Physics-Informed Deep Operator Networks for Surrogate Modeling of Incompressible Flows

Himanshu Mittal
Department of Aerospace Engineering
Indian Institute of Technology Kanpur
Kanpur, India 208016
himanshum@iitkalumni.org

Prashant Kumar
Department of Aerospace Engineering
Indian Institute of Technology Kanpur
Kanpur, India 208016
kumarp22@iitk.ac.in

Rajesh Ranjan
Department of Aerospace Engineering
Indian Institute of Technology Kanpur
Kanpur, India 208016
rajeshr@iitk.ac.in

Abstract

We present a surrogate modeling framework for incompressible Navier–Stokes flows using Physics-Informed Deep Operator Networks (PI-DeepONets). The approach builds upon DeepONet's ability to learn nonlinear solution operators while embedding governing equations into the training objective. Starting with a data-driven formulation, we progressively incorporate physics residuals and adaptive boundary condition weighting, resulting in improved physical fidelity and generalization. Two canonical benchmark problems—lid-driven cavity (LDC) flow over a wide Reynolds number range and laminar flow past a circular cylinder—are used to evaluate the models. The PI-DeepONet is trained on a limited set of CFD snapshots and generalizes to unseen Reynolds numbers in the same flow regime with significantly reduced test error and improved adherence to continuity and momentum balance. Results highlight the trade-off between numerical fidelity and physical constraint enforcement, demonstrating that physics-informed loss and adaptive weighting can enhance model robustness, particularly in high-Reynolds-number in the same regime.

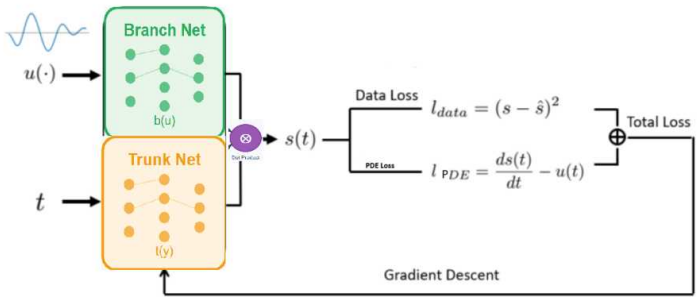
1 Introduction

Surrogate modeling offers a fast and accurate way to approximate complex CFD flow fields without repeatedly solving costly PDEs. While high-fidelity solvers are accurate, they are computationally expensive for tasks like optimization or real-time control. Physics-informed neural networks (PINNs) mitigate this by embedding governing equations into the loss, reducing data needs, but their focus on specific solutions limits generalization across varying conditions.

Deep Operator Networks (DeepONets; Lu et al.)represent a significant shift from solution regression to operator learning, allowing the network to learn mappings between function spaces. Instead of solving for a specific state, a DeepONet can infer the solution operator that maps input functions (e.g., boundary conditions, source terms, or parameters like Reynolds number) to entire solution fields. This property makes DeepONets inherently more flexible and suitable for parametric PDE problems.

Despite these advances, purely data-driven DeepONets may not strictly enforce governing equations, leading to accumulated physical inconsistencies, especially in extrapolative regimes. To address these challenges, we propose a Physics-Informed Deep Operator Network (PI-DeepONet; Wang et al. [2021]) framework for incompressible Navier–Stokes flows (see Figure 1). Our formulation combines

2nd Conference on Applied AI and Scientific Machine Learning (CASML 2025).



(a)

Figure 1: Physics Informed Deep Operator Network

the expressive operator-learning capabilities of DeepONets with the physics of PDE-constrained learning. By incorporating momentum and continuity residuals directly into the loss function, the network enforces physical laws even in data-sparse regions. This approach is particularly effective in problems with strong boundary-layer effects or complex wake structures, such as flow past a cylinder.

This study talks about PI-DeepONets as a scalable, physics-consistent surrogate modeling strategy for fluid dynamics, with potential applications in design optimization and reduced-order modeling.

2 Methodology

2.1 DeepONet Architecture

The base architecture comprises two subnetworks: a branch net that encodes parametric input (Reynolds number or boundary profile), and a trunk net that encodes spatial coordinates (x, y). Their inner product yields velocity and pressure predictions (u, v, p) (see Figure 1). We use 3 hidden layers with 64–128 neurons per layer and activation functions (SiLU or Wavelet Functions Zhao et al. [2023]), selected based on convergence characteristics.Lu et al.

