MOX-Report No. 64/2026

Physics-Informed Neural Networks for the High-Resolution Reconstruction of Flow Measurement Indicators in Fluid Dynamics

Radisic, I.; Tirotta, R.; Zingaro, A.; Pagani, S.; Dede', L.

MOX, Dipartimento di Matematica
Politecnico di Milano, Via Bonardi 9 - 20133 Milano (Italy)

mox-dmat@polimi.it https://mox.polimi.it

Physics-Informed Neural Networks for the High-Resolution Reconstruction of Flow Measurement Indicators in Fluid Dynamics

Irena Radišić[a,*], Raffaele Tirotta[b], Alberto Zingaro[a,c], Stefano Pagani[a], Luca Dede'[a]

[a] *MOX, Dipartimento di Matematica, Politecnico di Milano, Piazza Leonardo Da Vinci 32, Milan, 20133, Italy*
[b] *Politecnico di Milano, Piazza Leonardo Da Vinci 32, Milan, 20133, Italy*
[c] *ELEM Biotech S.L., Via Laietana 26, Barcelona, 08003, Spain*

Abstract

Accurate, spatially resolved flow field measurements are essential for the reliable assessment of hemodynamic quantities in cardiovascular research and clinical practice. Experimental techniques, such as 4D flow MRI, PIV, or Doppler ultrasound, often yield data that are sparse, noisy, or under-resolved, particularly near vessel walls and in regions of complex flow. This limits the fidelity of distributed or derived hemodynamic indicators such as the wall shear stress and the clinical utility of such measurements. To address these challenges, we propose a physics-informed neural network (PINN) framework that integrates the incompressible Navier-Stokes equations with velocity measurements coming from experimental flow field data. By embedding physical laws into data, PINN enhances the reconstruction of velocity fields, enables the estimation of unmeasured quantities such as pressure and wall shear stress, and improves the spatial resolution of hemodynamic indicators. We show the effectiveness of our approach using both in silico and experimental data. First, we apply our method to the FDA nozzle benchmark, leveraging both control particle image velocimetry (PIV) measurements and computational fluid dynamics (CFD) simulations. Next, we apply our method to the more complex case of blood flow in an aneurysm model, exploiting in vitro 4D flow MRI data. In both cases, the synergy between data-driven learning and physics-based regularization yields results that align more closely with ground truth observations than standard CFD or pure data-driven approaches. Our findings highlight the potential of PINNs to improve the fidelity of under-resolved flow field measurements and yield spatially resolved hemodynamic indicators.

Keywords: PINNs, 4D flow MRI, Computational hemodynamics, Scientific machine learning

1. Introduction

4D flow MRI is a clinically established and non-invasive imaging technique for the in vivo velocity measurement of cardiac and vascular flows [1–3]. It consists of a 3D time-varying description of the three components of the blood velocity vector field based on phase-contrast MRI [4]. The reconstructed space-time description of the blood velocity provides hemodynamics biomarkers for diagnosis [5], including the wall shear stress, which is used to assess the risk of rupture in a blood vessel or to assist the diagnosis of important cardiac pathologies like stenosis or aneurysms [6, 7]. However, the accuracy in reconstructing biomarker indicators is constrained by the limited spatio-temporal resolution and signal-to-noise ratio [2]. Indeed, 4D flow MRI is not able to capture the small scales or flow-field features, such as recirculation regions or transitions to turbulence, as pointed out in [8, 9]. Moreover, data acquisition is particularly dispersed and uncertain on the boundary [10, 11], reducing the reliability in the estimation of vessel stresses and quantifying other spatially-resolved biomarkers [12]. Another source of uncertainty arises from the time resolution of the measurements. Using low time-resolution acquisition techniques leads to the loss of

*Corresponding author: irena.radisic@polimi.it

information on instantaneous events in the flow field, such as the sudden formation of vortex cores in short time scale, which would support the evaluation of the rupture risk of cerebral aneurysms [13, 14].

On the other hand, the use of computational fluid dynamics (CFD) and simulation-based approaches for patient specific geometries yields fluid flow information with arbitrary space and time resolution. Standard high fidelity methods, such as the finite element method (FEM) or the finite volume method (see, e.g., [15]) can be very time consuming [16]. Moreover, these methods often fail to account for the inherent variability of the data, and the personalization of these models requires multiple stages and is particularly laborious. Indeed, a typical simulation pipeline consists of image segmentation, meshing, discretization and parameter tuning, and in the case of model personalization some of these steps may even be iterated.

The goal of this work is to develop a new approach based on scientific machine learning [17–19] to enhance 4D flow MRI post-processing in a computationally efficient and accurate manner. Specifically, we aim to reconstruct high-resolution velocity fields from 4D flow MRI data by leveraging physical constraints in a physics-informed neural networks (PINNs) framework [20]. PINNs are an established technique for data-model integration in scenarios characterized by shortage of data [21] and, additionally, avoid geometry segmentation and mesh generation due to their mesh-free formulation. To compensate the limitations of 4D flow MRI, the training of the neural network leverage physical knowledge encoded into the Navier-Stokes equations, which are enforced in the loss function by computing a residual on a dataset of collocation points. This physical regularization is able to improve the reconstruction of the velocity field and its derived quantities, like the wall shear stress, while also allowing the estimation of quantities that are hardly measurable in-vivo, like the pressure distribution.

Several approaches have already been developed to recover the pressure field and the wall shear stress from 4D flow MRI velocity measurements with an appropriate level of accuracy, as well as other hemodynamics biomarkers, see e.g. [22]. Zhang et al. [23] introduced a weighted least-squares reconstruction method for the spatial integration of pressure gradients. The main advantage of this approach is that the pressure estimation is obtained non-invasively. However, the pressure gradients are recovered a posteriori from the velocity field, so the accuracy on the pressure field is strongly dependent on the accuracy of the velocity field. In [24], 4D flow MRI data and CFD simulations are used to train a convolutional neural network in order to assess hemodynamic quantities, such as wall shear stress. Other than reconstructing quantities not present in the original data, the inclusion of the physics not only augments sparse data, but also has beneficial effects on the regularization of noisy data. For example, in [25], the authors use a divergence-free constraint in order to effectively denoise data coming from 4D flow MRI measurements.

PINNs are becoming widely used in computational hemodynamics [26], particularly alongside domain segmentation and 1D reduced order modeling. Important contributions are the ones for the reconstruction of blood pressure given by Kissas et al. [27], and for the estimation of brain hemodynamics and arterial wall dilatation by Sarabian et al. [28]. Both of these works are based on 4D flow MRI data. Moreover, outside of the field of computational hemodynamics, PINNs have already succesfully been used for the reconstruction of latent flow field quantities from imaging data [29].

In this paper, we propose a computational pipeline for high-resolution velocity, wall-shear stress and pressure reconstruction based on PINN. To validate the approach we consider two test cases: the FDA nozzle benchmark case, known as Computational Round Robin n. 1 [30, 31], and a realistic patient-derived aneurysm experimental model. We provide a graphical abstract of our work in Figure **??**. In the first test case, we generate an in silico dataset by carrying out CFD simulations of the benchmark problem provided by the FDA. To mimic 4D MRI data, we consider the velocity samples on some selected cross-sections, and we match the CFD results on the remaining part of the dataset. The procedure is then repeated on an experimental dataset given by the three velocity components obtained from PIV [32, 33]. Next, we assess the flow field in an aneurysm. In this case, the data are given by open-access 4D flow MRI data for an in vitro patient-derived aneurysm model [11]. In particular, our method allows to compute the reconstructed velocity, pressure and wall shear stress wherever these quantities are not available.

This paper is organized as follows. In Section 2, we introduce the methodology and our approach based on PINNs. Section 3.1 is devoted to the FDA nozzle benchmark. In Section 3.2, we apply our methodology to an aneurysm model. We discuss the results of the two different test cases in Section 4, and the limitations of our approach in Section 4.1. Finally, in Section 5 we draw our conclusions and we discuss the results.

2. Materials & Methods

PINNs are a class of artificial neural networks (ANNs) that are trained to simultaneously leverage data and physical knowledge. The physical knowledge is typically incorporated by PDEs in the loss function [21]. In this work, we apply PINNs in the context of vascular hemodynamics, and since we assume that blood obeys the incompressible Navier-Stokes equations for a Newtonian fluid [34], the PDE loss term is accordingly specified.

Let $\Omega \subset \mathbb{R}^3$ be a bounded domain with boundary $\partial\Omega = \Gamma_{\text{inl}} \cup \Gamma_{\text{wall}} \cup \Gamma_{\text{out}}$, with $\Gamma_{\text{inl}}, \Gamma_{\text{wall}}, \Gamma_{\text{out}}$ disjoint subsets of $\partial\Omega$. We consider the velocity field $\boldsymbol{u}_{\text{ex}}$ and pressure field p_{ex},

$$\boldsymbol{u}_{\text{ex}} : \Omega \times [0,T] \to \mathbb{R}^3, \qquad p_{\text{ex}} : \Omega \times [0,T] \to \mathbb{R},$$

exact solutions of the dimensionless steady-state Navier-Stokes equations in Ω with suitable boundary conditions, that is, for every time $t \in [0,T]$ it holds that:

$$\begin{cases} (\boldsymbol{u}_{\text{ex}} \cdot \nabla)\boldsymbol{u}_{\text{ex}} + \nabla p_{\text{ex}} - \dfrac{1}{\mathbb{R}\text{e}} \Delta \boldsymbol{u}_{\text{ex}} = \boldsymbol{0}, & \text{in } \Omega, \quad (1a) \\ \nabla \cdot \boldsymbol{u}_{\text{ex}} = 0, & \text{in } \Omega, \quad (1b) \\ \boldsymbol{u}_{\text{ex}} = \boldsymbol{g}, & \text{on } \Gamma_{\text{inl}}, \quad (1c) \\ \boldsymbol{u}_{\text{ex}} = \boldsymbol{0}, & \text{on } \Gamma_{\text{wall}}, \quad (1d) \\ \left(\dfrac{1}{\mathbb{R}\text{e}} \nabla \boldsymbol{u}_{\text{ex}} - p_{\text{ex}} \boldsymbol{I} \right) \boldsymbol{n} = \boldsymbol{0}, & \text{on } \Gamma_{\text{out}}. \quad (1e) \end{cases}$$

Eq.s (1a) and (1b) are the momentum balance and mass conservation equations respectively, Eq. (1c) is the Dirichlet inflow boundary condition, Eq. (1d) is the Dirichlet no-slip boundary condition, whereas Eq. (1e) is the homogeneous Neumann outflow boundary condition. Given a volumetric flow rate Q, we define the Reynolds number $\mathbb{R}\text{e}$ over a characteristic section as:

$$\mathbb{R}\text{e} = \frac{\rho U D_t}{\mu}, \qquad U = \frac{4Q}{\pi D_{\text{t}}^2},$$

where ρ and μ are respectively the fluid density and dynamic viscosity, U is a characteristic velocity, and D_t is a characteristic diameter. We report the fluid flow properties parameters used throughout this work in Table 1.

	Unit	FDA	ANE-CFD	ANE-4DFLOW
Q	$\text{m}^3\,\text{s}^{-1}$	$5.2062 \cdot 10^{-6}$	$3.8 \cdot 10^{-6}$	$3.8 \cdot 10^{-6}$
D_t	m	0.004	0.0035	0.013
μ	Pa s	0.0035	0.00089	0.00089
ρ	$\text{kg}\,\text{m}^{-3}$	1056	997	997
$\mathbb{R}\text{e}$	-	500	1500	1500

Table 1: Fluid flow properties values for the FDA nozzle benchmark model (FDA) in Section 3.1 and for the aneurysm model (ANE) from Sections 3.2.1 and 3.2.2. The acronyms are further explained in Section 3.

While no time derivative of the velocity appears in Eq. (1a), the solutions $\boldsymbol{u}_{\text{ex}}$ and p_{ex} may still be time dependent due to the time dependence of the inflow boundary condition $\boldsymbol{g}$ in Eq. (1c). Moreover, since Eq. (1a) is steady-state, the model is assumed to hold for every $t \in [0,T]$.

In view of the construction of loss functions that reflect the physics expressed by system (1), we proceed by defining the Navier-Stokes residuals, for every $t \in [0,T]$, as in [21], for any $\boldsymbol{u} : \Omega \times [0,T] \to \mathbb{R}^3$ and $p : \Omega \times [0,T] \to \mathbb{R}$ sufficiently smooth:

$$f_{\text{C}}(\boldsymbol{u},p) = f_1(\boldsymbol{u},p) = \nabla \cdot \boldsymbol{u}, \qquad \text{in } \Omega, \quad (2a)$$

$$\boldsymbol{f}_{\text{M}}(\boldsymbol{u},p) = \begin{pmatrix} f_2(\boldsymbol{u},p) \\ f_3(\boldsymbol{u},p) \\ f_4(\boldsymbol{u},p) \end{pmatrix} = (\boldsymbol{u} \cdot \nabla)\boldsymbol{u} + \nabla p - \frac{1}{\mathbb{R}\text{e}} \Delta \boldsymbol{u}, \qquad \text{in } \Omega, \quad (2b)$$

where in Eq. (2a) is the mass conservation i.e. continuity equation residual f_{C}, and in Eq. (2b) is the momentum balance residual $\boldsymbol{f}_{\mathrm{M}}$.

Similarly to the residuals (2) associated to the PDE, we define the residuals associated to the boundary conditions (1c), (1d) and (1e) as:

$$\boldsymbol{f}_{\mathrm{BC},1}(\boldsymbol{u},p) = \boldsymbol{u} - \boldsymbol{g}, \qquad \text{on } \Gamma_{\mathrm{inl}}, \tag{3a}$$

$$\boldsymbol{f}_{\mathrm{BC},2}(\boldsymbol{u},p) = \boldsymbol{u}, \qquad \text{on } \Gamma_{\mathrm{wall}}, \tag{3b}$$

$$\boldsymbol{f}_{\mathrm{BC},3}(\boldsymbol{u},p) = \left(\frac{1}{\mathbb{R}\mathrm{e}}\nabla\boldsymbol{u}_{\mathrm{ex}} - p_{\mathrm{ex}}\boldsymbol{I}\right)\boldsymbol{n}, \qquad \text{on } \Gamma_{\mathrm{out}}. \tag{3c}$$

While the residual associated to the PDE can be evaluated in any point $(\boldsymbol{x},t)$ of the spatio-temporal domain $\Omega \times [0,T]$, the residuals associated with the boundary conditions are only defined on the corresponding subsets of the boundary of the spatio-temporal domain. Consequently, $\boldsymbol{f}_{\mathrm{BC},1}$ can only be evaluated in $\Gamma_{\mathrm{inl}} \times [0,T]$, $\boldsymbol{f}_{\mathrm{BC},2}$ in $\Gamma_{\mathrm{wall}} \times [0,T]$ and $\boldsymbol{f}_{\mathrm{BC},3}$ in $\Gamma_{\mathrm{out}} \times [0,T]$. The residuals are well defined for all $\boldsymbol{u}$ and p sufficiently smooth. Moreover, for $\boldsymbol{u}_{\mathrm{ex}}$ and p_{ex} all the residuals are identically zero since this pair satisfies the strong Navier-Stokes equations (1).

