Physics Informed Neural Network Framework for Full Order and Reduced Order External CFD Solver

Rahul Halder*
Mathematics Area, mathLab, SISSA, via Bonomea 265, I-34136 Trieste, Italy
rhalder@sissa.it

Giovanni Stabile
The Biorobotics Institute, Sant'Anna School of Advacended Studies
V.le R. Piaggio 34, 56025, Pontedera, Pisa - Italy
giovanni.stabile@santannapisa.it

Gianlugi Rozza
Mathematics Area, mathLab, SISSA, via Bonomea 265, I-34136 Trieste, Italy
grozza@sissa.it

Abstract

This work describes the development of a physics-informed neural network (PINN) with a physics-based loss term derived from an external Computational Fluid Dynamics (CFD) solver such as OpenFOAM. The major difficulty in coupling PINN with external forward solvers is the inability to access the discretized form of the governing equation directly from the PINN. This creates a significant challenge to conventional automatic-differentiation-based derivative computation of physics-based loss terms with respect to the weight matrix and bias vectors in neural networks. Therefore, we propose modifying the physics-based loss term to compute the derivative required for the optimization machinery. Furthermore, to overcome the problem arising from the large dimensionality, the governing equations are cast in the linear manifold Stabile and Rozza [2018], and the resulting reduced form of the equation is used as a residual for the physics-based loss term in the PINN. The main objective of the current work is to couple the available numerical data with existing forward solvers (in full-order and reduced-order forms) and use them for several inverse and ill-posed problems. The major potential of the current work lies in offloading the task of residual computation, boundary, and initial condition to dedicated external forward solvers such as OpenFOAM. The additional implementation of governing physics in the conventional PINN framework is no longer required. We show two applications of the coupling method, the Burgers' equation of full order implemented in OpenFOAM and the reduced-order incompressible viscous flow past a cylinder implemented in ITHACA-FV Stabile and Rozza [2018] with physics-informed neural network framework.

1 Introduction

Early attempts to solve partial differential equations (PDEs) using neural networks began with the work of Dissanayake and Phan-Thien [1994]. Recent advancements in neural network research have revitalized this area, as seen in the work of Raissi et al. [2019]. Physics-informed neural networks (PINNs) address the challenge of large training dataset requirements by incorporating additional physics-based loss terms, which are derived from the governing equations. Various types of neural

*Corresponding author: rhalder@sissa.it

1st Conference on Applied AI and Scientific Machine Learning (CASML 2024).

networks have been explored in the context of PINNs. Artificial neural networks (ANNs) have been used by Raissi et al. [2019], bin Waheed et al. [2021], Tartakovsky et al. [2020], and Schiassi et al. [2021]. Similarly, convolutional neural networks (CNNs) have been coupled with physics constraints by Gao et al. [2021] and Fang and Zhan [2019]. An extensive review of advancements in PINNs has been provided by Cuomo et al. [2022]. Cheng and Zhang [2021] introduced a tensor differentiator to calculate derivatives of state variables, coupling LSTM networks with physics from governing equations for applications in nonlinear structural problems. Typically, PINNs rely on automatic differentiation (AD) for gradient computations Baydin et al. [2018]. However, some efforts have utilized numerical methods such as finite difference, finite element, and finite volume techniques to compute the physics-based loss terms. For instance, Ranade et al. [2021] proposed Discretization-net, incorporating finite volume-based residuals into the PINN loss term. Similarly, Aulakh et al. [2022] integrated a physics-informed ANN with finite volume discretization using the OpenFOAM solver. Despite their benefits, PINNs often require more computation time than traditional solvers due to the high dimensionality of the network output. To address this, conventional dimensionality reduction techniques, such as Proper Orthogonal Decomposition (POD), can be employed to simplify the governing equations before coupling them with the loss terms, as demonstrated by Chen et al. [2021] and Hijazi et al. [2023]. The full potential of discretized physics-based neural networks could be realized by coupling PINN solvers with external forward solvers, each implemented in distinct coding environments.

