Proceedings of the Canadian Society for Mechanical Engineering International Congress
32nd Annual Conference of the Computational Fluid Dynamics Society of Canada
Canadian Society of Rheology Symposium
CSME-CFDSC-CSR 2025
May 25–28, 2025, Montréal, Québec, Canada

Experimental Validation of Pressure Drop in Turbulent Annular Flow Pressure Drop Using CFD and Physical Informed Neural Networks

Alaaeddin Elhemmali[1*], Mohammad Mojammel Huque[1] , Syed Imtiaz[1], Mohammad Azizur Rahman[2] , Salim Ahmed[1]
[1]Process Engineering, Memorial University of Newfoundland, St John's, Canada
[2]College of Science and Engineering (CSE), Hamad Bin Khalifa University (HBKU), Doha, Qatar
*Corresponding: ae2043@mun.ca

***Abstract*—** This work presents a comprehensive study in machine learning for fluid dynamics, with a focus on turbulent flow analysis and the application of Physics Informed Neural Networks (PINNs). Motivated by the need to validate PINNs against experimental data. This research investigates a lab scale setup that provides insights which can be extended to practical applications in the future. The experimental setup mimics annular flow in oil drilling were used for collecting flow and pressure data for validation then compared to the results of computational simulations that conducted using the Reynolds-Averaged Navier-Stokes (RANS) equations with the (k−ω SST) turbulence model in ANSYS Fluent. Finally, PINNs were employed to solve governing equations directly, integrating physical laws and data without requiring turbulence models. The findings underscore the potential of PINNs as a resilient tool for fluid flow prediction, offering advantages in computational efficiency and simplicity over conventional CFD models.

Keywords; Physics Informed Neural Networks (PINNs), Annuls Turbulent Flow, Reynolds-Averaged Navier-Stokes (RANS) Equations, Computational Fluid Dynamics (CFD), Pressure Drop.