Once defined the residuals associated to the PDE (2) and the boundary conditions (3), we introduce the physics-based loss functions, a key characteristic of PINNs. We aim at finding ANNs $\boldsymbol{u}(\boldsymbol{\theta}) : \Omega \times [0,T] \to \mathbb{R}^3$ and $p(\boldsymbol{\theta}) : \Omega \times [0,T] \to \mathbb{R}$, where $\boldsymbol{\theta}$ are the parameters of the ANN, such that they are approximations of the exact solutions of system (1) $\boldsymbol{u}_{\mathrm{ex}}$, p_{ex}, that is such that

$$\boldsymbol{u}(\boldsymbol{x},t;\boldsymbol{\theta}) \simeq \boldsymbol{u}_{\mathrm{ex}}(\boldsymbol{x},t), \;\; p(\boldsymbol{x},t;\boldsymbol{\theta}) \simeq p_{\mathrm{ex}}(\boldsymbol{x},t), \;\; \forall (\boldsymbol{x},t) \in \Omega \times [0,T].$$

The parameters of the ANNs, corresponding to the vector $\boldsymbol{\theta}$, can be found by solving the following minimization problem:

$$\boldsymbol{\theta} = \arg\min_{\tilde{\boldsymbol{\theta}} \in \Theta} \mathcal{L}(\tilde{\boldsymbol{\theta}}), \tag{4}$$

where $\mathcal{L}(\tilde{\boldsymbol{\theta}})$ is the total loss function evaluated on a particular parametrization of the ANN given by $\tilde{\boldsymbol{\theta}}$, and Θ is the parameter space in which the minimization is performed.

The PINN method is based on defining the total loss function as:

$$\mathcal{L}(\tilde{\boldsymbol{\theta}}) = \mathcal{L}_{\mathrm{PDE}}(\tilde{\boldsymbol{\theta}}) + \mathcal{L}_{\mathrm{BC}}(\tilde{\boldsymbol{\theta}}) + \mathcal{L}_{\mathrm{train}}(\tilde{\boldsymbol{\theta}}), \tag{5}$$

where $\mathcal{L}_{\mathrm{PDE}}$ and $\mathcal{L}_{\mathrm{BC}}$ are the loss functions associated to the physics, and $\mathcal{L}_{\mathrm{train}}$ is a suitably defined data-driven contribution to the total loss function. In order to define the physics-based loss functions $\mathcal{L}_{\mathrm{PDE}}$ and $\mathcal{L}_{\mathrm{BC}}$, associated to the residuals (2) and (3), we introduce discrete (finite-dimensional) subsets of the spatio-temporal domain, made of the so-called collocation points, $\Gamma_{\mathrm{BC},1} \subset \Gamma_{\mathrm{inl}} \times [0,T]$ with $|\Gamma_{\mathrm{BC},1}| = N_{\mathrm{BC},1}$, $\Gamma_{\mathrm{BC},2} \subset \Gamma_{\mathrm{wall}} \times [0,T]$ with $|\Gamma_{\mathrm{BC},2}| = N_{\mathrm{BC},2}$, $\Gamma_{\mathrm{BC},3} \subset \Gamma_{\mathrm{out}} \times [0,T]$ with $|\Gamma_{\mathrm{BC},3}| = N_{\mathrm{BC},3}$, and $\Omega_{\mathrm{PDE}} \subset \Omega \times [0,T]$ with $|\Omega_{\mathrm{PDE}}| = N_{\mathrm{PDE}}$.

For any family of sufficiently smooth functions $\boldsymbol{u}(\boldsymbol{\theta})$ and $p(\boldsymbol{\theta})$ parametrized by $\boldsymbol{\theta}$, we define the loss function associated to the PDE evaluated at $\boldsymbol{\theta}$ as:

$$\mathcal{L}_{\mathrm{PDE}}(\boldsymbol{\theta}) = \sum_{i=1}^{N_{\mathrm{PDE}}} \sum_{j=1}^{4} \frac{\lambda_{\mathrm{PDE},j}}{N_{\mathrm{PDE}}} \left(|f_j(\boldsymbol{u}_i(\boldsymbol{\theta}), p_i(\boldsymbol{\theta}))|^2 + \lambda_{\mathrm{PDE,MSLE}} \log |f_j(\boldsymbol{u}_i(\boldsymbol{\theta}), p_i(\boldsymbol{\theta})) + 1|^2 \right), \tag{6}$$

where the subscript i indicates that the functions $\boldsymbol{u}(\boldsymbol{\theta})$ and $p(\boldsymbol{\theta})$ are evaluated at the i-th PDE collocation point $(\boldsymbol{x}_i, t_i) \in \Omega_{\mathrm{PDE}}$, that is $\boldsymbol{u}_i(\boldsymbol{\theta}) = \boldsymbol{u}(\boldsymbol{x}_i, t_i; \boldsymbol{\theta})$ and $p_i(\boldsymbol{\theta}) = p(\boldsymbol{x}_i, t_i; \boldsymbol{\theta})$. We remark that the relative weights $\{\lambda_{\mathrm{PDE},j}\}_{j=1}^{4}$ assigned to each residual term are in general different from each other and that the choice of the collocation points is a priori arbitrary. These aspects allow for a larger flexibility when defining the loss function (6) [35]. Our choice of adding the mean squared logarithmic error (MSLE) term in (6) in addition to the mean squared error (MSE) is meant to enable accurate flow approximation in regions

where the physics residuals have small magnitudes. Similarly, we define the loss function associated to the boundary conditions as:

$$\mathcal{L}_{\mathrm{BC}}(\boldsymbol{\theta}) = \sum_{j=1}^{3} \sum_{i=1}^{N_{\mathrm{BC},j}} \frac{\lambda_{\mathrm{BC},j}}{N_{\mathrm{BC},j}} \left| \boldsymbol{f}_{\mathrm{BC},j}(\boldsymbol{u}_{j,i}(\boldsymbol{\theta}), p_{j,i}(\boldsymbol{\theta})) \right|^2, \tag{7}$$

where the double subscript j_i indicates that the functions $\boldsymbol{u}(\boldsymbol{\theta})$ and $p(\boldsymbol{\theta})$ are evaluated at the i-th collocation point of the j-th boundary set of collocation points $(\boldsymbol{x}_{j,i}, t_{j,i}) \in \Gamma_{\mathrm{BC},j}$. Once again, the weights $\{\lambda_{\mathrm{BC},j}\}_{j=1}^{3}$ associated to the different boundary conditions are in general different from each other.

The definition of the physics-based loss functions (6) and (7) is central to the formulation of the PINN method. In the following, we will consider $\mathcal{L}_{\mathrm{train}}$ to be expressed as:

$$\mathcal{L}_{\mathrm{train}}(\boldsymbol{\theta}) = \mathcal{L}_{\boldsymbol{u}}(\boldsymbol{\theta}) + \mathcal{L}_{\hat{\boldsymbol{u}}}(\boldsymbol{\theta}), \tag{8}$$

where $\mathcal{L}_{\boldsymbol{u}}$ is the loss term of the velocity with respect to the data, and $\mathcal{L}_{\hat{\boldsymbol{u}}}$ is the loss term with respect to the velocity direction unitary vector data. The data are high-fidelity known values of the solution, available for a discrete subset of the spatio-temporal domain. These data are typically obtained from empirical or experimental measurements, and the type of data depends on the problem at hand. For fluid dynamics problems, the data can come in the form of the fluid velocity from 4D flow MRI acquisitions [36], like the focus of this work, or from PIV measurements [32]. Once the data are available, the entire dataset is split into training and testing datasets [17, 37]. The training dataset is used to define the loss function $\mathcal{L}_{\mathrm{train}}$ in Eq. (5). In particular, given the N_{train} training data points comprising of the velocity data $\{\boldsymbol{u}_{\mathrm{ref},i}\}_{i=1}^{N_{\mathrm{train}}}$ at N_{train} different points of the spatio-temporal domain, the data-driven loss terms in (8) are defined as:

$$\mathcal{L}_{\boldsymbol{u}}(\boldsymbol{\theta}) = \sum_{i=1}^{N_{\mathrm{train}}} \sum_{j=1}^{3} \frac{\lambda_{(\boldsymbol{u})_j}}{N_{\mathrm{train}}} \left(|(\boldsymbol{u}_i(\boldsymbol{\theta}))_j - (\boldsymbol{u}_{\mathrm{ref},i})_j|^2 + \lambda_{\boldsymbol{u},\mathrm{MSLE}} \left| \log\left(\frac{(\boldsymbol{u}_i(\boldsymbol{\theta}))_j + 1}{(\boldsymbol{u}_{\mathrm{ref},i})_j + 1} \right) \right|^2 \right), \tag{9a}$$

$$\mathcal{L}_{\hat{\boldsymbol{u}}}(\boldsymbol{\theta}) = \sum_{i=1}^{N_{\mathrm{train}}} \frac{\lambda_{\hat{\boldsymbol{u}}}}{N_{\mathrm{train}}} \left(|\hat{\boldsymbol{u}}_i(\boldsymbol{\theta}) - \hat{\boldsymbol{u}}_{\mathrm{ref},i}|^2 + \left| \log\left(\frac{\hat{\boldsymbol{u}}_i(\boldsymbol{\theta}) + \mathbf{1}}{\hat{\boldsymbol{u}}_{\mathrm{ref},i} + \mathbf{1}} \right) \right|^2 \right). \tag{9b}$$

The terms in $\mathcal{L}_{\boldsymbol{u}}$ (9a) correspond to the MSE and MSLE of the velocity vector, where the subscript j in (9a) indicates the j-th Cartesian component of the velocity vector. On the other hand, by introducing the loss term $\mathcal{L}_{\hat{\boldsymbol{u}}}$ (9b) related to the velocity vector field direction $\hat{\boldsymbol{u}}$, with $\mathbf{1} \in \mathbb{R}^3$, we enhance the velocity direction information obtained from the training dataset, even when the velocity magnitude is small [38]. To avoid problems with zero-magnitude vectors, we compute the unit vector $\hat{\boldsymbol{u}}$ as:

$$\hat{\boldsymbol{u}} = \frac{\boldsymbol{u}}{\max\{|\boldsymbol{u}|_2, \varepsilon\}}, \quad \text{with} \quad \varepsilon = 10^{-2}.$$

We remark that we do not use pressure data for the ANN training, since we aim to reconstruct it Eq. (2b); this choice is consistent with 4D flow MRI data, where only velocity measurements are available.

We train the PINN, consisting of solving the minimization problem (4), either by using a stochastic gradient descent algorithm, such as the ADAM optimizer [17, 39], for large datasets, or by using a quasi-Newton optimization algorithm such as L-BFGS-B [40] in the case of small datasets [21].

Finally, the accuracy of the ANN approximations of the velocity and the pressure is evaluated by using the remaining N_{test} data comprising the testing dataset. To this end, analogously to Eq. (9a), we define the velocity and pressure test loss functions, $\mathcal{L}_{\boldsymbol{u}}^{\mathrm{test}}$ and $\mathcal{L}_{p}^{\mathrm{test}}$ respectively:

$$\mathcal{L}_{\boldsymbol{u}}^{\mathrm{test}}(\boldsymbol{\theta}) = \frac{1}{N_{\mathrm{test}}} \sum_{i=1}^{N_{\mathrm{test}}} |\boldsymbol{u}_i(\boldsymbol{\theta}) - \boldsymbol{u}_{\mathrm{ref},i}|^2, \tag{10a}$$

$$\mathcal{L}_{p}^{\mathrm{test}}(\boldsymbol{\theta}) = \frac{1}{N_{\mathrm{test}}} \sum_{i=1}^{N_{\mathrm{test}}} |p_i(\boldsymbol{\theta}) - p_{\mathrm{ref},i}|^2. \tag{10b}$$

The test data loss function $\mathcal{L}_{p}^{\mathrm{test}}$ is well-defined only when pressure data are available.

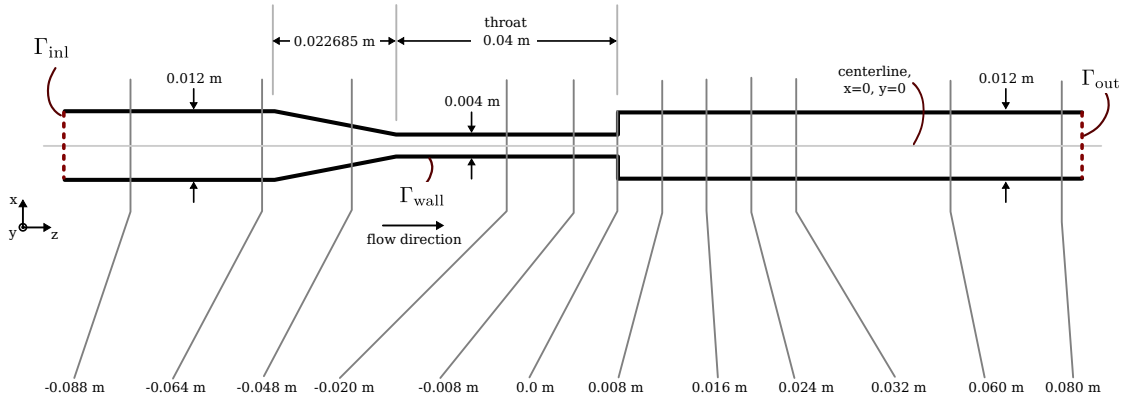


Figure 1: FDA nozzle domain geometry, characteristic dimensions with sketch of inlet boundary Γ_{inl}, outlet boundary Γ_{out}, and wall Γ_{wall}. Vertical lines in gray are cross-sections where experimental data are available.

3. Results

In this section we numerically verify and validate our computational pipeline based on the PINN methodology presented in Section 2 on two test cases. To this end, we employ both real measurement data and FEM generated in silico data.

Section 3.1 describes the FDA nozzle benchmark, whereas in Section 3.1.1, the FDA-CFD cases address the in silico verification of different PINN configurations by training and testing FEM data, whereas in Section 3.1.2 the FDA-EXP cases are devoted to experimental validation by training and testing the PINNs on PIV measurement data. On the other hand, Section 3.2 focuses on the fluid flow in an aneurysm geometry. In Section 3.2.1 the ANE-CFD networks are trained on an in silico FEM generated dataset of fluid flow through an aneurysm geometry, and in Section 3.2.2 the ANE-4DMRI cases are trained using experimental in vitro data.

3.1. FDA nozzle benchmark

First, we apply the PINN method to the laminar case of the FDA nozzle benchmark [32, 33]. We consider a laminar flow through an axially symmetric nozzle, whose geometry and dimensions are represented in Figure 1. First, in Section 3.1.1, we verify the proposed PINN method by performing a CFD simulation of our problem and by using a subset of the generated in silico data to train and test the PINN. Then, in Section 3.1.2, we validate the method by employing real experimental data from PIV measurements [32, 33].

3.1.1. Numerical verification against in silico data

We discuss the verification of the PINN method for the FDA nozzle benchmark by training and testing against FEM generated in silico data. In Section 3.1.1 we describe the setup for our problem, focusing first on data generation, then on the training of the PINNs. Following this, in Section 3.1.1, we present a sensitivity study of the FDA-CFD PINN output by considering different definitions of the loss function.

In particular we study 7 different PINN configurations, named FDA-CFD and numbered 1 through 7, using 5 different tests. The exact configurations are reported in Table 2. The tests are designed as follows:

- In Test 1, we examine the effects of modifying the weights of the radial components in the velocity. This is done by comparing FDA-CFD-1 and 2, where FDA-CFD-2 has a higher weights on the radial components of the velocity in the data-driven term;
- In Test 2, we examine the effect of assigning different weight on terms coming from the PDE-based loss function on the velocity reconstruction accuracy. To this end, we compare FDA-CFD-2 and 3, by forcing the residuals in the latter to be of the same order of magnitude;

- In Test 3, by introducing in FDA-CFD-4 the MSLE, we improve the reconstruction of the centerline pressure;
- In Test 4, we test the centerline pressure and velocity accuracy reconstruction depending on the PINN size, by using twice the number of neurons per layer in FDA-CFD-5 with respect to FDA-CFD-4;
- In Test 5, we test the performance of FDA-CFD-6 and FDA-CFD-7 by including the velocity-direction dependent loss function in differently sized networks

Setup. To generate data for the training and testing of the PINNs, we first carry out CFD simulations. We perform the simulations using `life`$^{\mathtt{X}}$ [41, 42], a high-performance solver of multiphysics and multiscale differential models developed at the MOX laboratory of Politecnico di Milano. The software is based on the `deal.II` finite element core [43, 44] and mainly targets cardiocirculatory applications.