The contributions of this work are summarized as follows:

- A novel approach is proposed to integrate existing forward solvers into the PINN framework. In this work, OpenFOAM is coupled for the full-order system, and ITHACA-FV, a derivative of OpenFOAM for reduced-order modeling Stabile and Rozza [2018], is integrated with the PINN framework.
- An enhanced Physics-Informed Neural Network (PINN) algorithm is introduced (detailed in Halder et al. [2023]), which includes an additional step to enable seamless integration with external solvers. This eliminates the need to embed the discretized governing equations directly within the computational graph of the neural network.

2 Governing Equations and Reduced Order Representations

The governing equations associated with different computational mechanics' problems can be cast into the general form of a nonlinear parameterized dynamical system as shown in Equation 1

$$\frac{\partial u(t;\mu)}{\partial t} = \mathbf{A}(\mu)u(t;\mu) + \mathbf{F}(u;\mu;t)u(t;\mu) + \mathbf{B}(\mu)f(\mu), \;\; u(0;\mu) = u^0(\mu), \tag{1}$$

Let $\mu \in \Omega_\mu \subset \mathbb{R}^{N_\mu}$ represent a vector containing the parameters of the dynamical system. The time-dependent state variable is denoted by $u : [0, T] \times \mathbb{R}^{N_\mu} \to \mathbb{R}^N$, while the initial state is given by $u^0 : \mathbb{R}^{N_\mu} \to \mathbb{R}^N$. The function $f : [0, T] \times \mathbb{R}^{N_\mu} \to \mathbb{R}^{N_i}$ describes the time-dependent input variable, which is independent of the state variable. Here, N, N_i, and N_μ represent the number of degrees of freedom of the high-fidelity solution, the dimension of the input variable, and the parameter space, respectively. The linear component of the governing equation is defined by the constant operator $\mathbf{A} : \mathbb{R}^{N_\mu} \to \mathbb{R}^{N\times N}$, while the nonlinear component is characterized by the operator $\mathbf{F} : \mathbb{R}^{N_\mu} \to \mathbb{R}^{N\times N}$, which depends on the state variable. Additionally, the operator $\mathbf{B} : \mathbb{R}^{N_\mu} \to \mathbb{R}^{N_i\times N_i}$, associated with the input variable, may also depend on the state variable depending on the nature of the governing equation. Although any time integration method can be used to discretize the system in time, we demonstrate the backward Euler time-discretization with an implicit approach for illustrative purposes. For simplicity, the $\mathbf{B}$ matrix is assumed to be independent of the state variable in this demonstration.

$$(I_N - \Delta t^{(k)}A(\mu) - \Delta t^{(k)}F(u^{(k)};\mu;t))u^{(k)} = u^{(k-1)} + \Delta t^{(k-1)}B(\mu;t)f(\mu), \tag{2}$$

whereas if we consider the explicit numerical discretization approach,

$$u^{(k)} = u^{(k-1)} + (\Delta t^{(k-1)}A(\mu) + \Delta t^{(k-1)}F(u^{(k-1)};\mu;t))u^{(k-1)} + \Delta t^{(k-1)}B(\mu;t)f(\mu), \tag{3}$$

where, I_N denotes the identity matrix of size $\mathbb{R}^{(N\times N)}$, Δt is the time step and and k denotes the k^{th} time instant. The following subsection will discuss the reduced-order formulation of the discretized governing equations.