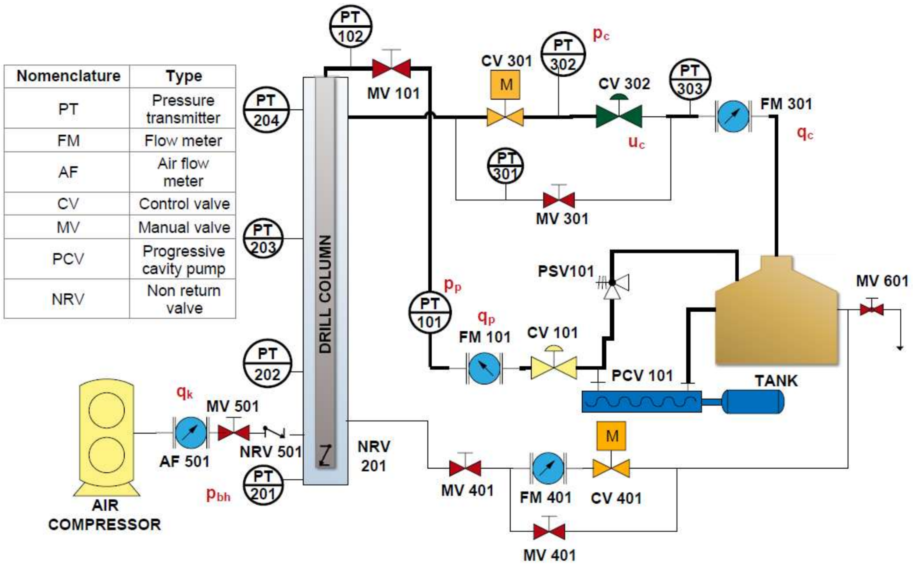
I. Introduction

Recent research in fluid dynamics machine learning has been focused on exploring data-driven approaches for turbulent flow analysis such as approximating the effects of turbulent fluctuations on fluid flow, forecasting how turbulent flow parameters vary over time, deriving turbulence theory directly from experimental data, measurement of flow properties in turbulent flows without physically disturbing the flow field, manipulating flow conditions to achieve desired outcomes. Starting with deep neural networks, Julia Ling and Jeremy Templeton (2016) proposed a novel neural network architecture embedding Galilean invariance, ensuring that the model behaves consistently with the physical laws regardless of the coordinate system [1]. Their study showcased significant improvements over conventional RANS models, highlighting the potential of deep neural networks in capturing complex flow phenomena. Building on this, Chao Jiang et al. (2021) introduced a universally interpretable machine learning framework for turbulence modeling, their approach demonstrated the development of an invariant, realizable, unbiased, and robust data- driven turbulence model, underscoring the importance of incorporating domain knowledge into machine learning algorithms for accurate predictions [2]. Next, in the realm of temporal dynamics prediction, Prem Srinivasan et al. (2019) investigates the capabilities of neural networks in prediction of temporal dynamics in turbulent flows, specifically multilayer perceptron (MLP) and long short-term memory (LSTM) networks [3], in predicting temporally evolving turbulent flows. Their study revealed the effectiveness of LSTM networks in capturing turbulence statistics and dynamical behavior, suggesting the potential of machine learning frameworks in developing data-driven sub grid-scale models for large-eddy simulations. Similarly, Hamidreza Eivazi et al. (2020) presented an innovative data-driven approach employing deep learning networks for Reduced Order Modeling (ROM) and long short-term memory networks (LSTM) to forecast future velocity fields. Their method shows significantly better performance in predicting fluid flow evolution when compared to conventional techniques such as Dynamic Mode Decomposition (DMD) and Proper Orthogonal Decomposition (POD) [4]. Expanding on the use of recurrent architecture, Hamidreza Eivazi et al. (2021) also studied the capabilities of neural networks in prediction of temporal dynamics in turbulent flows, but he employed recurrent neural networks (RNNs) and Koopman-based frameworks for temporal predictions in a low-order model of turbulence. Their findings showcased the ability of RNNs and Koopman-based frameworks to accurately reproduce long-term statistics and dynamic behavior with minimal computational expense [5]. Convolutional neural networks (CNNs) have also played a significant role in fluid dynamics studies. Takaaki Murata et al. (2020) used convolutional neural networks (CNNs) for nonlinear model decomposition in fluid dynamics, demonstrating improved capability in capturing flow features compared to traditional methods such as proper orthogonal decomposition (POD). Their study emphasized the potential of CNNs in extracting interpretable flow features in lower dimensions, thus contributing to the development of efficient reduced-order models for turbulent flows [6]. Luca Guastoni et al. (2021) further explored the application of convolutional neural network (CNN) models for predicting wall-bounded turbulence from wall quantities. By training CNNs using data from direct numerical simulations, they demonstrated improved predictions of velocity fluctuations at different wall-normal locations, outperforming traditional methods like extended proper orthogonal decomposition (EPOD). Moreover, they explored the feasibility of transfer learning for enhancing prediction accuracy across different Reynolds numbers, showcasing the potential for non-intrusive sensing models in closed-loop control applications [7]. Generative adversarial networks (GANs) have also been applied in turbulent flow analysis. Alejandro Güemes et al. (2021) evaluated the applicability of super-resolution generative adversarial networks (SRGANs) for reconstructing turbulent-flow quantities from coarse wall measurements. Their study demonstrated that SRGANs can effectively enhance the resolution of wall measurements and accurately reconstruct wall-parallel velocity fields, even for challenging cases with high down-sampling factors. By using SRGANs, the researchers provided a promising methodology for non-intrusive sensing of turbulent flows, essential for closed-loop control applications [8]. Additionally, machine learning methods have made their mark in active flow control research. Hongwei Tang et al. (2020) focused on active flow control using deep reinforcement learning (DRL) techniques, particularly proximal policy optimization (PPO), to control mass flow rates in synthetic jets for a range of Reynolds numbers. Their study showcased significant reductions in lift and drag fluctuations, highlighting the potential of DRL in achieving robust flow control across different flow configurations and Reynolds numbers [9]. Building on this, Feng Ren et al. (2021) extended the application of DRL to active flow control in weakly turbulent conditions, specifically targeting drag reduction for a circular cylinder at intermediate Reynolds numbers. Despite the challenges posed by turbulent flow features, their study demonstrated effective control strategies using PPO DRL, achieving remarkable drag reductions, and mitigating turbulent fluctuations in the cylinder wake [10]. The utilized machine learning methods in the previous mentioned studies share the fundamental objective of learning patterns and making predictions based on relying on extensive and varied datasets to train and apply their knowledge effectively. The quality and accessibility of data significantly influence the performance and precision of these algorithms. Acquiring data can be costly, time consuming, or unfeasible, especially in fields that demand specialized knowledge. Moreover, biases or constraints inherent in the training data may affect the decisions made by deep learning models, possibly resulting in deviation in the predictive model [11]. Beyond traditional data-driven approaches, physical neural networks (PINNs) have emerged as a powerful tool in fluid dynamics as it incorporates data with physical laws into the architecture of Networks in the form of partial differential equations [12] tackling complex phenomena with limited data. Raissi et al. (2019) studied deep neural networks ability to predict lift and drag forces on vibrating structures from sparse velocity data, marking a shift in traditional simulation approaches [13]. Hidden fluid mechanics (HFM), introduced by Raissi et al. (2020), extracts velocity and pressure fields with unprecedented accuracy from flow visualizations [14]. Mojtaba et al. (2020) demonstrate data-driven super-resolution and denoising of 4D-Flow MRI for improved vascular disease diagnostics [15]. Amirhossein Arzani et al. (2021) utilize physics-informed neural networks to quantify near-wall blood flow dynamics. These studies highlight deep learning transformative role in enhancing understanding and prediction of fluid dynamics [16]. Moreover, PINNs are also successfully applied for solving Reynolds-averaged Navier–Stokes (RANS) equations, as demonstrated by the work of Hamidreza Eivazi et al. (2022). By integrating data on domain boundaries with the governing equations, PINNs offer an alternative approach for simulating turbulent flows without specific turbulence models. Their results showcase the applicability of PINNs for accurate prediction of flow characteristics, even in scenarios with strong pressure gradients and turbulent flows over complex geometries [17]. Despite the advancements and potential of PINNs in capturing complex flow phenomena and improving predictive accuracy, there still exists a notable gap in validating this method against experimental data, which is essential to ensure its reliability in real-world applications under general conditions. Therefore, the primary goal of this work is to address this validation gap by validating physics informed neural networks (PINNs) not only against computational data but also verses experimental data. The aim is to demonstrate the capability of PINNs in accurately predicting flow pressure drop along annular flow experimental measurements that are not used in the PINNs architecture. This validation process aims to establish the reliability of PINNs as a practical tool for fluid dynamics analysis and prediction.