We consider a laminar flow in the nozzle characterized by a Reynolds number at the throat section of $\mathbb{R}\mathrm{e} = 500$. As the characteristic velocity U we consider the averaged velocity over the throat cross-section. We set the fluid flow parameters, reported in Table 1, first column, in accordance with the FDA nozzle benchmark [31].

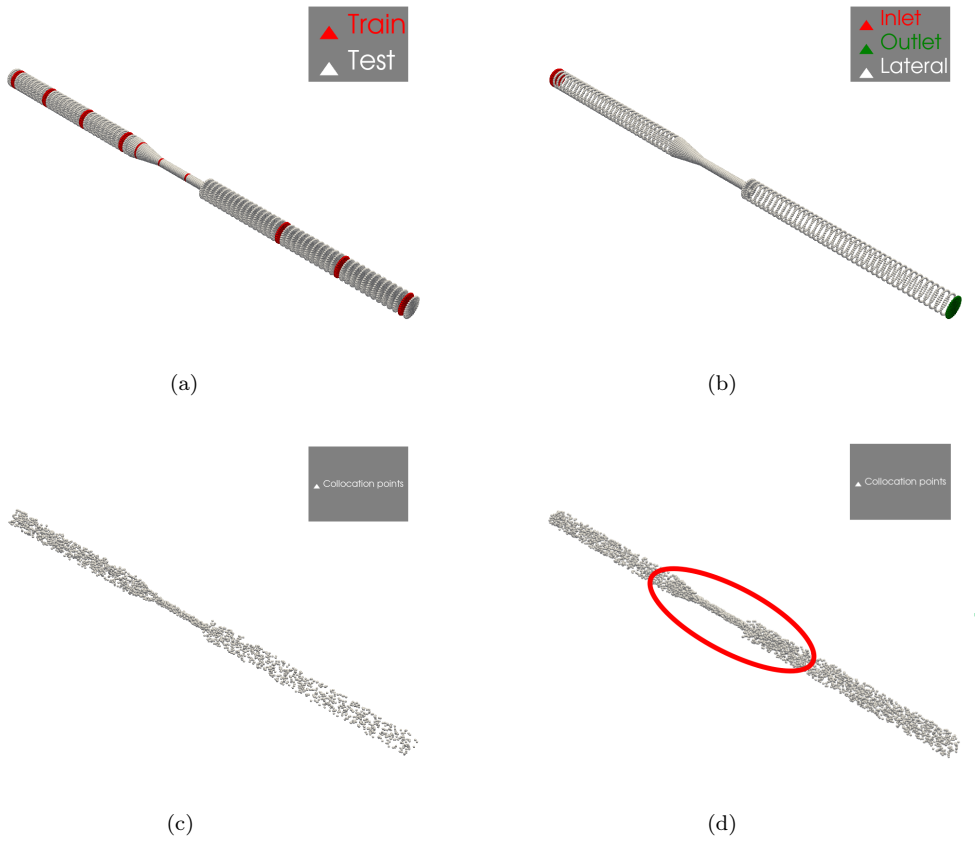
In accordance with Eq.s (1c), (1e) and (1d), we impose a parabolic velocity profile with a volumetric flow rate of Q at the inlet section Γ_{inl}, a homogeneous Neumann boundary condition at the outlet section Γ_{out}, and a no-slip condition at the lateral wall of the nozzle Γ_{wall}. As the final time of the simulation we set $T = 2$ s.

For the space discretization we use trilinear continuous nodal finite elements (FE) for both velocity and pressure, stabilized with the Variational-Multiscale Large Eddy Simulation (VMS-LES) method [8, 45]. We bring the solution to a steady-state regime by using a smooth ramp lasting 0.01 s starting from a null velocity initial condition. For the time integration of the unsteady solver we use the Backward Euler method with a semi-implicit treatment of the nonlinear term [45]. We consider two hexahedral mesh refinement levels, a coarse level and a fine level. We report the simulation parameters in Table A.8. Simulations were ran on 20 cores on the `gigat` cluster of the Department of Mathematics, Politecnico di Milano [46]. Once we have generated the in silico FEM data, we perform the ANN initialization, training and testing.

The PINN inputs are the spatial coordinates x, y, z, whereas the outputs are the pressure p and the velocity $\boldsymbol{u}$. We use three hidden layers and hyperbolic tangent activation functions [47]. These dimensions were chosen on the basis of trial and error, and while it is possible to render them hyperparameters and tune them further using automatic parameter optimization, we skip this procedure due to its significant additional computational cost. In order to address the problem of vanishing and exploding gradients during training, we use batch-normalization before each layer [48, 49]. We consider the total loss function $\mathcal{L}$ defined in Eq. (5). We recall that the pressure does not enter the definition of the total loss function $\mathcal{L}$, of Eq. (5). Instead, the pressure field will be directly reconstructed from the velocity using Eq. (2b). In order to avoid saturation of the activation function [50], we normalize the features by making spatial coordinates and flow properties dimensionless. Moreover, normalizing the outputs guarantees that they will all have the same order of magnitude [51].

We split the dataset into a training and testing subsets by selecting specific cross-sections as depicted in Figure 2a. We consider 10 training sections, with a total number of 2922 data points. We randomly sampled 25% of the data points over each section, thus obtaining a total of $N_{\text{train}} = 730$ training data points. The collocation points of the boundary portions are sketched in Figure 2b. We use a total of $N_{\text{BC},2} = 2045$ collocation points at the wall. The N_{PDE} collocation points are randomly sampled over the domain, and are depicted in Figure 2c and Figure 2d, depending on the required refinement level.

We consider different numerical setups and ANN architectures. We study the accuracy of the reconstruction of the velocity and pressure fields depending on the scaling by modifying the weights λ associated with each loss term in Eq. (5). Moreover, we examine the effect of including a higher number of PDE collocation points in the convergent region, where we expect the solution to exhibit high gradients. Lastly, we study the effects of increasing the number of neurons per layer N_{neuron} on the ANN accuracy. The parameters of the loss functions used for the training of the different PINNs are reported in Table 2.



(a) (b)

(c) (d)

Figure 2: Collocation points for the FDA nozzle benchmark. **(a)** Training dataset sections (red) and testing dataset sections (gray). **(b)** Boundary collocation points. Inlet boundary (red), outlet boundary (green) and lateral wall (gray). **(c)** PDE collocation points with $N_{\text{PDE}} = 3093$. **(d)** PDE collocation points with a higher refinement in the circled region (red), $N_{\text{PDE}} = 4341$.

All ANN initialization, training and testing was done using `TensorFlow` [52] and `Keras` [53]. Training was performed using 100 epochs of the ADAM optimizer and 30 000 epochs of L-BFGS-B for the smaller ANNs having 16 neurons per layer, whereas 100 epochs of the ADAM optimizer and 50 000 epochs of L-BFGS-B were used for the larger ANNs having 32 neurons per layer.

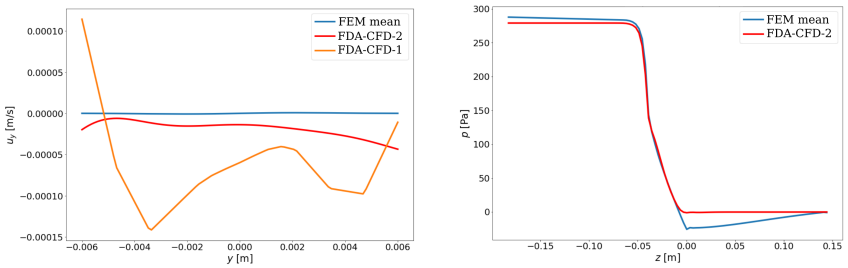
Test 1. First, we compare the effects of modifying the scaling, i.e. the weights λ associated with the data-driven loss term $\mathcal{L}_{\boldsymbol{u}}$ in (8) on the accuracy of the reconstructed velocity. We investigate the use of larger weights for the radial velocity components with respect to the axial one in $\mathcal{L}_{\boldsymbol{u}}$ (8), and therefore compare the two PINNs trained with the FDA-CFD-1 and the FDA-CFD-2 setup.

The radial velocity components values computed by the ANN are large compared to the FEM solution which is similar to a Poiseuille flow, as reported in Figure 3a. Figure 3a also shows that using larger weights for the radial components in the loss function $\mathcal{L}_{\boldsymbol{u}}$ can mitigate this effect, as evidenced by the radial velocity reconstruction of FDA-CFD-2. We believe that the worse approximation results in regions where flow-field gradients are low are in part due to the higher relative contributions of the physics loss function $\mathcal{L}_{\text{PDE}}$ in regions with high velocity gradients, such as the throat region.

We recall that there is no training data on pressure, and it is instead reconstructed from the momentum balance in Eq. (2b). We obtain a constant pressure before the convergent and after the diffuser, as shown in Figure 3b, where we compare the pressure reconstruction obtained from the PINN with the FEM solution.

	FDA-CFD						
	1	2	3	4	5	6	7
$\lambda_{(\boldsymbol{u})_1}$	0.91	100	100	100	100	0.17	0.17
$\lambda_{(\boldsymbol{u})_2}$	0.91	100	100	100	100	0.17	0.17
$\lambda_{(\boldsymbol{u})_3}$	0.91	0.91	0.91	0.91	0.91	0.17	0.17
$\lambda_{\text{BC},1}$	0.5	0.5	0.5	0.5	0.5	0.5	0.5
$\lambda_{\text{BC},2}$	1.7	1.7	1.7	1.7	1.7	1.7	1.7
$\lambda_{\text{BC},3}$	0.5	0.5	0.5	0.5	0.5	0.5	0.5
$\lambda_{\text{PDE},1}$	2	2	2	3	3	3	3
$\lambda_{\text{PDE},2}$	1.6	1.6	0.308	0.308	0.308	0.231	0.601
$\lambda_{\text{PDE},3}$	1.6	1.6	0.308	0.308	0.308	0.231	0.601
$\lambda_{\text{PDE},4}$	15	15	2.88	2.88	2.88	2.88	2.88
$\lambda_{\text{PDE,MSLE}}$	0	0	0	1	1	1	1
$\lambda_{\boldsymbol{u},\text{MSLE}}$	0	0	0	1	1	1	1
$\lambda_{\hat{\boldsymbol{u}}}$	0	0	0	0	0	0.1	0.1
N_{PDE}	3093	3093	3093	3093	3093	4341	4341
N_{neuron}	16	16	16	16	32	16	32

Table 2: Parameters for the loss functions used for PINNs training in Section 3.1.1.

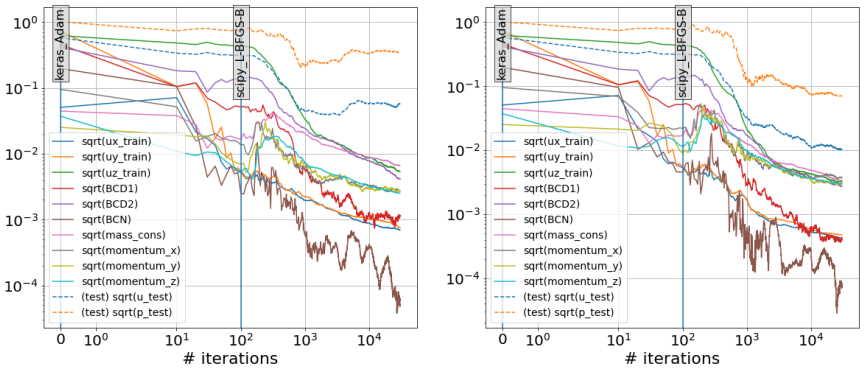


(a) (b)

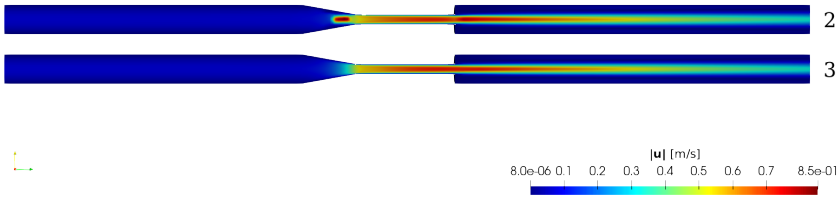
Figure 3: Test 1: Comparison of approximation properties of FDA-CFD-1 and FDA-CFD-2 for low magnitude quantities and gradients. **(a)** Velocity component along the y-axis at $z = -0.1$ m. **(b)** Comparison of the centerline pressure along the z-axis.

Test 2. In this test, we compared the effects of a physically consistent scaling on the accuracy of the velocity reconstruction. We choose the weights $\{\lambda_{\text{PDE},j}\}_{j=1}^{4}$ for FDA-CFD-3 such that the total contributions of the mass, momentum and Dirichlet no-slip residuals to $\mathcal{L}_{\text{PDE}}$ (6) are all of the same order of magnitude. We found that this choice leads to more accurate results compared to FDA-CFD-2. We report the values of the loss functions during training for FDA-CFD-2 and FDA-CFD-3 in Figure 4a. In Figure 4a, we see that by reducing the weights for the momentum balance residuals in FDA-CFD-3, we achieve lower values of the test loss functions $\mathcal{L}_{\text{test}}^{\boldsymbol{u}}$ (10a) and $\mathcal{L}_{\text{test}}^{p}$ (10b) compared to FDA-CFD-2, for which the test loss functions cease to decrease after about 1000 iterations. In Figure 4b we show the velocity magnitude for the two setups. The results of FDA-CFD-2 show a nonphysical behavior in the convergent of the nozzle. Moreover, we observed that FDA-CFD-2 has a larger dependence on the weights initialization than FDA-CFD-3. This indicates that FDA-CFD-3, for which the contributions of the physics residuals (3) and (2) are of the same order of magnitude, provides a better and more robust approximation than its counterpart, FDA-CFD-2.

Test 3. For the FDA-CFD-4 PINN, we study the influence of the inclusion of the MSLE into the training loss function on the approximation of low magnitude data. By including the MSLE in $\mathcal{L}_{\text{PDE}}$, we minimize more the residuals where flow-field gradients are low and we achieve a better pressure gradient reconstruction. This is shown in Figure 5b: by adding the MSLE in FDA-CFD-4, we predict a nonzero pressure gradient at the end of the duct. This is different from what we observed in FDA-CFD-2, in which we obtain a zero



(a)



(b)

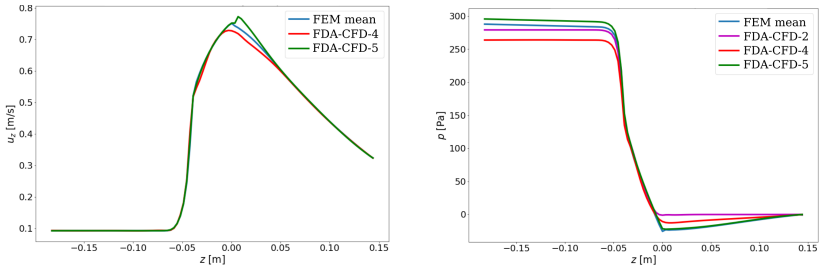
Figure 4: Test 2: Comparison between FDA-CFD-2 and FDA-CFD-3 setups. **(a)** Total loss function components values during minimization. Left: FDA-CFD-2. Right: FDA-CFD-3.**(b)** Velocity magnitudes. Top: FDA-CFD-2. Bottom: FDA-CFD-3.

pressure gradient after the diffuser, which is not consistent with the FEM solution.