2.2 Hybrid Loss Formulation

The total training loss is defined as:

$$\mathcal{L} = \mathcal{L}_{\text{data}} + \lambda_p \mathcal{L}_{\text{PDE}} + \lambda_b \mathcal{L}_{\text{BC}}, \tag{1}$$

where $\mathcal{L}_{\text{data}}$ is the mean squared error to CFD data, $\mathcal{L}_{\text{PDE}}$ enforces the Navier–Stokes momentum and continuity equations, and $\mathcal{L}_{\text{BC}}$ enforces Dirichlet boundary conditions.

The two-dimensional incompressible Navier–Stokes equations in dimensional form are given as:

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} = -\frac{1}{\rho}\frac{\partial p}{\partial x} + \nu\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right), \tag{2}$$

$$\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} = -\frac{1}{\rho}\frac{\partial p}{\partial y} + \nu\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}\right), \tag{3}$$

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0. \tag{4}$$

To obtain the non-dimensional form, we introduce characteristic scales:

- Length scale: L
- Velocity scale: U
- Time scale: L/U

- Pressure scale: ρU^2

We define the non-dimensional variables as:

$$x^* = \frac{x}{L}, \quad y^* = \frac{y}{L}, \quad u^* = \frac{u}{U}, \quad v^* = \frac{v}{U}, \quad t^* = \frac{t}{L/U}, \quad p^* = \frac{p}{\rho U^2}.$$

Substituting these into the dimensional equations the non-dimensional Navier–Stokes equations become:

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} = -\frac{\partial p}{\partial x} + \frac{1}{Re}\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right), \tag{5}$$

$$\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} = -\frac{\partial p}{\partial y} + \frac{1}{Re}\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}\right), \tag{6}$$

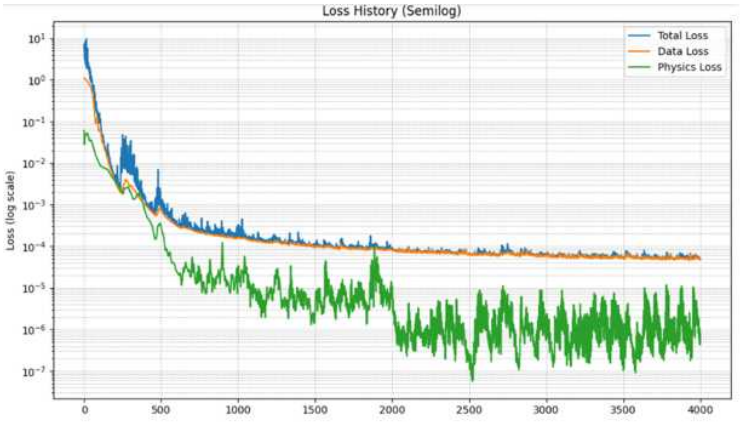
$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0, \tag{7}$$

where $Re = \dfrac{UL}{\nu}$ is the Reynolds number.

These equations govern the fluid motion and its residuals serve as the physical loss of PDE for developing physics-informed DeepONet models. An adaptive weighting strategy dynamically adjusts λ_p and λ_b to prevent loss domination and improve stability.

2.3 Optimization Strategy

We employ a multi-stage optimization strategy to ensure stable and efficient convergence of the PI-DeepONet models. The network is first trained using the Adam optimizer (initial learning rate 10^{-3}) for rapid descent and good initialization, followed by L-BFGS quasi-Newton refinement to achieve low residuals efficiently. To further enhance convergence, we incorporate the SOAP (Second Order Adjoint Propagation) optimizer, which leverages second-order curvature information through adjoint computations, offering improved stability for stiff PDE losses and achieving lower final training residuals than standard first-order methods.Vyas et al. [2024]



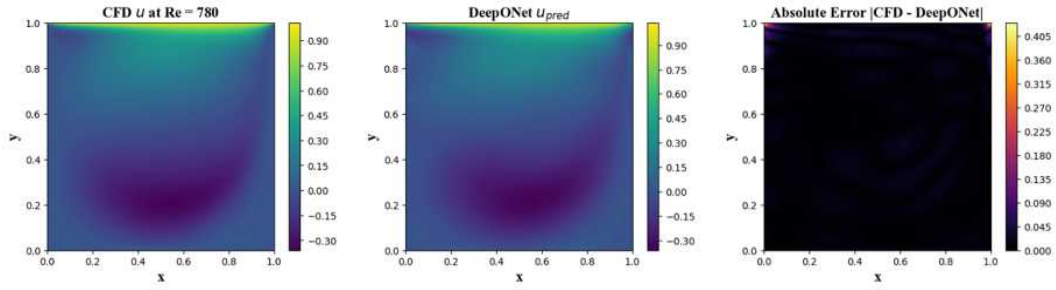
(a)