II. Methodology

A. Experimental Setup

The experimental setup shown in Figure1. was designed and constructed at Memorial University of Newfoundland [18], was specifically made to replicate and control various scenarios encountered during oil drilling within a laboratory setting [19]. It consists of a drill column, measuring 1.5 inches in diameter, positioned concentrically and vertically within a 3 inches pipe acting as the casing. Water is circulated throughout the system via a water tank connected to a pump, ensuring a closed-loop configuration. A standard drilling scenario was simulated, where water from the tank is pumped into the 16.5 feet drill column and returned to the supply tank via the top of the casing, with both valves (MV 401) and (MV 301) kept closed, to maintain a single-fluid system without the introduction of kick. The air compressor, though only used for gas kick simulations, was kept shut during this experiment. Pressure readings were obtained at various points along the casing using transmitters capable of providing a current signal within the range of 4-20 mA with 0.1% accuracy [20]. Additionally, real-time flow measurements were recorded by a flow meter generating a current signal within the same range, all of which were subsequently converted into standard units for analysis.



Nomenclature	Type
PT	Pressure transmitter
FM	Flow meter
AF	Air flow meter
CV	Control valve
MV	Manual valve
PCV	Progressive cavity pump
NRV	Non return valve

Figure 1. Experimental Setup, [18]

B. Computational Model

The computational model includes components, geometry, mesh generation, and formulation, focusing on the annular between the drill pipe and casing, meshed as shown in Figure 2.