Test 4. We study the impact of increasing the number of neurons of the PINN on the accuracy of the pressure and velocity reconstructions. Figure 5b shows that the pressure reconstruction becomes more accurate as more neurons are used in the network. This is especially true in the region downstream the throat. In Figure 6a, we report the magnitude of the momentum equation residual in the z direction. Indeed, for FDA-CFD-5, we obtain a lower residual compared to FDA-CFD-4, which reflects the higher accuracy when reconstructing the pressure.

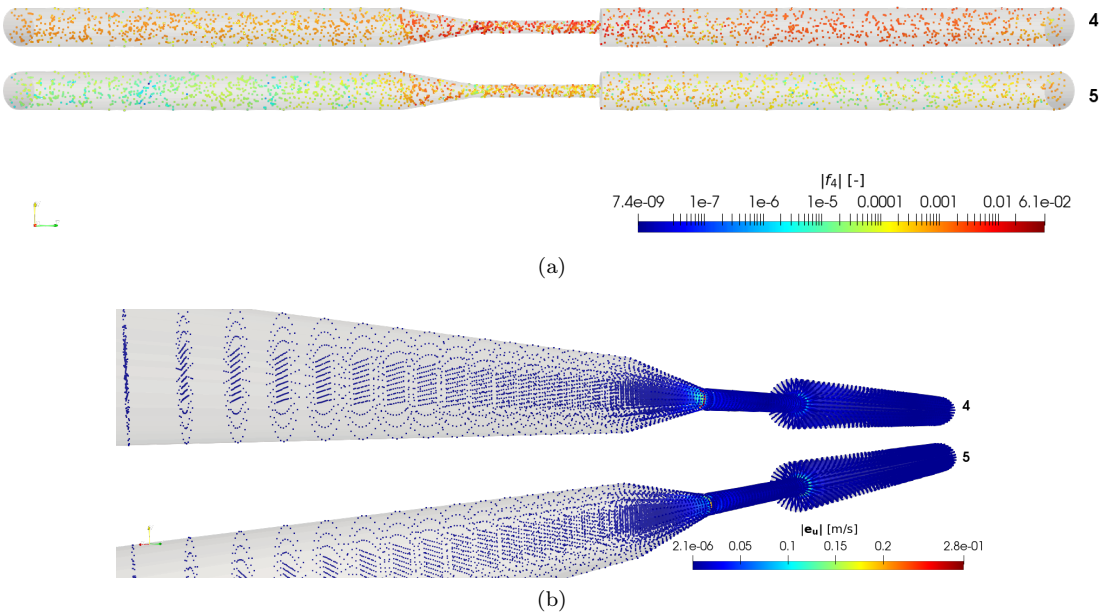
While the centerline pressure reconstruction for FDA-CFD-5 improves, the velocity $(\boldsymbol{u})_z$ evaluated at the duct axis has an overshoot with respect to the FEM simulation after the diffuser, as seen in Figure 5a. We remark that we do not give any training data section to the ANN immediately after the diffuser, as reflected in Figure 2a, so the solution there is retrieved only from the physics regularization term. We observe that the momentum balance residuals along the z-axis are larger at the end of the convergent, as shown in Figure 6a. The same also holds for the absolute errors of the approximated velocity with respect to the FEM simulation, reported in Figure 6b.

Test 5. In this test, we examine the effects of adding the velocity direction data in the training loss function. In Figures 7a and 7b we can see the results of FDA-CFD-6 and FDA-CFD-7 for the centerline pressure and the velocity. In Figure 7a we see that FDA-CFD-7 captures the peak of the centerline velocity from the FEM simulation, whereas FDA-CFD-6 underestimates it and smooths it out. We recall once more that we do not give any training data near the diffuser to the ANN, as depicted in Figure 2a. Similarly to FDA-CFD-4, FDA-CFD-6 predicts a zero pressure gradient at the beginning of the duct. Meanwhile, the pressure reconstruction in FDA-CFD-7 is coherent with the CFD simulation results. On the other hand, the higher number of parameters of the larger ANN, FDA-CFD-7, means that it overfits the numerical noise in the FEM simulations with respect to its counterpart with a lower number of parameters, FDA-CFD-6, as reflected in Figure 7c.



(a) (b)

Figure 5: Tests 3 and 4: Comparison of computed quantities for the FEM simulation (blue) and the PINNs. **(a)** Centerline axial velocity $(\boldsymbol{u})_z$. **(b)** Centerline pressure p.



(a)

(b)

Figure 6: Test 4: Pointwise comparison between FDA-CFD-4 and FDA-CFD-5. **(a)** Absolute value of the momentum balance residual along the z-axis, $|f_4|$, in log scale, evaluated at the collocation points. FDA-CFD-4 (top) and FDA-CFD-5 (bottom). **(b)** Velocity absolute error magnitude. FDA-CFD-4 (top) and FDA-CFD-5 (bottom).

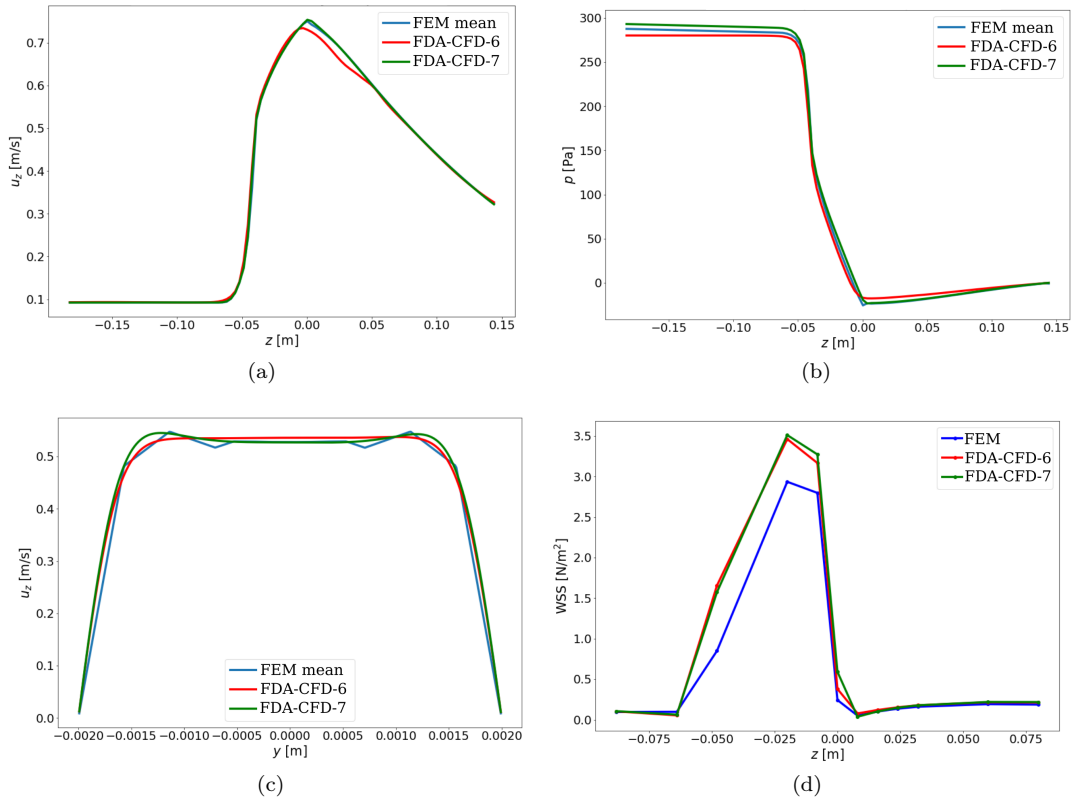


Figure 7: Test 5: Comparison of computed quantities for the FEM simulation (blue) and the PINNs. **(a)** Centerline axial velocity $(\boldsymbol{u})_z$. **(b)** Centerline pressure p. **(c)** Axial velocity $(\boldsymbol{u})_z$ distribution over training section at $z = -0.03875$ m. **(d)** Wall-averaged WSS magnitude comparison.

	FDA-EXP-1	FDA-EXP-2	FDA-EXP-3
Removed section	None	1	1 and 2
Section sampling	15%	20%	20%

Table 3: Setup for the different PINNs in Section 3.1.2. For section numbering we refer to Figure 8c.

Starting from the FEM simulation and ANN outputs, we compute the wall shear stress (WSS), a hemodynamic index commonly used for disease diagnosis [6, 7], as follows:

$$\epsilon = \frac{1}{2}\left(\nabla \boldsymbol{u} + (\nabla \boldsymbol{u})^T\right), \qquad \tau = 2\mu\epsilon, \qquad \mathbf{WSS} = \tau \boldsymbol{n} - (\tau \boldsymbol{n} \cdot \boldsymbol{n})\boldsymbol{n}. \tag{11}$$

In Eq. (11), the second-order tensor ϵ is the symmetric part of the velocity gradient, called the shear rate tensor, the second-order tensor τ is the shear stress tensor, and **WSS** is the vector WSS, defined on Γ_{wall}. We compute the WSS[1] magnitude at each axial coordinate z and average it over the wall. We report the computed wall-averaged WSS for FDA-CFD-6 and FDA-CFD-7 compared to the mean FEM simulation in Figure 7d. Both ANNs predict a similar WSS distribution along the z-axis, which is larger than the one predicted using the FEM simulation.

3.1.2. Validation against experimental data

We train and test the ANN using solely the experimental data obtained from PIV measurements [32] for the FDA nozzle benchmark dataset [33]. When using the in silico dataset, the fluid properties were available in all points of the computational domain. Differently, in this case, the velocity is available only on certain cross-sections. First, in Section 3.1.2, we describe the setup. Subsequently, in Test 6, we test the FDA-EXP networks by progressively removing data sections on which the PINNs are trained.

Setup. We describe the problem setup using the PIV experimental data. The data consist of axial and radial velocity samples on 12 different cross-sections, depicted in Figure 8c. Five different data acquisitions are available for the FDA nozzle benchmark. We employ the dataset labeled `999` for the training, as it provides the most extensive results [32]. The axial and radial velocity samples are represented in Figures 8a and 8b.

The training dataset is obtained starting from the 8992 data points by removing select cross-sections and randomly sampling the data points over each remaining cross-section. We consider three different setups whose sampling rates and removed cross-sections are reported in Table 3. To distribute the samples circumferentially over each section, we rotate the points and the velocity samples on the y-axis with respect to the z-axis by 22.5° increments, resulting in angles ranging from 0° to 157.5°. The resulting velocity vector data alignment is shown in Figure 8d.

The ANN architecture, minimization algorithm and loss function parameters are the same as for FDA-CFD-7 in Section 3.1.1, with the parameters reported in Table 2, the only difference being we set $\lambda_{\hat{\boldsymbol{u}},\text{MSLE}} = \lambda_{\boldsymbol{u},\text{MSLE}} = 0$. We disregard the MSLE in the training loss function since we observed no major differences by incorporating it along with the unit velocity vector loss function $\mathcal{L}_{\hat{\boldsymbol{u}}}$.

Test 6. We compare the three different PINNs, trained with different sampling rates and by removing different cross-sections, as reported in Table 3. In Figure 9a we report the reconstructed normalized pressure along the duct axis, where we see that all PINN setups performed far better at replicating experimental data than the FEM simulation.

By comparing the reconstructed and experimental axial velocity for FDA-EXP-2 at the removed section at $z = 0.024$ m, in Figure 9b, we see that both PINNs compensate for the slight asymmetry in the experimental data. This also holds for FDA-EXP-3, where we have removed two sections from the training dataset, the first one at $z = 0.024$ m as in FDA-EXP-2, and the second one at $z = -0.008$ m. While the first

[1]For the FEM simulation results we compute the WSS in postprocessing using Paraview [54], whereas for the PINN results we use differential operators implemented in `nisaba`, that rely on automatic differentiation.

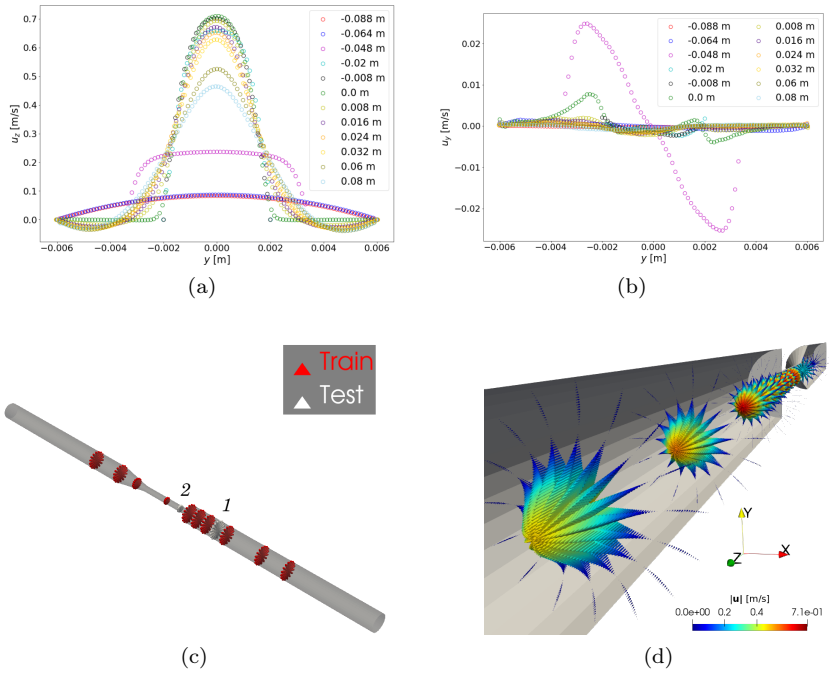


Figure 8: PIV measurements from the 999 dataset [32]. **(a)** Axial velocity $(\boldsymbol{u})_z$ profile along the y-axis at different z values. **(b)** Radial velocity $(\boldsymbol{u})_x$ and $(\boldsymbol{u})_y$ along the y-axis at different z values. **(c)** Training and testing data sections partition. Test sections are: 1 at $z_1 = -0.008$ m and 2 at $z_2 = 0.024$ m. **(d)** Vector plot of the rotated data.

section is in a region where there are several other training sections, the second section is in a region where data are scarce. Moreover, it is located near the discontinuity due to the diffuser. Even so, the obtained velocity profile for the second section is very close to the experimental data, as evidenced by Figure 9c.

In order to quantify the accuracy of the PINN approximations an to examine the effects of adding the experimental data in the loss function, we define errors between the axial velocity over each cross-section from the FEM simulation $(\boldsymbol{u})_z$ and an average among the axial velocities of the five available experimental datasets $(\boldsymbol{u}_{\text{exp}})_z$.