2.1 Linear subspace solution representation

The computational complexity of Equation 2 and Equation 3 is $\mathcal{O}(N)$. To reduce the computational cost of solving the full-order system, the state variables can be approximated as a linear combination of a reduced number of basis vectors Φ. The resulting reduced-order equations can then be solved using a time-marching scheme similar to the full-order model. In section 3, we propose an alternative approach to solving the reduced-order equations in the context of physics-informed neural networks. The basis vectors can be derived from a set of solution vectors corresponding to different parameter values μ as follows:

$$u(\mu) \approx \tilde{u}(\mu) \equiv \Phi \hat{u}(\mu), \tag{4}$$

where $\hat{u}(\mu) : \mathbb{R}^{N_\mu} \to \mathbb{R}^n$ with $n \ll N$. The basis vectors $\Phi \in \mathbb{R}^{N\times n}$ are defined as:

$$\Phi = \begin{bmatrix} \phi_1 & \phi_2 & \dots & \phi_n \end{bmatrix}. \tag{5}$$

To compute the basis vectors, solution snapshots u are collected at various time instants for m parameter values $[\mu_1, \mu_2, \dots, \mu_m]$. At a specific parameter μ_p, the snapshot vectors for N_t time instants are organized as $U_p = [u^{(1)}(\mu_p), u^{(2)}(\mu_p), \dots, u^{(N_t)}(\mu_p)] \in \mathbb{R}^{N\times N_t}$. By concatenating snapshots across all parameter values, the total snapshot matrix $U \in \mathbb{R}^{N\times mN_t}$ is obtained as $U = [U_1 \quad U_2 \quad \dots \quad U_m]$.

Proper Orthogonal Decomposition (POD) Stabile and Rozza [2018] can be used to compute the spatial basis vectors Φ. This is done by performing Singular Value Decomposition (SVD) on the snapshot matrix U:

$$U = W\Sigma V^T = \sum_{i=1}^{l} \sigma_i w_i v_i^T, \tag{6}$$

where $l = \min(N, mN_t)$ and $n < mN_t$. Here, $W \in \mathbb{R}^{N\times l}$ and $V \in \mathbb{R}^{mN_t\times l}$ are orthogonal matrices, and $\Sigma \in \mathbb{R}^{l\times l}$ is a diagonal matrix containing the singular values. The POD basis Φ is formed by selecting the first n columns of W, minimizing the Frobenius norm $\|U - \Phi\Phi^T U\|_F$.

The reduced-order system of equations is derived from Equation 2 as:

$$(\Phi^T I_N \Phi - \Delta t^{(k)} \Phi^T \mathbf{A}(\mu)\Phi - \Delta t^{(k)} \Phi^T \mathbf{F}(\hat{u}^{(k)}; \mu, t)\Phi)\hat{u}^{(k)} = \hat{u}^{(k-1)} + \Delta t^{(k)} \Phi^T \mathbf{B}(\mu; t) f(\mu), \tag{7}$$

which simplifies to:

$$(I_n - \Delta t^{(k)} A_R(\mu))\hat{u}^{(k)} - \Delta t^{(k)} \Phi^T \mathbf{F}(\Phi\hat{u}^{(k)}, \mu, t)\Phi\hat{u}^{(k)} = \hat{u}^{(k-1)} + \Delta t^{(k)} B(\mu; t) f(\mu), \tag{8}$$

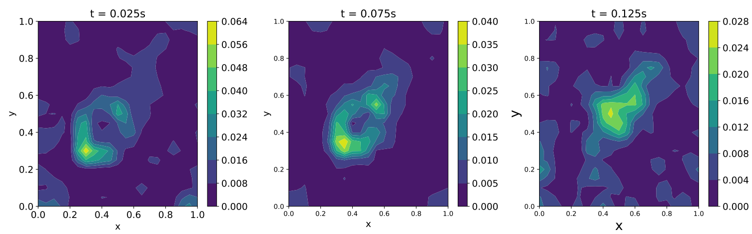
where $A_R = \Phi^T \mathbf{A}(\mu)\Phi$. The reduced-order system can also be derived from the explicit formulation in Equation 3. Beyond linear projection methods like POD, nonlinear manifold-based approaches, such as those employing convolutional autoencoders Romor et al. [2023], can also be used for dimensionality reduction.