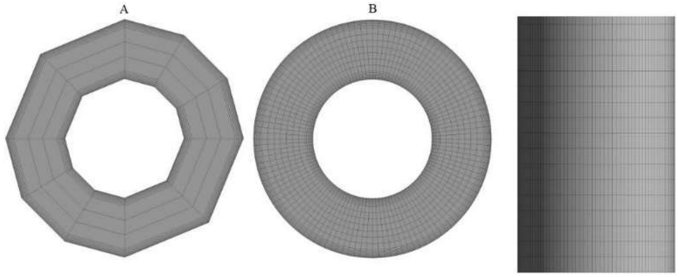


Figure 2. Meshing Geometry

The mesh independence test [21], shown in Figure 3, incrementally increased node numbers from shape A to B, monitoring the residual mass flow rate [22].

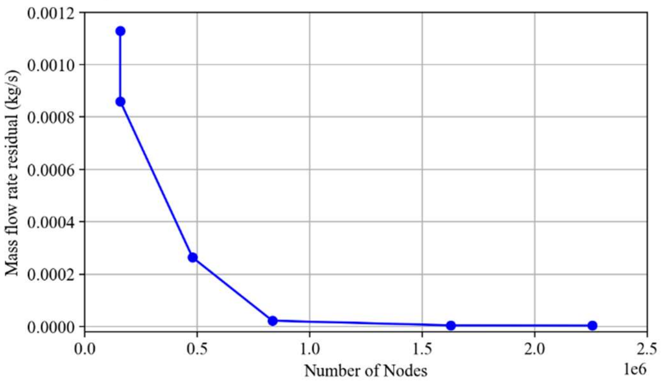


Figure 3. Mesh Independency Test

A finer mesh with 1,625,670 nodes, 1,563,120 elements, and ten inflation layers (growth rate 1.2) ensured stability and mesh independence. For a comprehensive grasp of the underlying principles behind the ANSYS computations, the model initiates with the Reynolds-Averaged Navier-Stokes (RANS) equation [23].

$$\rho \frac{\partial(\bar{u}_\iota \bar{u}_j)}{\partial x_j} = -\frac{\partial P}{\partial x_i} + \frac{\partial}{\partial x_j}(\mu \frac{\partial \bar{u}_\iota}{\partial x_j} - \rho \overline{u_\iota' u_j'}) \qquad (1)$$

In this context, (ρ) stands for density, $(\bar{\mathrm{u}})$ denotes the mean velocity, (P) represents pressure, (μ) is dynamic viscosity, and (u') signifies velocity fluctuations due to turbulence. Contrasting with the laminar version of the Navier-Stokes equation, an additional term, known as the Reynolds stress (τ_{T}) [24], appears on the left-hand side of equation (1) where:

$$\tau_T = -\rho \overline{u_\iota' u_j'} \qquad (2)$$

To tackle the Reynolds-Averaged Navier-Stokes (RANS) equations, the Reynolds stress is conventionally estimated by relating it to the mean velocity gradient, following Joseph Boussinesq's hypothesis from 1877 [25] [26]:

$$-\rho \overline{u_\iota' u_j'} = \mu_t (\frac{\partial \bar{u}_\iota}{\partial x_j} + \frac{\partial \bar{u}_J}{\partial x_i}) - \frac{2}{3}\delta_{ij}(\rho k + \mu_t \frac{\partial \overline{u_k}}{\partial x_k}) \qquad (3)$$

Where (μ_{t}) denotes an artificial attribute termed as eddy viscosity or turbulent viscosity. Unlike dynamic viscosity, it is not a fluid property, but instead depends on flow dynamics, and (δ_{ij}) represents the Kronecker delta, which combines eddy shear stress and normal eddy stresses into one equation, The variable (k) signifies turbulent kinetic energy associated with fluctuating velocity, as depicted in the equation below [23].

$$k = \frac{1}{2}(\overline{u_\iota' u_\iota'} + \overline{u_j' u_j'} + \overline{u_k' u_k'}) \qquad (4)$$

Identify applicable sponsor/s here. *(sponsors)*

To complete the equations system, modeling the eddy viscosity and the turbulent kinetic energy is necessary, which are then employed in accordance with Boussinesq assumption to incorporate turbulent stress into the momentum equation. The approach employed in this study for this purpose is the (k- ω) SST model, falling within a class of two-model frameworks that involve solving for both the turbulent kinetic energy (k) as in equation (5) and the specific turbulent dissipation rate (ω) as in equation (6). Widely employed in computational software packages, this model closely resembles the baseline BSL (k- ω) model, incorporating a blended function to determine diffusivity effectiveness, thus facilitating a smooth transition between the standard (k- ω) approach near walls and the standard (k-ε) approach further from the boundary region. Additionally, it adjusts the turbulent viscosity to accommodate the transport of the primary turbulent shear stress (SST).