We define the errors E_z and $\overline{E_z}$ for a generic cross-section with axial coordinate z as:

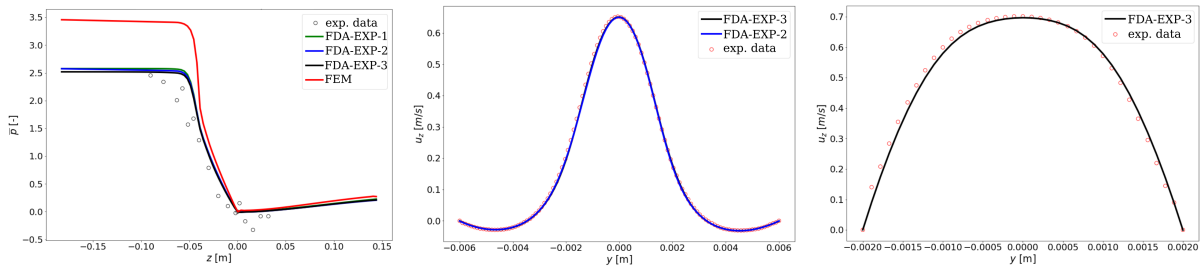
$$E_z = \frac{1}{N_{z,\text{sec}}} \sum_{i=1}^{N_{z,\text{sec}}} \left| \frac{(\boldsymbol{u}_{\text{exp},i})_z - (\boldsymbol{u}_i)_z}{(\boldsymbol{u}_{\text{exp},i})_z} \right|, \qquad \overline{E_z} = \frac{1}{N_{z,\text{sec}}} \sum_{i=1}^{N_{z,\text{sec}}} \left| \frac{(\boldsymbol{u}_{\text{exp},i})_z - (\boldsymbol{u}_i)_z}{(\bar{\boldsymbol{u}}_{\text{exp}})_z} \right|, \tag{12}$$

where $N_{z,\text{sec}}$ is the total number of points on the given cross-section and $\bar{\boldsymbol{u}}_{\text{exp}}$ is the average of the derived experimental values $\boldsymbol{u}_{\text{exp},i}$ over the cross-section,

$$\bar{\boldsymbol{u}}_{\text{exp}} = \frac{1}{N_{z,\text{sec}}} \sum_{i=1}^{N_{z,\text{sec}}} \boldsymbol{u}_{\text{exp},i}.$$

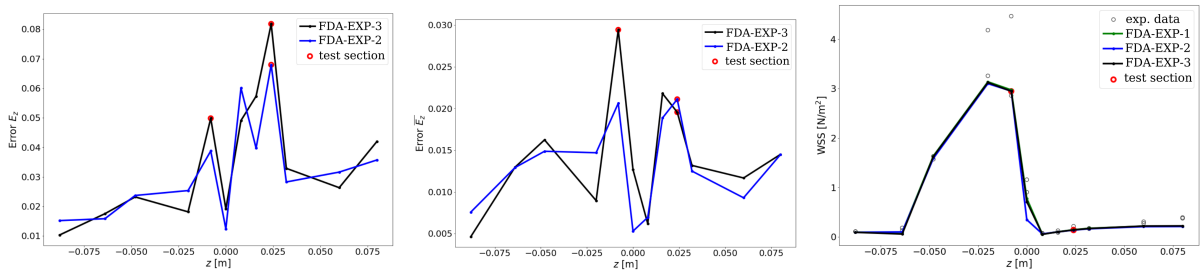
In other words, in Eq. (12), E_z is the average of the axial velocity relative errors on the cross-section, while in $\overline{E_z}$ the relative error is computed with respect to the average experimental axial velocity on the cross-section. In particular, by considering $\overline{E_z}$ we avoid divisions by small values of $u_{z,\text{exp}}$ near the wall that might cause E_z to assume very large values.

We compute the errors with respect to the experimental dataset using Eq. (12), and report them in Figures 10a and 10b. Both FDA-EXP-2 and FDA-EXP-3 display a similar error trend. In addition to the approximated velocities, we compute the wall-averaged WSS using Eq. (11) and compare it to the



(a) (b) (c)

Figure 9: Test 6: Comparison of the PINN results with the 999 experimental data [32].**(a)** Normalized center-line pressure $\overline{p}$. **(b)** Axial velocity profile at $z = 0.024$ m. **(c)** Axial velocity profile at $z = -0.008$ m.



(a) (b) (c)

Figure 10: Test 6: Comparison of errors computed from the PINNs with the ones computed using the experimental data on the sections. Red circles represent values at test sections. **(a)** Cross-section average of the relative error E_z. **(b)** Error of the average cross-section velocity $\overline{E_z}$. **(c)** Wall-averaged WSS magnitude comparison.

experimental data in Figure 10c. The computed wall-averaged WSS approximates the experimental data well in the low-stress region, even though the data are quite dispersed. However, in the convergent region, where the WSS data are highest, all ANNs underestimate the WSS.

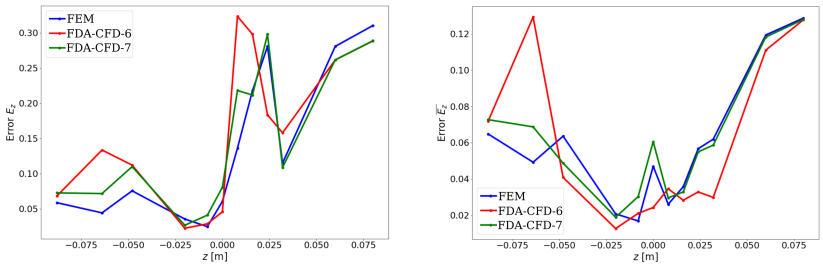
In order to examine the effect of adding the experimental dataset to the loss function, we compute the errors using Eq. (12) of the PINNs in Section 3.1.1, that are trained only using in silico data. We report the computed errors in Figures 11a and 11b. From Figures 11a and 11b we can see that the FDA-CFD-6 setup leads to larger errors in the first part of the duct, whereas the errors obtained using the FDA-CFD-7 setup follow more closely the error of the FEM simulation. This is coherent with the fact that the training data are taken from the FEM simulation and that FDA-CFD-7 has a larger number of neurons and therefore parameters. Overall, we see that PINNs trained using only the FEM data exhibit overall larger errors than the PINNs trained using the experimental data.

3.2. Application to blood flows in aneurysm

We apply the PINN method to describe the flow in a realistic aneurysm geometry. First, in Section 3.2.1, we verify the PINN method using FEM simulation data. Then, in Section 3.2.2, we show results for the validation of our method using 4D flow MRI data [55].

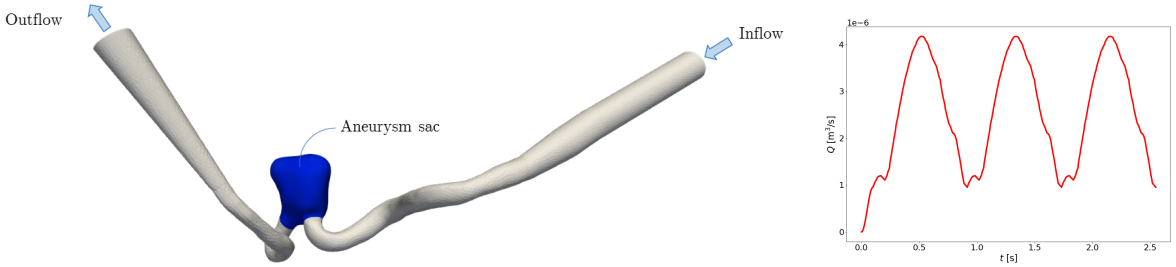
3.2.1. Numerical verification against in silico data

We perform PINN validation with in silico data obtained from FEM simulations. In Section 3.2.1 we describe the FEM simulation setup and results used for the training. Next, in Section 3.2.1 we discuss the results of the PINN training.



(a) (b)

Figure 11: Test 5: Comparison of errors with respect to the experimental data computed from the PINNs with the ones computed using FEM data. **(a)** Cross-section average of the relative error E_z. **(b)** Error of the average cross-section velocity $\overline{E_z}$.



(a) (b)

Figure 12: Aneurysm geometry and simulation features. **(a)** Aneurysm model, provided by [55]. The region where 4D flow MRI data are available is in blue. **(b)** Imposed inlet flow rate over time.

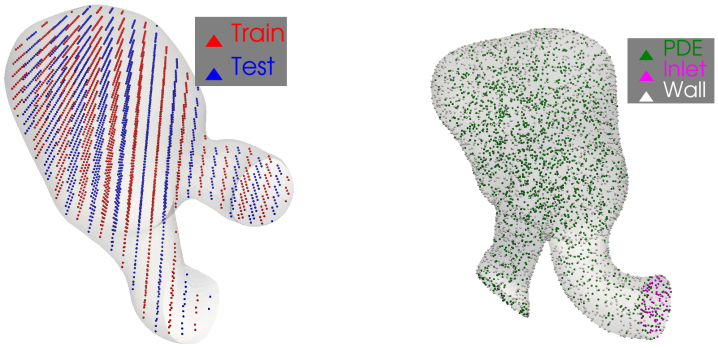
Setup. We consider the aneurysm model represented in Figure 12a, provided in [11]. In their study, the geometry is obtained from an in vitro patient-derived 3D-printed aneurysm model. It is segmented from an intraophthalmic, extradural internal carotid artery aneurysm of a patient (52 years, female) [55].

We carry out the FEM simulations using `lifeˣ` [41]. We set the flow properties, reported in Table 1, second column, in accordance with the experimental data [55]. The Reynolds number is $\mathbb{R}\mathrm{e} \approx 1500$ at the inlet of the aneurysm sac. We impose a parabolic velocity profile at the inlet with a volumetric flow rate as shown in Figure 12b. We apply a homogeneous Neumann boundary condition at the outlet, and a Dirichlet no-slip boundary condition at the lateral wall. As in Section 3.1.1, we apply the Backward Euler scheme with a semi-implicit treatment of the nonlinear term along with the VMS-LES method. We consider simulations on two different mesh refinement levels and simulate a total of three heartbeats. We report the simulation parameters in Table A.8. Simulations were ran on 32 cores on the `gigatlong` cluster of the Department of Mathematics, Politecnico di Milano [46].

For training/test data generation we consider a part of the domain in the proximity of the aneurysm sac. We sample the velocity field over a uniform grid with a resolution of $0.7 \times 0.7 \times 0.7$ mm^3 along the Cartesian directions, which yields a velocity distribution coherent with the 4D flow MRI data [11]. We obtain the training dataset by removing half of the cross-sections along the z direction, according to Figure 13a.

Regarding the loss function associated to the PDE, we use the same definition of residual as for the previously analyzed steady-state case in Section 3.1, by fixing the time in Eq. (2) to $t = 2.024$ s. Hence, we avoid the computational cost of adding the physical time as an input variable to the PINNs. Some information on the time-dependent terms is nonetheless recovered through the data-driven loss function $\mathcal{L}_{\text{train}}$. We define the total loss function $\mathcal{L}$ as:

$$\mathcal{L}(\tilde{\boldsymbol{\theta}}) = \mathcal{L}_{\text{PDE}}(\tilde{\boldsymbol{\theta}}) + \mathcal{L}_{\text{BC}}(\tilde{\boldsymbol{\theta}}) + \mathcal{L}_{\bar{p}}(\tilde{\boldsymbol{\theta}}) + \mathcal{L}_{\text{train}}(\tilde{\boldsymbol{\theta}}), \tag{13}$$



(a) (b)

Figure 13: Collocation points. **(a)** Training and testing dataset partition. **(b)** Collocation points for the PDE regularization.

where $\mathcal{L}_{\bar{p}}$ represents the zero mean constraint penalty on the pressure:

$$\mathcal{L}_{\bar{p}}(\tilde{\boldsymbol{\theta}}) = \frac{\lambda_{\bar{p}}}{N_{\text{PDE}}} \left| \sum_{i=1}^{N_{\text{PDE}}} p_i(\tilde{\boldsymbol{\theta}}) \right|^2 , \tag{14}$$

where $\lambda_{\bar{p}}$ is a positive parameter and the reconstructed pressure p is evaluated at the same N_{PDE} collocation points as $\mathcal{L}_{\text{PDE}}$. The role of the zero mean pressure loss function $\mathcal{L}_{\bar{p}}$ in Eq. (14) is akin to the zero mean constraint used in FEM simulations to guarantee pressure uniqueness when no Neumann boundary condition is applied. We introduce (14) since we observed that using it instead of the Neumann boundary condition residual (3c), by setting $\lambda_{\text{BC},3} = 0$, we improve the accuracy of the pressure reconstruction.

We use $N_{\text{train}} = 1420$ points for training and $N_{\text{test}} = 1436$ points for testing. We randomly seed $N_{\text{PDE}} = 1959$ collocation points for the PDE loss function and $N_{\text{BC},2} = 2057$ collocation points at the wall, which we report in Figure 13b.

For the ANNs we use three hidden layers of 32 neurons each, with batch normalization before each layer. We solve the minimization problem related to the training using 100 epochs of the ADAM optimizer, with a learning rate of 10^{-2}, and 14 000 epochs of L-BFGS-B. The total time to perform the numerical minimization[2] is approximately 3 hours.

We train three different PINNs with decreasing weights on the training data loss terms with respect to the physics loss term. We summarize the weights for the three different configurations in Table 4.

Results. We report the results of the PINNs trained using the FEM simulation output. In Figure 14, we show a comparison between the velocity magnitude field obtained from the FEM simulation and the velocity magnitude obtained from the PINNs. We observe that by decreasing the weight of the data in the loss function the peak velocity magnitude decreases. Next, in Figure 15, we report the reconstructed pressure fields, where we zero average the FEM simulation pressure over the domain. We observe that the main features of the pressure field are matched by the PINN output. However, AN-CFD-1 and AN-CFD-3 predict lower pressure gradients with respect to the FEM simulation in the two vessel branches. On the other hand, AN-CFD-2 matches the FEM simulation best, with some discrepancy close to the outlet.

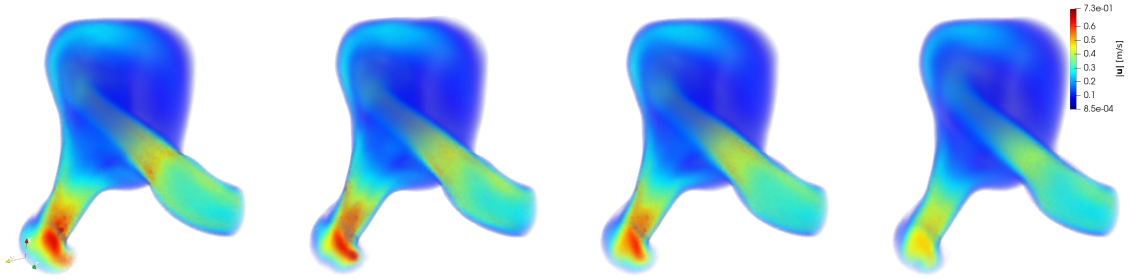
To avoid problems with the division by near-zero values, we compute the normalized error with respect to the average velocity magnitude over each train and test section using the following expression:

$$E_{\boldsymbol{u}} = \begin{cases} \dfrac{|\boldsymbol{u} - \boldsymbol{u}_{\text{ref}}|}{|\overline{\boldsymbol{u}}|}, & \text{if } |\overline{\boldsymbol{u}}| > 10^{-3}, \\ |\boldsymbol{u} - \boldsymbol{u}_{\text{ref}}|, & \text{otherwise.} \end{cases} \tag{15}$$

[2]Computational resources used for training are CPU 11th Gen Intel(R) Core(TM) i7-11800H @ 2.30GHz, GPU NVIDIA GeForce RTX 3070

	AN-CFD-1	AN-CFD-2	AN-CFD-3
λ_{u_1}	170	17	1.7
λ_{u_2}	170	17	1.7
λ_{u_3}	170	17	1.7
$\lambda_{\text{BC},1}$	0.5	0.5	0.5
$\lambda_{\text{BC},2}$	21.6	21.6	21.6
$\lambda_{\text{BC},3}$	0	0	0
$\lambda_{\text{PDE},1}$	5.76	5.76	5.76
$\lambda_{\text{PDE},2}$	1	1	1
$\lambda_{\text{PDE},3}$	1	1	1
$\lambda_{\text{PDE},4}$	1.56	1.56	1.56
$\lambda_{\hat{\boldsymbol{u}}}$	17	1.7	0.17
$\lambda_{\bar{p}}$	0.5	0.5	0.5

Table 4: Parameters for the loss functions used for PINNs training in Section 3.2.1.



