterial pressure wave reflections and pulse wave velocity, Art. Res. 3 (2009) 79–88.

[8] G. Bertaglia, A. Navas-Montilla, A. Valiani, M.I.M. García, J. Murillo, V. Caleffi, Computational hemodynamics in arteries with the one-dimensional augmented fluid-structure interaction system: viscoelastic parameters estimation and comparison with in-vivo data, J. Biomech. 100 (2020) 109595.

[9] J. Murillo, P. García-Navarro, A solution of the junction riemann problem for 1d hyperbolic balance laws in networks including supersonic flow conditions on elastic collapsible tubes, Symmetry 13 (2021).

[10] J. Murillo, P. García-Navarro, Numerical coupling of 0d and 1d models in networks of vessels including transonic flow conditions. Application to short-term transient and stationary hemodynamic simulation of postural changes, Int. J. Numer. Methods Biomed. Eng. 39 (2023) e3751.

[11] A. Sen, E. Ghajar-Rahimi, M. Aguirre, L. Navarro, C.J. Goergen, S. Avril, Physics-Informed graph neural networks to solve 1-d equations of blood flow, Comput. Methods Programs Biomed. 257 (2024) 108427.

[12] J. Orera, J. Ramírez, P. Garcia-Navarro, J. Murillo, RoePINNs: an integration of advanced CFD solvers with Physics-Informed neural networks and application in arterial flow modeling, Comput. Methods Appl. Mech. Eng. 440 (2025) 117933.

[13] J. Orera, P. Garcıa-Navarro, J. Murillo, J. Ramırez, Combined estimation of Cross-Sectional area, flow rate and pulse wave velocity using Physics-Informed neural networks with 1d hemodynamic model data, in: Computing in Cardiology Conference, 2024.

[14] S.L. Brunton, J.L. Proctor, J.N. Kutz, Discovering governing equations from data by sparse identification of nonlinear dynamical systems, Proc. Natl. Acad. Sci. 113 (2016) 3932–3937.

[15] S.H. Rudy, S.L. Brunton, J.L. Proctor, J.N. Kutz, Data-driven discovery of partial differential equations, Sci. Adv. 3 (2017) e1602614.

[16] J.N. Kutz, Machine learning for parameter estimation, Proc. Natl. Acad. Sci. 120 (2023) e2300990120.

[17] E. Burman, M.G. Larson, K. Larsson, C. Lundholm, Stabilizing and Solving Inverse Problems using Data and Machine Learning, arXiv preprint arXiv:2412.04409, 2024.

[18] D. Kochkov, J.A. Smith, A. Alieva, Q. Wang, M.P. Brenner, S. Hoyer, Machine learning–accelerated computational fluid dynamics, Proc. Natl. Acad. Sci. 118 (2021) e2101784118.

[19] P. Solán-Fustero, J.L. Gracia, A. Navas-Montilla, P. García-Navarro, Combination of intrusive POD-based reduced-order models and augmented riemann solvers applied