Figure 2: Training loss with SOAP optimizer with Adaptive weighting

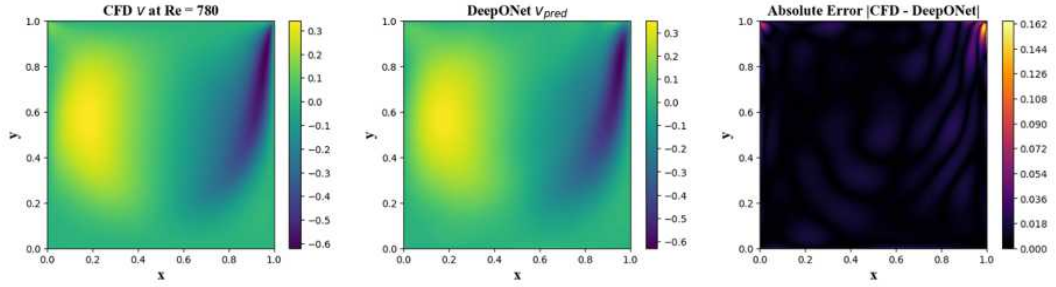
3 Application Cases

3.1 Lid-Driven Cavity Flow

The first benchmark involves a 2D unit square domain with lid velocity $u = 1$ at $y = 1$ and no-slip boundaries elsewhere. Training data are generated for some random Reynolds numbers in the



(a)



(b)

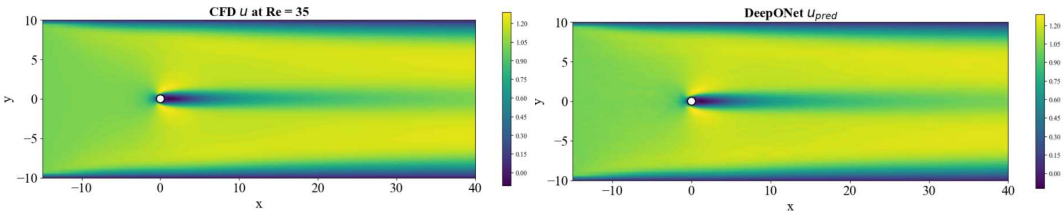
Figure 3: Velocity magnitude contours for the Lid-Driven Cavity flow at $Re = 780$. PI-DeepONet accurately captures the primary vortex and corner eddies compared to the CFD reference solution.

range of 400–1000 using a structured mesh with 161×161 points, and tested on unseen Reynolds numbers up to 1000.Luo et al. [2023] PI-DeepONet shows lower test errors and better vortex structure reconstruction compared to data-only DeepONet, particularly at higher Re. **??** shows the comparison between CFD and model predictions at $Re = 780$.

3.2 Laminar Flow Past a Cylinder

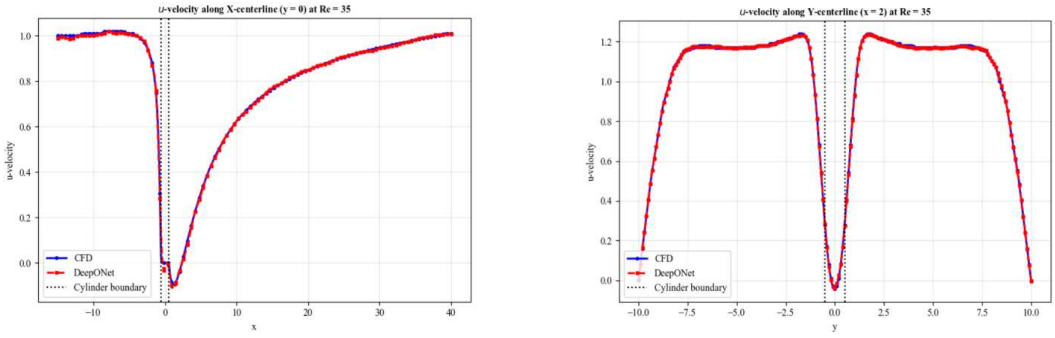
The second case examines steady laminar flow around a cylinder for $Re = 30, 35, 40$ [Luo et al., 2023]. This open-domain setup places greater emphasis on wall boundary enforcement. To capture complex wake dynamics and steep near-wall velocity gradients, a ResNet-enhanced DeepONet was used, with residual blocks embedded in both branch (Reynolds input) and trunk (spatial encoding) networks. Residual connections improved gradient flow and stability, allowing the model to resolve sharp boundary-layer features effectively.