(a) (b) (c) (d)

Figure 14: Computed magnitude of the velocity field. **(a)** FEM. **(b)** AN-CFD-1. **(c)** AN-CFD-2. **(d)** AN-CFD-3.

We report the computed errors in Figure 16. The computed errors for AN-CFD-1 are substantially larger on the test data than on the training data, whereas the errors on the test data for AN-CFD-3 follow more closely the errors on the training data. Even so, their overall magnitude is significantly larger than for the other two cases. On the other hand, AN-CFD-2, provides a good compromise between the two, since the error on the test data follows the training data error trend and both errors are of a smaller magnitude than the ones for AN-CFD-3.

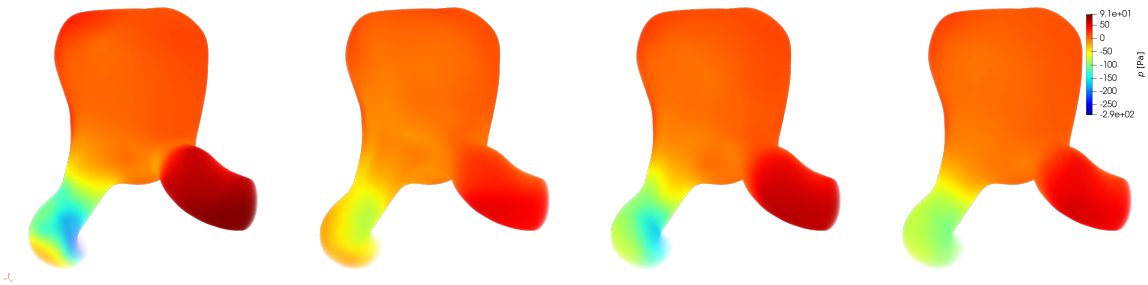
3.2.2. Validation against experimental data

We validate the PINNs using 4D flow MRI data. First, in Section 3.2.2, we provide a description of the 4D flow MRI data from [11] and the setup for the PINNs training. Next, in Section 3.2.2, we examine the obtained results.

Setup. The 4D flow MRI data provided by [11] are available only in the aneurysm sac, as shown in Figure 12a. We report the fluid flow properties in Table 1, third column. The flow is pulsatile, with the period corresponding to a single heartbeat of a duration of 0.816 s. The three velocity components are available in each point of the spatial domain in 24 time samples, with a time interval between each sample of 0.034 s.

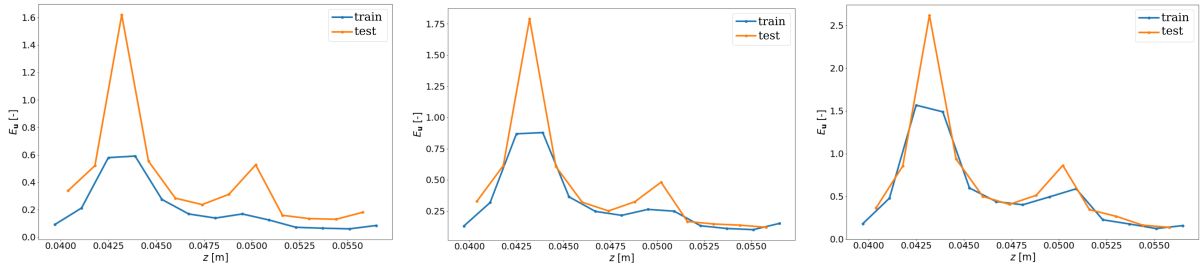
Following the error analysis in [11], we use the the lower resolution 1.5 mm^3 voxel dataset for training and the higher resolution 1.0 mm^3 voxel dataset for comparison. We use the datasets with a CS factor of 2.5. For the two datasets, 144 and 224 samples are given in the x direction, 144 and 224 in the y direction, and 40 and 60 samples in the z direction.

Due to the data having a different orientation from the model depicted in Figure 12a, we recenter the measurements with respect to the model. In Figure 17a we depict the measurement points with respect



(a) (b) (c) (d)

Figure 15: Reconstructed pressure field. **(a)** FEM. **(b)** AN-CFD-1. **(c)** AN-CFD-2. **(d)** AN-CFD-3.



(a) (b) (c)

Figure 16: Normalized error $E_{\boldsymbol{u}}$ computed on different sections. **(a)** AN-CFD-1. **(b)** AN-CFD-2. **(c)** AN-CFD-3.

to the aneurysm domain. The 4D flow MRI dataset is very noisy and dispersed at the boundary. The measurement points cross the lateral boundaries, while at the bottom of the aneurysm sac there is a lack of data, making the boundary very difficult to identify. We filter out the points outside of the domain and we report the resulting distribution of the velocity magnitudes over all time samples in Figure 17b. Most of the velocity magnitude samples have low values, close to 0.1 $\mathrm{m\,s^{-1}}$.

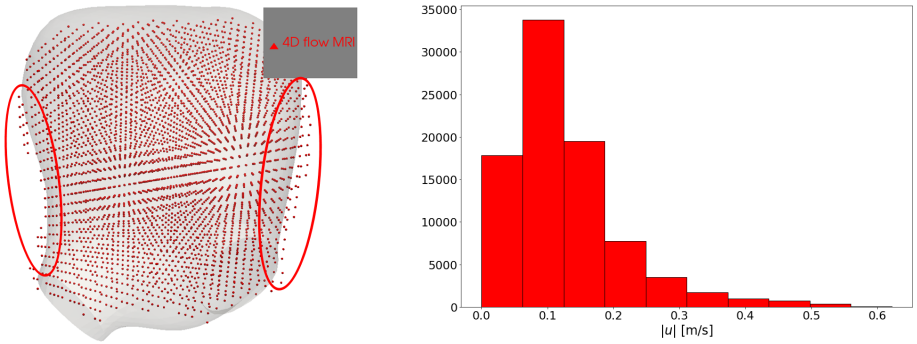
We train the PINNs following what we did in Section 3.2.1, using the loss function (13), where we neglect the inlet boundary loss term by setting $\lambda_{\mathrm{BC},1} = 0$, since no information is available on its spatiotemporal distribution. We use ANNs made of three hidden layers, with batch normalization before each layer.

We test four different PINN configurations at a single timestep corresponding to $t = 0.34$ s. Configurations AN-4DMRI-1, 2 and 3 correspond to decreasing weights on the data in the definition of the total loss function, whereas AN-4DMRI-4 disregards the regularization with the PDE residual $\boldsymbol{f}_{\mathrm{PDE}}$, and is used to assess the contribution of the wall no-slip boundary condition residual $\boldsymbol{f}_{\mathrm{BC},2}$. The parameters of the different PINN configurations are summarized in Table 5.

For AN-4DMRI-1, 2 and 3, after filtering out the points outside of the domain, we use $N_{\mathrm{train}} = 1107$ points for training and $N_{\mathrm{test}} = 2249$ points for testing. We randomly seed $N_{\mathrm{PDE}} = 4420$ collocation points for the PDE loss term and $N_{\mathrm{BC},2} = 3198$ for collocation points at the wall. Wall boundary condition and PDE collocation points are depicted in Figures 18a and 18b, respectively. For the fourth configuration, we use 80% of the total dataset for training, and the remaining for testing purposes.

For the training, we use 100 epochs of the ADAM optimizer, followed by 14 000 epochs of L-BFGS-B for the first three configurations and 7000 epochs for the fourth configuration.

Results. We report the reconstructed velocity fields from the PINNs along with the experimentally observed velocity field in Figure 19. All PINNs retrieved the main velocity field features, however, by decreasing the the weights on the data, the peak velocity decreases and the result matches the boundary conditions better,

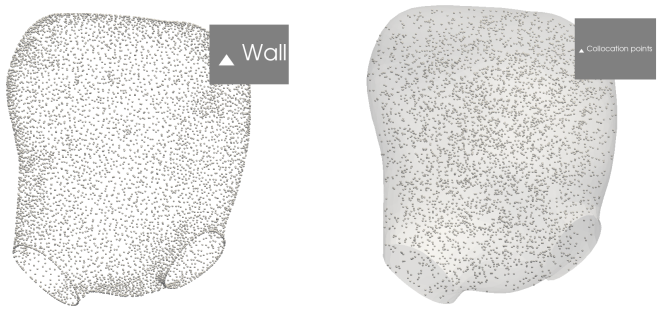


(a) (b)

Figure 17: Collocation points and velocity distribution for the 1.5 mm^3 voxel dataset. **(a)** Measurement points (red) and aneurysm domain (gray). Measurement points that cross the boundary are circled in red. **(b)** Recorded velocity magnitudes distribution.

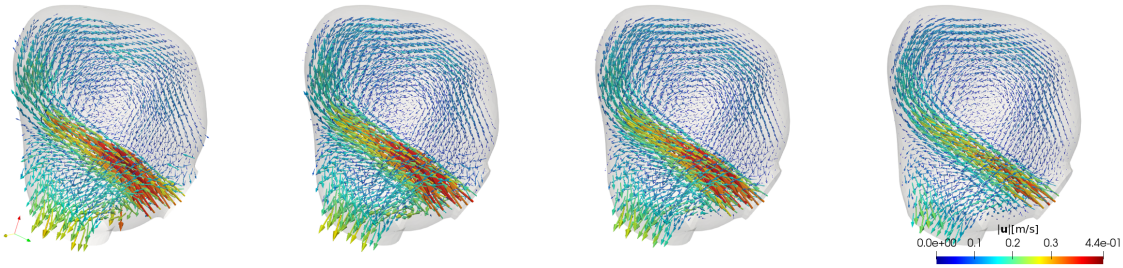
	AN-4DMRI-1	AN-4DMRI-2	AN-4DMRI-3	AN-4DMRI-4
λ_{u_1}	170	17	1.7	1.7
λ_{u_2}	170	17	1.7	1.7
λ_{u_3}	170	17	1.7	1.7
$\lambda_{\hat{\boldsymbol{u}}}$	17	1.7	0.17	0.17
$\lambda_{\text{BC},1}$	0	0	0	0
$\lambda_{\text{BC},2}$	21.6	21.6	21.6	36
$\lambda_{\text{BC},3}$	0.5	0.5	0.5	0
$\lambda_{\text{PDE},1}$	5.76	5.76	5.76	0
$\lambda_{\text{PDE},2}$	1	1	1	0
$\lambda_{\text{PDE},3}$	1	1	1	0
$\lambda_{\text{PDE},4}$	1.56	1.56	1.56	0
$\lambda_{\bar{p}}$	0.5	0.5	0.5	0
N_{neuron}	32	32	32	16

Table 5: Parameters of different loss functions used for PINNs training in Section 3.2.2.



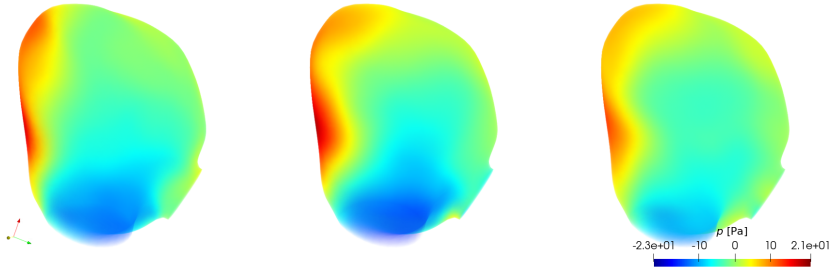
(a) (b)

Figure 18: Collocation points. **(a)** Collocation points at wall. **(b)** PDE collocation points.



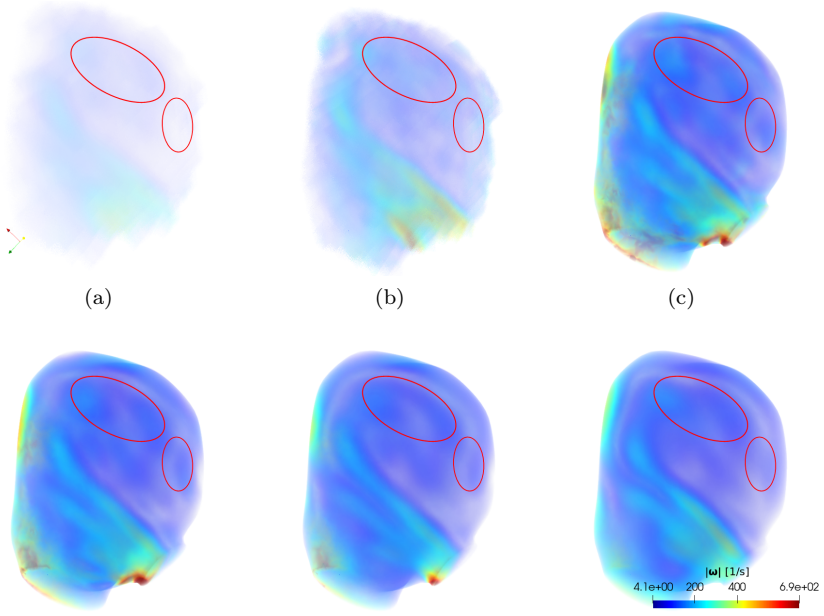
(a) (b) (c) (d)

Figure 19: Velocity field evaluated at the collocation points of the 1.5 mm^3 voxel dataset. **(a)** 4D flow MRI 1.5 mm^3 data. **(b)** AN-4DMRI-1. **(c)** AN-4DMRI-2. **(d)** AN-4DMRI-3.



(a) (b) (c)

Figure 20: Pressure field reconstruction. **(a)** AN-4DMRI-1. **(b)** AN-4DMRI-2. **(c)** AN-4DMRI-3.



(a) (b) (c)

(d) (e) (f)

Figure 21: Computed vorticity field for the 4D flow MRI data and for the PINN outputs. Recovered vorticity paths are circled in red. **(a)** 4D flow MRI 1.5 mm^3 data. **(b)** 4D flow MRI 1.0 mm^3 data. **(c)** AN-4DMRI-1. **(d)** AN-4DMRI-2. **(e)** AN-4DMRI-3. **(f)** AN-4DMRI-4.

	AN-4DMRI-1	AN-4DMRI-2	AN-4DMRI-3	AN-4DMRI-4
$\sqrt{\mathcal{L}^{\boldsymbol{u}}_{\text{train}}}$	$2.10 \cdot 10^{-2}$	$4.20 \cdot 10^{-2}$	$7.35 \cdot 10^{-2}$	$6.76 \cdot 10^{-2}$
$\sqrt{\mathcal{L}^{\boldsymbol{u}}_{\text{test}}}$	$4.21 \cdot 10^{-2}$	$4.83 \cdot 10^{-2}$	$6.40 \cdot 10^{-2}$	$5.39 \cdot 10^{-2}$
$e_{\boldsymbol{u}}$ $[\text{m}\,\text{s}^{-1}]$	$8.07 \cdot 10^{-3}$	$1.11 \cdot 10^{-2}$	$1.63 \cdot 10^{-2}$	$1.46 \cdot 10^{-2}$

Table 6: Computed errors for the four PINN configurations in Section 3.2.2.

particularly near the inlet, where the data are highly dispersed. We report the pressure reconstruction obtained from Eq. (2b) in Figure 20. From Figure 20b we see that all three pressure reconstructions have a similar pressure distribution. We remark that these represent non-invasive pressure measurements, as no pressure experimental data were introduced. Next, we compute the vorticity $\boldsymbol{\omega} = \nabla \times \boldsymbol{u}$, and the wall shear stress **WSS** for the reconstructed solutions. In Figure 21, we report the computed vorticities for the 4D flow MRI data and for the PINN outputs. AN-4DMRI-1, AN-4DMRI-2 and AN-4DMRI-3 all have a high value of the vorticity close to the inlet, the outlet and the left wall. Moreover, they all appear to have an overshoot of the vorticity near the inlet. As already discussed, the loss of accuracy at the inlet may be due to the fact that the Dirichlet inlet boundary condition is not available and the data at the inlet is scarce. We also see that using PINNs we are able to identify some vorticity paths that are not visible in the initial training data. Indeed, the ANN was trained using the 1.5 mm^3 voxel dataset, whereas the retrieved vorticity field is closer to the 1.0 mm^3 voxel data. This can be better appreciated for the AN-4DMRI-1 and AN-4DMRI-2 configurations.