3 PINN for Discretized Full-Order and Reduced Order System

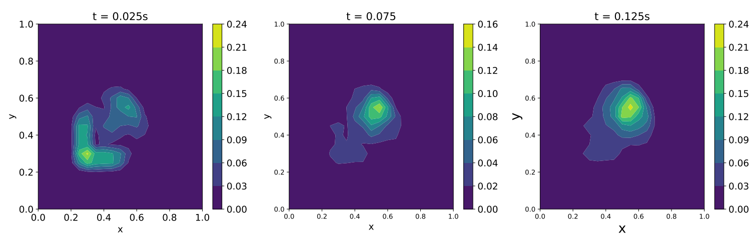
In this section, the discretized full-order or reduced-order governing equation is employed in the physics-informed neural network framework. The input and output structure of a simple ANN network is straightforward. For an unsteady problem, $[x_1, x_2, x_3, \ldots, x_t]$ is the input associated with different time instants and $[y_1, y_2, y_3, \ldots, y_t]$ are the output values corresponding to the inputs. Hence, a one-to-one correspondence exists between the input and output formulations of each time instant. In the Artificial neural network augmented discretized physics-informed neural network (ANN-DisPINN) proposed here, an additional loss penalty is introduced alongside the data-driven loss part, which is obtained from the discretized governing equation in the external solver as explained in Halder et al. [2023].

4 Results

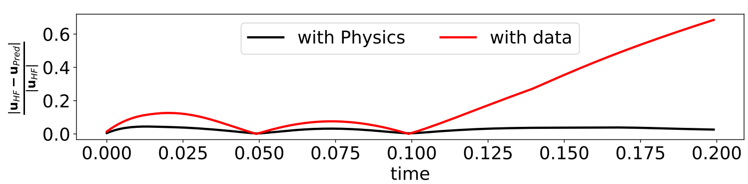
In this section, we consider coupling the unsteady 2-dimensional viscous Burgers' equation with the physics-informed neural network.



(a) Data+Physics



(b) Only Data

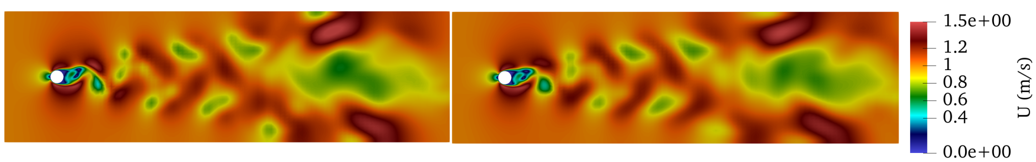


(c) Error Comparison with Time

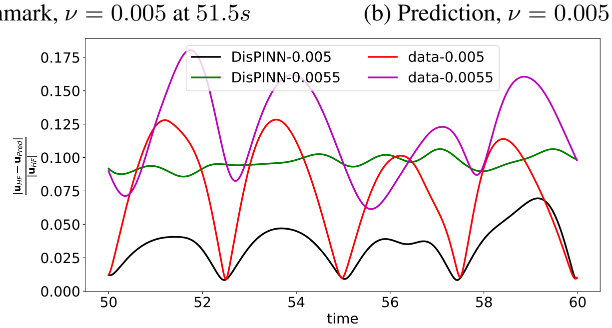
Figure 1: Absoute error with respect to the benchmark data when (a) both data-driven and physics-driven loss term is considered (b) only the data-driven loss term is considered. (c) $L2$-error comparison considering only data and both data and physics-based loss from the external solver.