$$\frac{\partial}{\partial t}(\rho k) + \frac{\partial}{\partial x_i}(\rho k\bar{u}_\iota) = \frac{\partial}{\partial x_j}(\Gamma_k \frac{\partial k}{\partial x_j}) + G_k - Y_k + G_b \tag{5}$$

$$\frac{\partial}{\partial t}(\rho \omega) + \frac{\partial}{\partial x_i}(\rho \omega\bar{u}_\iota) = \frac{\partial}{\partial x_j}(\Gamma_\omega \frac{\partial \omega}{\partial x_j}) + G_\omega - Y_\omega + D_\omega + G_{\omega b} \tag{6}$$

These coefficients are designated as follows; (Γ_k) and (Γ_ω) denote the effective diffusivity of (k) and (ω), respectively. (G_k) represents the production of (k), while (G_ω) signifies the generation of (ω). (Y_k) and (Y_ω) denote the dissipation of (k) and (ω), respectively. Additionally, (D_ω) is referred to as a cross diffusion term, and (G_b , $G_{\omega b}$) account for the bouncy term. ANSYS Theory Guide, reference [26], offers detailed insights into how these coefficients are modeled in the ANSYS software.

C. *Phsics Informed Neural Networks*

The primary idea behind employing the Physics Informed Neural Network as shown in Figure 4. is to address the fundamental governing equation directly, eliminating the necessity for modeling or introducing artificial variables [27].

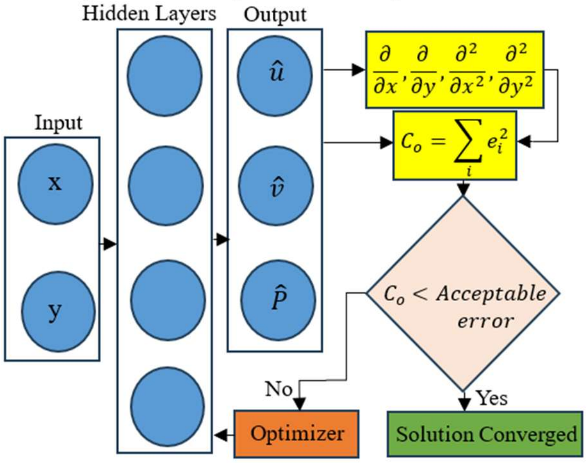


Figure 4. Physics Neural Networks Chart

This means, it solves equation (1) in conjunction with the fundamental mass equation (7) under the given boundary conditions.

$$\frac{\partial \bar{u}_\iota}{\partial x_i} + \frac{\partial \bar{u}_J}{\partial x_j} = 0 \tag{7}$$

The solution process commences at the input layer by providing the network with randomly selected coordinates situated both inside the domain and at its boundaries. These coordinates are subsequently passed to the hidden layer, where they undergo a sequence of linear transformations employing weights and biases. Following this, non-linear transformations utilizing an activation function are applied to capture intricate relationships within each neuron. In this study, the hyperbolic tangent function is employed for its continuous differentiability. The output layer then presents the final solution, which is subsequently assessed using a cost function (C_o). This function is established by accumulating all residuals, computed as follows: by denoting outcomes obtained from implementing Physics-Informed Neural Networks with a hat superscript, the residual of each governing equation resulting from the application of the physical neural networks can be computed.

$$\epsilon_1 = (\rho \frac{\partial \bar{\hat{u}}_i \bar{\hat{u}}_j}{\partial x_j} + \frac{\partial \hat{P}}{\partial x_i} - \frac{\partial}{\partial x_j}(\mu \frac{\partial \bar{\hat{u}}_\iota}{\partial x_j} - \rho \overline{\hat{u}'_\iota \hat{u}'_j}))^2 \tag{8}$$

$$\epsilon_2 = (\frac{\partial \bar{\hat{u}}_\iota}{\partial x_i} + \frac{\partial \bar{\hat{u}}_J}{\partial x_j})^2 \tag{9}$$