Our analysis can be extended to different timesteps. To quantify the variation of the flow properties during a single heartbeat, we introduce the kinetic energy E_k and the enstrophy $E_{\boldsymbol{\omega}}$, both dimensional integral quantities computed as:

$$E_k = \int_\Omega \frac{1}{2}\rho|\boldsymbol{u}|^2, \qquad E_{\boldsymbol{\omega}} = \int_\Omega \frac{1}{2}\rho|\boldsymbol{\omega}|^2,$$

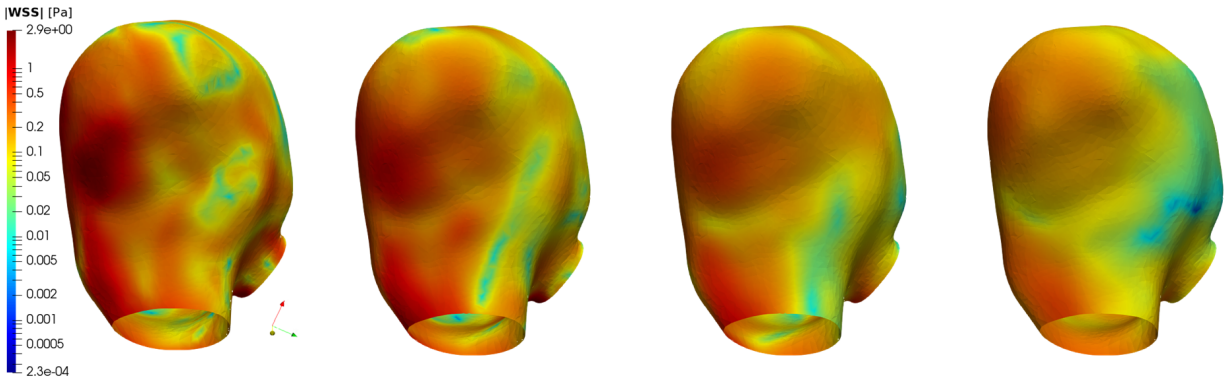
in dimensional units. In Figure 23, we report the variation in time of the integral quantities associated to the flow. The flow field close to the inlet is disregarded during the integration for the enstrophy computation, due to the strong overshoots in the observed vorticity. The ANN models were trained at time instances of $t = 0.136$ s, $t = 0.34$ s, $t = 0.544$ s and $t = 0.748$ s. In Figure 23a, we show the variation of the kinetic energy during the course a single heartbeat. While AN-4DMRI-1 gives larger estimates of the kinetic energy, and AN-4DMRI-3 strongly underestimates it, AN-4DMRI-2 provides a good compromise between the two and overall a better estimate over the duration of the whole heartbeat. In Figure 23b we report the enstrophy, which increases when going from the 1.5 mm^3 to the 1.0 mm^3 voxel dataset. On the other hand, the enstrophy computed from the PINN outputs decreases with decreasing weights on the data, i.e. from AN-4DMRI-1 to AN-4DMRI-3, in agreement with the distribution of the vorticity reported in Figure 21. We remark that the high dispersion of the enstrophy is in part due to errors in the computation of the vorticity, since it is performed with discrete derivatives.

We report in Figure 22 the computed WSS. In all the networks, the highest WSS is concentrated in the same wall region. However, AN-4DMRI-4 strongly underestimates the wall shear stress, with an order of magnitude of difference. The wall shear stress prediction decreases from AN-4DMRI-1 to AN-4DMRI-3. Moreover, an overshoot of the wall shear stress appears near the inlet. As discussed for the vorticity, this may be due to the scarcity of data. The results of the wall shear stress are poor and difficult to assess, since so much dispersion arises in our study.

Lastly, we compute the errors as the square root of the training and test loss function for the velocity at $t = 0.34$ s. We compute an average absolute error for the 1.5 mm^3 voxel dataset. The results are summarized in Table 6.

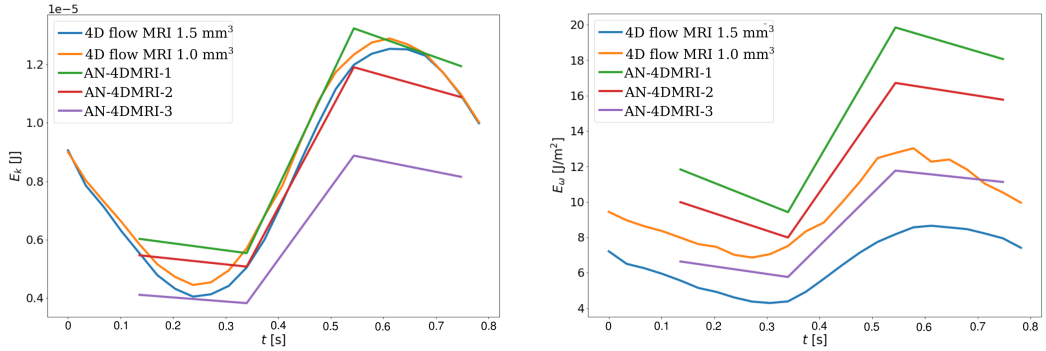
4. Discussion

This study demonstrates the potential of PINNs to integrate sparse, noisy, or under-resolved experimental data with physical modeling, enabling more accurate and spatially resolved reconstruction of hemodynamic



(a) (b) (c) (d)

Figure 22: Computed wall shear stress magnitude in log scale. **(a)** AN-4DMRI-1. **(b)** AN-4DMRI-2. **(c)** AN-4DMRI-3. **(d)** AN-4DMRI-4.



(a) (b)

Figure 23: Integral quantities during a single heartbeat. **(a)** Kinetic energy. **(b)** Enstrophy.

quantities. Our focus was on enhancing 4D flow MRI data, which, while non-invasive, suffers from limited spatial and temporal resolution, especially near vessel walls and in regions of complex flow. By embedding the incompressible Navier–Stokes equations into the PINNs, we leveraged both data and physical laws to improve the reconstruction the velocity field.

The velocity, pressure, and WSS reconstruction have shown strong agreements both with FEM simulations and with experimental data for the FDA nozzle benchmark. Initially, training the ANN using only data without physics-based regularization yielded poor results, particularly when data were limited or spatially under-resolved. This limitation was addressed by incorporating a the Navier-Stokes residuals into the loss function, which acted as a regularizer and compensated for the sparse nature of the data. The physics-based regularization not only improved the accuracy of the reconstructed fields but also enabled the estimation of quantities not directly measured, such as the pressure and WSS.

One of the main challenges in PINN training was the a priori selection of optimal hyperparameters for the numerical optimization. This process can be automated using software libraries for hyperparameter tuning, such as `scikit-learn`'s `GridSearchCV` [56]. Another significant challenge was the minimization of loss terms spanning multiple orders of magnitude. We mitigated this by introducing a logarithmic transformation of the residuals in the loss function, which improved the network's ability to capture both high- and low-magnitude flow features.

The placement and density of collocation points also played a crucial role. Increasing the number of collocation points in regions of high flow gradients, such as the convergent section of the nozzle, enhanced

the training process and the accuracy of the results. Furthermore, incorporating velocity unit vectors and increasing the number of neurons in the ANN improved the reconstruction without inducing overfitting. When trained with PIV measurements, PINNs provided results that more closely matched experimental observations compared to classical CFD simulations, highlighting the robustness of the physics-informed approach.

For the aneurysm case, PINNs demonstrated the ability to compute key hemodynamic quantities of interest, such as velocity, pressure and WSS, from 4D flow MRI data. The integration of physical modeling was particularly valuable in regions where data were sparse or noisy, such as near vessel walls. The reconstructed velocity fields were more accurate and the vorticity distributions better matched high-resolution 4D flow MRI data compared to pure data-driven approaches. While the estimation of WSS remains challenging due to its high sensitivity to boundary conditions and data quality, the physics-based regularization was essential for obtaining non-invasive pressure information and improving the overall fidelity of the hemodynamic indicators.

4.1. Limitations

While the proposed procedure can be easily extended to other applications, the present analysis is limited to two specific test cases: the FDA nozzle benchmark and an in vitro aneurysm model.

Regarding the FDA nozzle benchmark, only the laminar flow regime was considered. However, transitional and turbulent flow regimes, which are common in physiological scenarios, present more complex fluid flow structures and were not explored in this work. Future studies should investigate the PINNs performance under these conditions.

The present analysis also relies on a quasi-steady flow assumption to reduce the number of parameters in the PINNs. Although the aneurysm case study incorporates pulsatile flow, the framework's ability to accurately capture rapid, unsteady flow features, such as those occurring during systolic peaks, has not been thoroughly validated. The quasi-steady assumption may not hold in highly dynamic scenarios, potentially limiting the accuracy of the results.

The aneurysm case study relied on in vitro 4D flow MRI data which are inherently less noisy and more controlled than in vivo acquisitions. In vivo 4D flow MRI data are often affected by significant noise, motion artifacts and measurement errors. The robustness of the proposed approach in handling such noisy, real-world data remains to be thoroughly tested. Additionally, the lack of a ground truth for the aneurysm model limited the ability to validate the reconstructed quantities, particularly the wall shear stress (WSS), which exhibited high variability and dispersion in the results.

The accuracy of the PINN also depends on the correct specification of the boundary conditions. In the aneurysm case study, the lack of precise inlet velocity profiles or outlet pressure conditions introduced uncertainties, particularly near the boundaries. This limitation is exacerbated in patient-specific geometries, where boundary conditions are often approximated or unknown.

The performance of the PINN framework is highly sensitive to the selection of hyperparameters, including the weights assigned to different terms in the loss function and the number and placement of collocation points for the PDE regularization. Balancing data-driven and physics-based loss terms is non-trivial and often requires extensive trial and error. Suboptimal weights can lead to poor convergence, nonphysical solutions, or overfitting, as observed in some of the test cases. Due to the manual optimization of the PINN hyperparameters, there is no guarantee of their optimality. Further works could explore the integration of automated optimization tools into the current framework and strategies for optimal collocation point placement, particularly in complex geometries where flow features are not known a priori.

5. Conclusion

In this work, we developed and tested a PINN-based method to enhance measurement-based data, with a primary emphasis on improving data from 4D flow MRI. Our findings highlight the value of incorporating physical knowledge with sparse or noisy measurements, enabling the reconstruction of high-fidelity hemodynamic fields and estimation of quantities not directly measured, such as the pressure and the WSS.

For the FDA nozzle benchmark, we successfully combined experimental observations with the Navier-Stokes equations, demonstrating that physics-based regularization improves the reconstruction of velocity and pressure fields from localized samples. Training with PIV measurements confirmed that PINNs can achieve good agreement with experimental data, outperforming classical CFD approaches.

In the aneurysm case, PINNs improved the reconstruction of velocity fields and vorticity distributions, and enabled non-invasive pressure estimation from 4D flow MRI data. While the WSS estimation remains challenging, the integration of physical modeling was crucial for obtaining meaningful results in data-sparse regions in the study of the aneurysm.

Advancements in machine learning techniques, such as adaptive activation functions [57], or enhanced 4D flow MRI preprocessing, as for instance using a singular value decomposition [58], could further enhance accuracy. Overall, we have shown PINNs to be a valuable tool for enhancing data – particularly for emerging technologies such as 4D flow MRI – by effectively combining sparse measurements with physical modeling.

Acknowledgements

LD and IR acknowledge the support by the FAIR (Future Artificial Intelligence Research) project, funded by the NextGenerationEU program within the PNRR-PE-AI scheme (M4C2, investment 1.3, line on Artificial Intelligence), Italy. LD, SP, and IR acknowledge their membership to INdAM GNCS – Gruppo Nazionale per il Calcolo Scientifico (National Group for Scientific Computing), Italy. The research of LD, SP, and IR is part of the activities of "Dipartimento di Eccellenza 2023–2027", MUR, Italy, Dipartimento di Matematica, Politecnico di Milano.

Appendix A. Appendix A

Appendix A.1. Training sections coordinates

We report the z coordinates of the training cross-sections for the FDA nozzle benchmark training by in silico data in Section 3.1.1 and tests in Sections 3.1.1, 3.1.1, 3.1.1, 3.1.1, and 3.1.1.

Figure 2a z coords [m]	-0.179, -0.145, -0.108, -0.07, -0.057, -0.039, -0.016, 0.058, 0.099, 0.141

Table A.7: z coordinates for the training sections from FEM simulation.

Appendix A.2. FEM simulation parameters

We report the FEM simulation parameters from Sections 3.1.1 and 3.2.1.

	FDA-coarse	FDA-fine	ANE-coarse	ANE-fine
Time step [s]	10^{-3}	10^{-3}	10^{-3}	10^{-3}
FE spaces	$\mathbb{Q}_1 - \mathbb{Q}_1$	$\mathbb{Q}_1 - \mathbb{Q}_1$	$\mathbb{P}_1 - \mathbb{P}_1$	$\mathbb{P}_1 - \mathbb{P}_1$
Number of cells	7325	36 128	133 166	1 242 082
Velocity DOFs	24 696	116 442	78 621	632 682
Pressure DOFs	8232	38 814	26 207	210 894
Total DOF	32 928	155 256	104 828	843 576
h_{min} [m]	$2.31 \cdot 10^{-3}$	$1.23 \cdot 10^{-3}$	$3.87 \cdot 10^{-4}$	$1.88 \cdot 10^{-4}$
h_{max} [m]	$8.38 \cdot 10^{-3}$	$4.10 \cdot 10^{-3}$	$1.14 \cdot 10^{-3}$	$6.32 \cdot 10^{-4}$
h_{mean} [m]	$6.20 \cdot 10^{-3}$	$3.10 \cdot 10^{-3}$	$7.93 \cdot 10^{-4}$	$3.76 \cdot 10^{-4}$

Table A.8: Simulation parameters for the FDA nozzle benchmark problem for the coarse and fine hexahedral meshes.

References

[1] Z. Stankovic, B. D. Allen, J. Garcia, K. B. Jarvis, M. Markl, 4D flow imaging with MRI, Cardiovascular Diagnosis and Therapy 4 (2014) 173.

[2] S. Miyazaki, K. Itatani, T. Furusawa, T. Nishino, M. Sugiyama, Y. Takehara, S. Yasukochi, Validation of numerical simulation methods in aortic arch using 4D Flow MRI, Heart and Vessels 32 (2017) 1032–1044.

[3] K. Sughimoto, Y. Shimamura, C. Tezuka, K. Tsubota, H. Liu, K. Okumura, Y. Masuda, H. Haneishi, Effects of arterial blood flow on walls of the abdominal aorta: distributions of wall shear stress and oscillatory shear index determined by phase-contrast magnetic resonance imaging, Heart and Vessels 31 (2016) 1168–1175.

[4] M. Markl, A. Frydrychowicz, S. Kozerke, M. Hope, O. Wieben, 4D flow MRI, Journal of Magnetic Resonance Imaging 36 (2012) 1015–1036.

[5] B. Casas, J. Lantz, P. Dyverfeldt, T. Ebbers, 4D flow MRI-based pressure loss estimation in stenotic flows: Evaluation using numerical simulations, Magnetic Resonance in medicine 75 (2016) 1808–1821.