We have considered 200 total time steps ranging from $0s$ to $0.2s$ to obtain the ground truth solution. The total mesh size is 21×21. The benchmark or the ground truth solution is derived using the finite-volume-based method with a time step size of 0.001. The objective of the current section is to develop a neural network with time instants t as an input to the network and the solution field $\mathbf{u}$ as the output. In the neural network, 4 layers with $124, 100, 80$ and 64 neurons are considered respectively. $softplus$ activation function is considered and the learning rate is 0.006. The total number of the $epoch$ is 4000. Now, in the neural network platform, we consider numerical data from previously computed snapshots of the benchmark solution for the data-driven loss term and the physics-based loss term arising from external solver OpenFOAM at a few intermediate time-instants of the total time considered. The obtained neural network will be utilized to predict the solution fields $\mathbf{u}$ at the future time instants. We have considered 3 intermediate time-steps numerical data (at 0, 50, 100 time steps) for the data-driven loss term in the physics-informed neural network. Now, we want to reconstruct the solution field at time instants $t = 0.025$, $t = 0.075s$ and $t = 0.125s$ and the absolute error of the predicted results with respect to the benchmark are shown in Figure 1. The comparison of the predicted results with the benchmark results is carried out at $t = 0.025$, $t = 0.075$ and $t = 0.125$. The physics-based residual from the external solver is exported from the external solver to the PINN solver at every 10 time step (ranging from $0s$ to $0.2s$) for all the spatial grid locations. Figure 1c, show the comparison of $L2$-error when only data-driven loss term is considered (termed as 'with data') and both the physics-based and data-driven loss terms are considered (with physics). Since, numerical data is considered between 0 to 100 time-steps, the neural network with only numerical data demonstrates good prediction only upto the $0.1s$, whereas the PINN with both data and physics show significantly low error in the full time-span upto $0.2s$.

Now we consider combining reduced-order governing physics with the physics-informed neural network. The full-order model associated with the incompressible inviscid flow is projected using POD-Galerkin projection to obtain a reduced-order algebraic relationship described in Stabile and Rozza [2018]. The reduced order system is solved with a time step of 0.01. We have considered both temporal and parametric variation for the reduced-order PINN framework. The kinematic viscosity considered are 0.005 and 0.006. The input of the neural network comprised of time (t) and kinematic viscosity ν as parameters. The output of the neural network consists of POD coefficients. The residual arising from the reduced-order Ordinary Differential Equation (ODE) as described in Stabile and Rozza [2018] at different time instants is considered as the physics-based residual in the neural network. We have considered the numerical datasets associated with 5 time steps such as at $[0, 250, 500, 750, 1000]$ for the parameters 0.005 and 0.006 at all spatial locations. We compare the predicted results with the benchmark results of the velocity profile after $1.5s$. We have considered 20 POD modes for the velocity and 10 modes for the pressure to obtain the reduced order operators. The predicted results compare pretty well with the benchmark results when we consider both data and physics as shown in Figure 2a and Figure 2b. However when only data is considered, the absolute error between the benchmark and predicted results is significantly high at $\nu = 0.005$ and $\nu = 0.0055$. However, when the reduced order physics is considered as physics-based residual in the ANN-DiPINN framework as demonstrated in subsection 2.1, $L2$ error associated with $\nu = 0.005$ and $\nu = 0.0055$ is lowered.



(a) Benchmark, $\nu = 0.005$ at $51.5s$ (b) Prediction, $\nu = 0.005$ at $51.5s$



(c) L2-error at different ν

Figure 2: Comparison of the (a) benchmark and (b) ANN-DisPINN prediction results of the reduced order incompressible viscous flow past a cylinder at $\nu = 0.005$ after $1.5s$. (c) $L2$-error comparison considering only data and both data and physics from the external solver $\nu = 0.005$ and $\nu = 0.0055$.

References

Deepinder Jot Singh Aulakh, Steven B Beale, and Jon G Pharoah. A generalized framework for unsupervised learning and data recovery in computational fluid dynamics using discretized loss functions. *Physics of Fluids*, 34(7):077111, 2022.

Atilim Gunes Baydin, Barak A Pearlmutter, Alexey Andreyevich Radul, and Jeffrey Mark Siskind. Automatic differentiation in machine learning: a survey. *Journal of Marchine Learning Research*, 18:1–43, 2018.