to unsteady 2d shallow water equations, Comput. Methods Appl. Mech. Eng. 436 (2025) 117702.
[20] F. Regazzoni, S. Pagani, A. Cosenza, A. Lombardi, A. Quarteroni, A physics-informed multi-fidelity approach for the estimation of differential equations parameters in low-data or large-noise regimes, Rendiconti Lincei 32 (2021) 437–470.
[21] S. Cai, Z. Mao, Z. Wang, M. Yin, G.E. Karniadakis, Physics-informed neural networks (PINNs) for fluid mechanics: a review, Acta Mech. Sin. 37 (2021) 1727–1738.
[22] H. Eivazi, M. Tahani, P. Schlatter, R. Vinuesa, Physics-informed neural networks for solving reynolds-averaged navier–stokes equations, Phys. Fluids 34 (2022).
[23] I.E. Lagaris, A. Likas, D.I. Fotiadis, Artificial neural networks for solving ordinary and partial differential equations, IEEE Trans. Neural Netw. 9 (1998) 987–1000.
[24] I.E. Lagaris, A.C. Likas, D.G. Papageorgiou, Neural-network methods for boundary value problems with irregular boundaries, IEEE Trans. Neural Netw. 11 (2000) 1041–1049.
[25] M. Raissi, P. Perdikaris, G.E. 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.
[26] F.M. Rohrhofer, S. Posch, B.C. Geiger, On the pareto front of physics-informed neural networks, arXiv preprint arXiv:2105.00862, 2021.
[27] S. Wang, S. Sankaran, H. Wang, P. Perdikaris, An expert's guide to training physics-informed neural networks, arXiv preprint arXiv:2308.08468, 2023.
[28] M. Fedele, R. Piersanti, F. Regazzoni, M. Salvador, P.C. Africa, M. Bucelli, A. Zingaro, A. Quarteroni, et al., A comprehensive and biophysically detailed computational model of the whole human heart electromechanics, Comput. Methods Appl. Mech. Eng. 410 (2023) 115983.
[29] Y. Shi, P. Lawford, R. Hose, Review of zero-d and 1-d models of blood flow in the cardiovascular system, Biomed. Eng. Online 10 (2011) 1–38.
[30] S. Epstein, M. Willemet, P.J. Chowienczyk, J. Alastruey, Reducing the number of parameters in 1d arterial blood flow modeling: less is more for patient-specific simulations, Am. J. Physiol.-Heart Circ. Physiol. 309 (2015) H222–H234.
[31] A.S. Khan, A. Shahzad, M. Zubair, A. Alvi, R. Gul, Personalized 0d models of normal and stenosed carotid arteries, Comput. Methods Programs Biomed. 200 (2021) 105888.
[32] A. Boccadifuoco, A. Mariotti, S. Celi, N. Martini, M.V. Salvetti, Impact of uncertainties in outflow boundary conditions on the predictions of hemodynamic simulations of ascending thoracic aortic aneurysms, Comput. Fluids 165 (2018) 96–115.
[33] R. Torii, M. Oshima, T. Kobayashi, K. Takagi, T.E. Tezduyar, Influence of wall elasticity in patient-specific hemodynamic simulations, Comput. Fluids 36 (2007) 160–168. Challenges and Advances in Flow Simulation and Modeling.
[34] Z. Li, W. Mao, A fast approach to estimating windkessel model parameters for patient-specific multi-scale CFD simulations of aortic flow, Comput. Fluids 259 (2023) 105894.
[35] J. Richter, J. Nitzler, L. Pegolotti, K. Menon, J. Biehler, W.A. Wall, D.E. Schiavazzi, A.L. Marsden, M.R. Pfaller, Bayesian Windkessel calibration using optimized 0D surrogate models, arXiv preprint arXiv:2404.14187, 2024.
[36] P. Heger, D. Hilger, M. Full, N. Hosters, Investigation of physics-informed deep learning for the prediction of parametric, three-dimensional flow based on boundary data, Comput. Fluids 278 (2024) 106302.
[37] S. Wassing, S. Langer, P. Bekemeyer, Physics-informed neural networks for parametric compressible euler equations, Comput. Fluids 270 (2024) 106164.
[38] J. Garay, J. Dunstan, S. Uribe, F.S. Costabal, Physics-informed neural networks for parameter estimation in blood flow models, Comput. Biol. Med. 178 (2024) 108706.
[39] H. Csala, A. Mohan, D. Livescu, A. Arzani, Physics-constrained coupled neural differential equations for one dimensional blood flow modeling, Comput. Biol. Med. 186 (2025) 109644.
[40] G. Kissas, Y. Yang, E. Hwuang, W.R. Witschey, J.A. Detre, P. Perdikaris, Machine learning in cardiovascular flows modeling: predicting arterial blood pressure from non-invasive 4d flow MRI data using physics-informed neural networks, Comput. Methods Appl. Mech. Eng. 358 (2020) 112623.
[41] G.-B. Huang, Q.-Y. Zhu, C.-K. Siew, Extreme learning machine: theory and applications, Neurocomputing 70 (2006) 489–501.
[42] L. Zhang, P.N. Suganthan, A survey of randomized algorithms for training neural networks, Inf. Sci. 364 (2016) 146–155.
[43] M. Tancik, P. Srinivasan, B. Mildenhall, S. Fridovich-Keil, N. Raghavan, U. Singhal, R. Ramamoorthi, J. Barron, R. Ng, Fourier features let networks learn high frequency functions in low dimensional domains, Adv. Neural Inf. Process. Syst. 33 (2020) 7537–7547.
[44] O. Sallam, M. Fürth, On the use of Fourier features-physics informed neural networks (ff-pinn) for forward and inverse fluid mechanics problems, Proc. Inst. Mech. Eng. Part M J. Eng. Marit. Environ. 237 (2023) 846–866.
[45] M. De Florio, Z. Zou, D.E. Schiavazzi, G.E. Karniadakis, Quantification of total uncertainty in the physics-informed reconstruction of cvsim-6 physiology, Philos. Trans. A 383 (2025) 20240221.
[46] M.S. Olufsen, C.S. Peskin, W.Y. Kim, E.M. Pedersen, A. Nadim, J. Larsen, Numerical simulation and experimental validation of blood flow in arteries with structured-tree outflow conditions, Ann. Biomed. Eng. 28 (2000) 1281–1299.
[47] S. Epstein, M. Willemet, P.J. Chowienczyk, J. Alastruey, Reducing the number of parameters in 1d arterial blood flow modeling: less is more for patient-specific simulations, Am. J. Physiol.-Heart Circ. Physiol. 309 (2015) H222–H234.
[48] O. Frank, The basic shape of the arterial pulse. First treatise: mathematical analysis, J. Mol. Cell. Cardiol. 22 (1990) 255–277.
[49] N. Westerhof, J.-W. Lankhaar, B.E. Westerhof, The arterial windkessel, Med. Biol. Eng. Comput. 47 (2009) 131–141.
[50] J. Alastruey, T. Passerini, L. Formaggia, J. Peiró, Physical determining factors of the arterial pulse waveform: theoretical analysis and calculation using the 1-d formulation, J. Eng. Math. 77 (2012) 19–37.
[51] J. Alastruey, K.H. Parker, J. Peiró, S.J. Sherwin, Lumped parameter outflow models for 1-d blood flow simulations: effect on pulse waves and parameter estimation, Commun. Comput. Phys. 4 (2008) 317–336.
[52] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, Dropout: a simple way to prevent neural networks from overfitting, J. Mach. Learn. Res. 15 (2014) 1929–1958.