[6] H. Baek, M. V. Jayaraman, G. E. Karniadakis, Wall shear stress and pressure distribution on aneurysms and infundibulae in the posterior communicating artery bifurcation, Annals of Biomedical Engineering 37 (2009) 2469–2487.

[7] C. G. Caro, Discovery of the role of wall shear in atherosclerosis, Arteriosclerosis, Thrombosis, and Vascular Biology 29 (2009) 158–161.

[8] A. Zingaro, L. Dedè, F. Menghini, A. Quarteroni, Hemodynamics of the heart's left atrium based on a Variational Multiscale-LES numerical method, European Journal of Mechanics-B/Fluids 89 (2021) 380–400.

[9] A. Harloff, et al., 3D blood flow characteristics in the carotid artery bifurcation assessed by flow-sensitive 4D MRI at 3T, Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine 61 (2009) 65–74.

[10] M. T. Ngo, C. I. Kim, J. Jung, G. H. Chung, D. H. Lee, H. S. Kwak, Four-dimensional flow magnetic resonance imaging for assessment of velocity magnitudes and flow patterns in the human carotid artery bifurcation: Comparison with computational fluid dynamics, Diagnostics 9 (2019) 223.

[11] M. S. Pravdivtseva, F. Gaidzik, P. Berg, P. Ulloa, N. Larsen, O. Jansen, J.-B. Hövener, M. Salehi Ravesh, Influence of spatial resolution and compressed sense acceleration factor on flow quantification with 4D Flow MRI at 3 Tesla, Tomography 8 (2022) 457–478.

[12] A. Zingaro, I. Fumagalli, L. Dede', M. Fedele, P. C. Africa, A. F. Corno, A. Quarteroni, A geometric multiscale model for the numerical simulation of blood flow in the human left heart, Discrete and Continuous Dynamical Systems - S 15 (2022) 2391–2427. doi:10.3934/dcdss.2022052.

[13] K. Futami, T. Uno, K. Misaki, S. Tamai, I. Nambu, N. Uchiyama, M. Nakada, Identification of vortex cores in cerebral aneurysms on 4D flow MRI, American Journal of Neuroradiology 40 (2019) 2111–2116.

[14] T. B. Le, D. R. Troolin, D. Amatya, E. K. Longmire, F. Sotiropoulos, Vortex phenomena in sidewall aneurysm hemodynamics: experiment and numerical simulation, Annals of Biomedical Engineering 41 (2013) 2157–2170.

[15] A. Quarteroni, Numerical Models for Differential Problems, volume 16 of *Modeling, Simulation and Applications*, 3rd ed., Springer International Publishing, Cham, Switzerland, 2017.

[16] A.-F. Totorean, I.-C. Totorean, S. I. Bernad, T. Ciocan, D. C. Malita, D. Gaita, E. S. Bernad, Patient-Specific Image-Based Computational Fluid Dynamics Analysis of Abdominal Aorta and Branches, Journal of Personalized Medicine 12 (2022) 1502. URL: https://www.mdpi.com/2075-4426/12/9/1502. doi:10.3390/jpm12091502.

[17] I. Goodfellow, Y. Bengio, A. Courville, Deep Learning, MIT press, 2016.

[18] T. Tassi, A. Zingaro, L. Dede', A machine learning approach to enhance the SUPG stabilization method for advection-dominated differential problems, Mathematics in Engineering 5 (2023) 1–26. doi:10.3934/mine.2023032.

[19] A. Quarteroni, P. Gervasio, F. Regazzoni, Combining physics-based and data-driven models: advancing the frontiers of research with scientific machine learning, Mathematical Models and Methods in Applied Sciences 35 (2025) 905–1071. URL: https://www.worldscientific.com/doi/10.1142/S0218202525500125. doi:10.1142/S0218202525500125.

[20] J. D. Toscano, V. Oommen, A. J. Varghese, Z. Zou, N. Ahmadi Daryakenari, C. Wu, G. E. Karniadakis, From PINNs to PIKANs: recent advances in physics-informed machine learning, Machine Learning for Computational Science and Engineering 1 (2025) 15. URL: https://link.springer.com/10.1007/s44379-025-00015-1. doi:10.1007/s44379-025-00015-1.

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

[22] N. K. Schiavone, P. J. Nair, C. J. Elkins, D. B. McElhinney, D. B. Ennis, J. K. Eaton, A. L. Marsden, Assessing the impact of cardiac output and valve orientation on bioprosthetic pulmonary valve hemodynamics using in vitro 4d-flow mri and high-speed imaging, Cardiovascular Engineering and Technology 16 (2025) 138–153. URL: https://doi.org/10.1007/s13239-024-00762-x. doi:10.1007/s13239-024-00762-x.

[23] J. Zhang, M. C. Brindise, S. Rothenberger, S. Schnell, M. Markl, D. Saloner, V. L. Rayz, P. P. Vlachos, 4D flow MRI pressure estimation using velocity measurement-error-based weighted least-squares, IEEE transactions on medical imaging 39 (2019) 1668–1680.

[24] E. Ferdian, D. J. Dubowitz, C. A. Mauger, A. Wang, A. A. Young, WSSNet: aortic wall shear stress estimation using deep learning on 4D flow MRI, Frontiers in Cardiovascular Medicine (2022) 1969.

[25] J. Zhang, S. M. Rothenberger, M. C. Brindise, M. B. Scott, H. Berhane, J. J. Baraboo, M. Markl, V. L. Rayz, P. P. Vlachos, Divergence-Free Constrained Phase Unwrapping and Denoising for 4D Flow MRI Using Weighted Least-Squares, IEEE Transactions on Medical Imaging 40 (2021) 3389–3399. URL: https://ieeexplore.ieee.org/document/9446842/. doi:10.1109/TMI.2021.3086331.

[26] S. Cai, Z. Mao, Z. Wang, M. Yin, G. E. Karniadakis, Physics-informed neural networks (PINNs) for fluid mechanics: A review, Acta Mechanica Sinica (2022) 1–12.

[27] 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, Computer Methods in Applied Mechanics and Engineering 358 (2020) 112623.

[28] M. Sarabian, H. Babaee, K. Laksari, Physics-informed neural networks for brain hemodynamic predictions using medical imaging, IEEE Transactions on Medical Imaging (2022).

[29] S. Cai, Z. Wang, F. Fuest, Y. J. Jeon, C. Gray, G. E. Karniadakis, Flow over an espresso cup: inferring 3-D velocity and pressure fields from tomographic background oriented Schlieren via physics-informed neural networks, Journal of Fluid Mechanics 915 (2021) A102. URL: https://www.cambridge.org/core/product/identifier/S002211202100135X/type/journal_article. doi:10.1017/jfm.2021.135.

[30] N. Hub, Benchmark dataset for validating computational fluid dynamic (CFD) simulation of blood flow through FDA nozzle and FDA blood pump — NCI Hub, 2022. URL: https://ncihub.org/wiki/FDA_CFD?version=30, [Online; accessed March 2022].

[31] S. F. Stewart, E. G. Paterson, G. W. Burgreen, P. Hariharan, M. Giarra, V. Reddy, S. W. Day, K. B. Manning, S. Deutsch, M. R. Berman, et al., Assessment of CFD performance in simulations of an idealized medical device: results of FDA's first computational interlaboratory study, Cardiovascular Engineering and Technology 3 (2012) 139–160.

[32] P. Hariharan, M. Giarra, V. Reddy, S. W. Day, K. B. Manning, S. Deutsch, S. F. Stewart, M. R. Myers, M. R. Berman, G. W. Burgreen, et al., Multilaboratory particle image velocimetry analysis of the fda benchmark nozzle model to support validation of computational fluid dynamics simulations, Journal of Biomechanical Engineering 133 (2011).

[33] P. Hariharan, R. A. Malinauskas, Round robin 1 data sets, 2017. URL: https://ncihub.org/publications/43/2. doi:doi:/10.17917/C78G69.

[34] A. Quarteroni, L. Dede', A. Manzoni, C. Vergara, et al., Mathematical modelling of the human cardiovascular system: data, numerical approximation, clinical applications, volume 33, Cambridge University Press, 2019.

[35] R. Bischof, M. A. Kraus, Multi-Objective Loss Balancing for Physics-Informed Deep Learning, Computer Methods in Applied Mechanics and Engineering 439 (2025) 117914. URL: https://linkinghub.elsevier.com/retrieve/pii/S0045782525001860. doi:10.1016/j.cma.2025.117914.

[36] G. Kissas, E. Hwuang, E. W. Thompson, N. Schwartz, J. A. Detre, W. R. Witschey, P. Perdikaris, Feasibility of vascular parameter estimation for assessing hypertensive pregnancy disorders, Journal of Biomechanical Engineering 144 (2022) 121011.

[37] F. Chollet, Deep Learning with Python, Simon and Schuster, 2021.

[38] F. Regazzoni, S. Pagani, A. Quarteroni, Universal Solution Manifold Networks (USM-Nets): non-intrusive mesh-free surrogate models for problems in variable domains, 2022. URL: https://arxiv.org/abs/2204.07805. doi:10.48550/ARXIV.2204.07805, version Number: 1.

[39] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, CoRR abs/1412.6980 (2015).

[40] C. Zhu, R. H. Byrd, P. Lu, J. Nocedal, Algorithm 778: L-BFGS-B: Fortran subroutines for large-scale bound-constrained optimization, ACM Transactions on mathematical software (TOMS) 23 (1997) 550–560.

[41] P. C. Africa, life^{x}: a flexible, high performance library for the numerical solution of complex finite element problems, SoftwareX 20 (2022) 101252.

[42] P. C. Africa, I. Fumagalli, M. Bucelli, A. Zingaro, M. Fedele, L. Dede', A. Quarteroni, lifex-cfd: An open-source computational fluid dynamics solver for cardiovascular applications, Computer Physics Communications 296 (2024) 109039. URL: https://linkinghub.elsevier.com/retrieve/pii/S0010465523003843. doi:10.1016/j.cpc.2023.109039.

[43] D. Arndt, W. Bangerth, D. Davydov, T. Heister, L. Heltai, M. Kronbichler, M. Maier, J.-P. Pelteret, B. Turcksin, D. Wells, The deal.II finite element library: Design, features, and insights, Computers & Mathematics with Applications 81 (2021) 407–422. URL: https://linkinghub.elsevier.com/retrieve/pii/S0898122120300894. doi:10.1016/j.camwa.2020.02.022.

[44] P. C. Africa, D. Arndt, W. Bangerth, B. Blais, M. Fehling, R. Gassmöller, T. Heister, L. Heltai, S. Kinnewig, M. Kronbichler, M. Maier, P. Munch, M. Schreter-Fleischhacker, J. P. Thiele, B. Turcksin, D. Wells, V. Yushutin, The deal.II library, Version 9.6, Journal of Numerical Mathematics 32 (2024) 369–380. URL: https://www.degruyter.com/document/doi/10.1515/jnma-2024-0137/html. doi:10.1515/jnma-2024-0137.

[45] D. Forti, L. Dedè, Semi-implicit BDF time discretization of the Navier–Stokes equations with VMS-LES modeling in a high performance computing framework, Computers & Fluids 117 (2015) 168–182.

[46] Hardware — mox.polimi.it, 2022. URL: https://mox.polimi.it/research-areas/hpcmox/hardware/, [Accessed 13-Aug-2022].

[47] D. V. Dung, N. D. Song, P. S. Palar, L. R. Zuhal, On The Choice of Activation Functions in Physics-Informed Neural Network for Solving Incompressible Fluid Flows, in: AIAA SCITECH 2023 Forum, American Institute of Aeronautics and Astronautics, National Harbor, MD & Online, 2023. URL: https://arc.aiaa.org/doi/10.2514/6.2023-1803. doi:10.2514/6.2023-1803.

[48] A. Géron, Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems, O'Reilly Media, Inc., 2019.

[49] S. Ioffe, C. Szegedy, Batch normalization: Accelerating deep network training by reducing internal covariate shift, in: International conference on machine learning, PMLR, 2015, pp. 448–456.

[50] X. Glorot, Y. Bengio, Understanding the difficulty of training deep feedforward neural networks, in: Proceedings of the thirteenth international conference on artificial intelligence and statistics, JMLR Workshop and Conference Proceedings, 2010, pp. 249–256.

[51] S. Raschka, Python Machine Learning, Packt publishing ltd, 2015.

[52] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, et al., TensorFlow: a system for Large-Scale machine learning, in: 12th USENIX symposium on operating systems design and implementation (OSDI 16), 2016, pp. 265–283.

[53] F. Chollet, et al., Keras, https://keras.io, 2015.

[54] J. Ahrens, B. Geveci, C. Law, Paraview: An end-user tool for large data visualization, The visualization handbook 717 (2005).

[55] M. S. Pravdivtseva, E. Peschke, T. Lindner, F. Wodarg, J. Hensler, D. Gabbert, I. Voges, P. Berg, A. J. Barker, O. Jansen, et al., 3D-printed, patient-specific intracranial aneurysm models: From clinical data to flow experiments with endovascular devices, Medical Physics 48 (2021) 1469–1484.

[56] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, Édouard Duchesnay, Scikit-learn: Machine learning in python, Journal of Machine Learning Research 12 (2011) 2825–2830. URL: http://jmlr.org/papers/v12/pedregosa11a.html.

[57] A. D. Jagtap, K. Kawaguchi, G. E. Karniadakis, Adaptive activation functions accelerate convergence in deep and physics-informed neural networks, Journal of Computational Physics 404 (2020) 109136.

[58] E. Henry, J. Hofrichter, [8] singular value decomposition: Application to analysis of experimental data, in: Methods in enzymology, volume 210, Elsevier, 1992, pp. 129–192.

MOX Technical Reports, last issues

Dipartimento di Matematica
Politecnico di Milano, Via Bonardi 9 - 20133 Milano (Italy)

62/2026 Ciaramella, G.; Gong, W.; Kwok, F.; Tan, Z.
On the Uniform Convergence Analysis of the Schwarz Alternating Method for Optimal Control Problems

60/2026 Ciaramella, G.; Gander, M.J.; Van Criekingen, S.; Vanzan, T.
Algebraic and Two-Level Parallel Substructured Schwarz Methods

63/2026 Ciaramella, G.; Gong, W.; Kwok, F., Tan, Z.
Uniform Convergence of the Schwarz Alternating Method for Optimal Control Problems

58/2026 Mapelli, A.; Massi, M.C.; Cuccuru, G.; Di Angelantonio, E.; Ieva, F.
Prior-informed conditional Gaussian graphical models: an application to protein interaction network reconstruction

57/2026 Fontana, N.; Secchi, P.; Di Angelantonio, E.; Ieva, F.
Modeling time-varying genetic effects on binary disease risk via functional Mendelian Randomization

56/2026 Botta, P.; Vitullo, P.; Ventimiglia, T.; Linninger, A.; Zunino, P.
Physics-Informed Learning of Microvascular Flow Models using Graph Neural Networks

51/2026 Bellezza P.; Ciaramella G.; Macchini C.; Mazzieri I.; Verani M.
ParaFlow: Parareal Acceleration of Gradient-Flow Minimization

53/2026 Dong Z., Jiang Y., Ng M., Ciaramella G., Yin J.
Chebyshev-Filtered Randomized Low-rank Preconditioners for Symmetric Positive Definite Linear Systems

55/2026 Beirao da Veiga, L.; Canuto, C.; Nochetto, R.H.; Vacc, G ; Verani, M.
A Virtual Element Method for elliptic problems on trimmed background meshes

54/2026 Antonietti, P. F.; Corti, M.; Leimer Saglio, C. B.; Pagani, S.
The lymph 2.0 library: p-adaptive algorithms and parallel assembly strategies for polytopal DG methods