Umair bin Waheed, Ehsan Haghighat, Tariq Alkhalifah, Chao Song, and Qi Hao. PINNeik: Eikonal solution using physics-informed neural networks. *Computers & Geosciences*, 155:104833, 2021.

Wenqian Chen, Qian Wang, Jan S. Hesthaven, and Chuhua Zhang. Physics-informed machine learning for reduced-order modeling of nonlinear problems. *Journal of Computational Physics*, 446:110666, 2021. ISSN 0021-9991. doi: https://doi.org/10.1016/j.jcp.2021.110666. URL `https://www.sciencedirect.com/science/article/pii/S0021999121005611`.

Chen Cheng and Guang-Tao Zhang. Deep learning method based on physics informed neural network with resnet block for solving fluid flow problems. *Water*, 13(4):423, 2021.

Salvatore Cuomo, Vincenzo Schiano Di Cola, Fabio Giampaolo, Gianluigi Rozza, Maziar Raissi, and Francesco Piccialli. Scientific machine learning through physics–informed neural networks: where we are and what's next. *Journal of Scientific Computing*, 92(3):88, 2022.

MWMG Dissanayake and Nhan Phan-Thien. Neural-network-based approximations for solving partial differential equations. *communications in Numerical Methods in Engineering*, 10(3):195–201, 1994.

Zhiwei Fang and Justin Zhan. A physics-informed neural network framework for PDEs on 3D surfaces: Time independent problems. *IEEE Access*, 8:26328–26335, 2019.

Han Gao, Luning Sun, and Jian-Xun Wang. PhyGeoNet: Physics-informed geometry-adaptive convolutional neural networks for solving parameterized steady-state PDEs on irregular domain. *Journal of Computational Physics*, 428:110079, 2021.

Rahul Halder, Giovanni Stabile, and Gianluigi Rozza. Physics informed neural network framework for unsteady discretized reduced order system. *arXiv preprint arXiv:2311.14045*, 2023.

Saddam Hijazi, Melina Freitag, and Niels Landwehr. POD-Galerkin reduced order models and physics-informed neural networks for solving inverse problems for the Navier–Stokes equations. *Advanced Modeling and Simulation in Engineering Sciences*, 10(1):1–38, 2023.

M. Raissi, P. Perdikaris, and G.E. Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378(1):686–707, 2019. doi: https://doi.org/10.1016/j.jcp.2018.10.045. URL `https://www.sciencedirect.com/science/article/pii/S0021999118307125`.

Rishikesh Ranade, Chris Hill, and Jay Pathak. Discretizationnet: A machine-learning based solver for navier–stokes equations using finite volume discretization. *Computer Methods in Applied Mechanics and Engineering*, 378:113722, 2021.

Francesco Romor, Giovanni Stabile, and Gianluigi Rozza. Non-linear manifold reduced-order models with convolutional autoencoders and reduced over-collocation method. *Journal of Scientific Computing*, 94(3):74, 2023.

Enrico Schiassi, Roberto Furfaro, Carl Leake, Mario De Florio, Hunter Johnston, and Daniele Mortari. Extreme theory of functional connections: A fast physics-informed neural network method for solving ordinary and partial differential equations. *Neurocomputing*, 457:334–356, 2021.

Giovanni Stabile and Gianluigi Rozza. Finite volume POD-Galerkin stabilised reduced order methods for the parametrised incompressible Navier–Stokes equations. *Computers & Fluids*, 173:273–284, 2018. ISSN 0045-7930. doi: https://doi.org/10.1016/j.compfluid.2018.01.035. URL `https://www.sciencedirect.com/science/article/pii/S0045793018300422`.

Alexandre M Tartakovsky, C Ortiz Marrero, Paris Perdikaris, Guzel D Tartakovsky, and David Barajas-Solano. Physics-informed deep neural networks for learning parameters and constitutive relationships in subsurface flow problems. *Water Resources Research*, 56(5):e2019WR026731, 2020.