Training combined Adam for initial descent with L-BFGS and SOAP for refinement, where SOAP notably improved convergence and reduced PDE residuals, especially at $Re = 35$. This led to lower L_2 velocity errors and improved wall condition enforcement (error reduced from 8.6×10^{-4} to 2.7×10^{-4}). Adaptive boundary weighting further enhanced accuracy by prioritizing wall regions, enabling precise wake reconstruction and reducing wake length error to 1.7% at $Re = 35$. Figure 4 shows the comparison between CFD and model predictions at $Re = 35$.



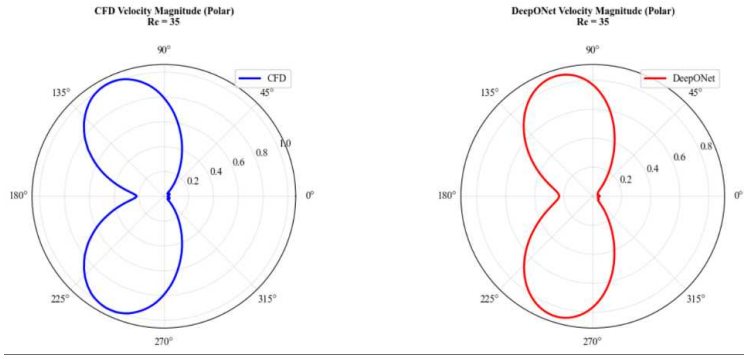
(a) (b)

Figure 4: Streamlines and velocity contours for flow past a cylinder at $Re = 35$. The DeepONet and ResNet model with adaptive BC weighting captures the near-wake structure with high accuracy.



(a) (b)

Figure 5: Graphs of the Velocity magnitude predicted by the model and the CFD Simulation along the X and Y Axis.



(a)

Figure 6: Velocity Magnitude predicted by the model radially around the Cylinder

4 Conclusions

The three model variants—(i) Data-only DeepONet, (ii) PI-DeepONet, and (iii) ResNet enhanced PI-DeepONet with adaptive BC weighting—were evaluated across 61 epochs using Adam, L-BFGS, and SOAP optimizers. The data-driven model reached a training loss plateau of $\mathcal{L}_{\text{train}} \approx 6.5 \times 10^{-6}$, whereas the PI-DeepONet attained $\mathcal{L}_{\text{train}} \approx 1.2 \times 10^{-4}$, reflecting the additional PDE residual constraints. Incorporating adaptive BC weighting further reduced boundary residual errors by approximately 78%.

In terms of optimizer performance, SOAP exhibited the fastest convergence, reaching a combined loss of $\sim 10^{-5}$ in 40 epochs, compared to 55 epochs for L-BFGS and 70 for Adam. Moreover, SOAP yielded lower PDE residual loss at convergence ($\mathcal{L}_{\text{PDE}} \approx 8.7 \times 10^{-6}$) relative to L-BFGS ($\mathcal{L}_{\text{PDE}} \approx 1.6 \times 10^{-5}$). These improvements are particularly significant for stiffer high-Re cases.

We have demonstrated a PI-DeepONet framework capable of learning incompressible flow operators from limited CFD data. Integration of PDE residuals and adaptive boundary weighting along with ResNet augmentation significantly enhance physical realism and generalization.

References

Lu Lu, Pengzhan Jin, and George Em Karniadakis. Deeponet: Learning nonlinear operators for identifying differential equations based on the universal approximation theorem of operators. *Nature Machine Intelligence*, 3(3).

Yining Luo, Yingfa Chen, and Zhen Zhang. Cfdbench: A large-scale benchmark for machine learning methods in fluid dynamics. *arXiv preprint arXiv:2301.XXXXX*, 2023.

Nikhil Vyas, Depen Morwani, Rosie Zhao, Mujin Kwun, Itai Shapira, David Brandfonbrener, Lucas Janson, and Sham Kakade. Soap: Improving and stabilizing shampoo using adam. *arXiv preprint arXiv:2401.XXXXX*, 2024.

Sifan Wang, Hanwen Wang, Paris Perdikaris, and George E. Karniadakis. Learning the solution operator of parametric partial differential equations with physics-informed deeponets. *Science Advances*, 7, 2021.

Zhiyuan Zhao, Xueying Ding, and B. Aditya Prakash. Pinnsformer: A transformer-based framework for physics-informed neural networks. In *Proceedings of the Conference on Neural Information Processing Systems (NeurIPS)*, 2023.