Similarly, the residual results arising from the boundary conditions can be expressed as follows:

$$\epsilon_3 = (P_{in} - \hat{P}_{in})^2 + (P_{out} - \hat{P}_{out})^2 \tag{10}$$

Where (Pin) and (Pout) represent the measured inlet and outlet pressure, while ($\hat{P}_{in}$) and ($\hat{P}_{out}$) represent the neural networks inlet and outlet pressure respectively. Accumulating all residuals will form the cost function for training (C_o) [28].

$$Co = \sum_i \epsilon_i \tag{11}$$

Iteratively adjusting the weights and biases through backpropagation reduces the cost function, enhancing the neural networks prediction and accuracy while fulfilling both governing equations and boundary conditions.

III. RESULTS AND DISCUSSION

In this section, we will compare the data obtained from physical neural networks with computational and experimental data. As shown in Figure 5, a discrepancy is observed in the transient region, which is expected due to its transient nature, even in experimental measurements. However, for higher Reynolds numbers, the data alignment with the experiment improves significantly as shown in Figure 6. The regression coefficient (R^2) for the CFD ranges between 0.78 and 0.98, while the regression coefficient for the PINNs ranges between 0.97 and 0.99, indicating a better fit for the PINNs model.

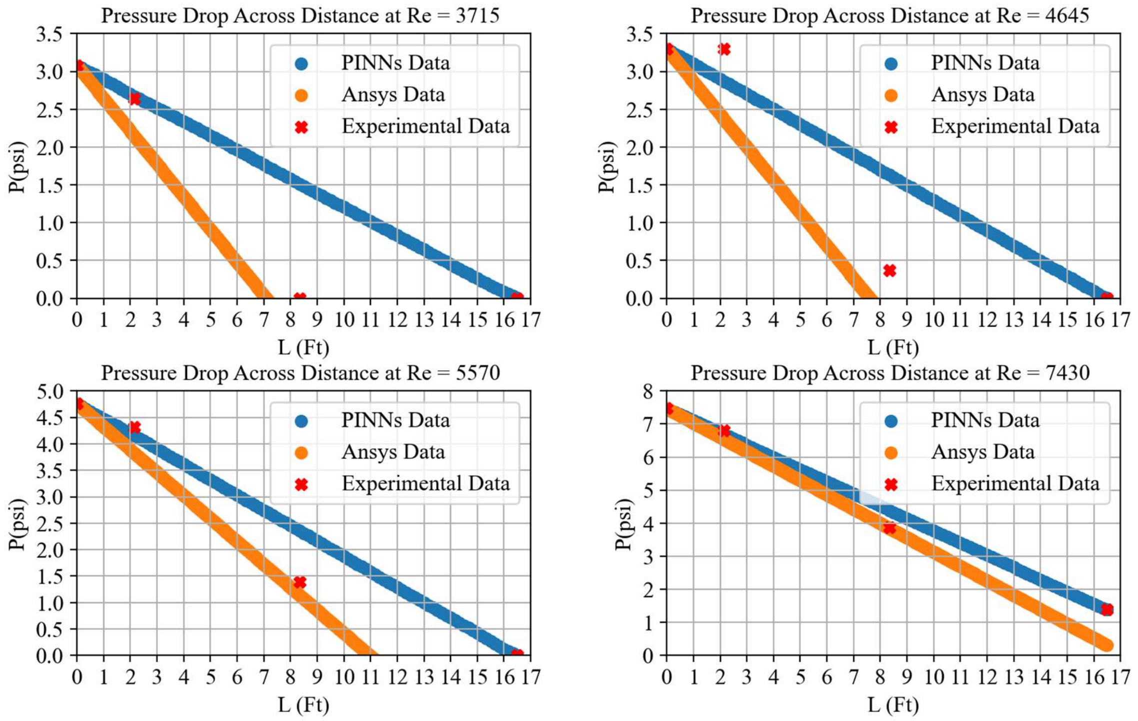


Figure 5. Pressure Drop at Transient Flow Region

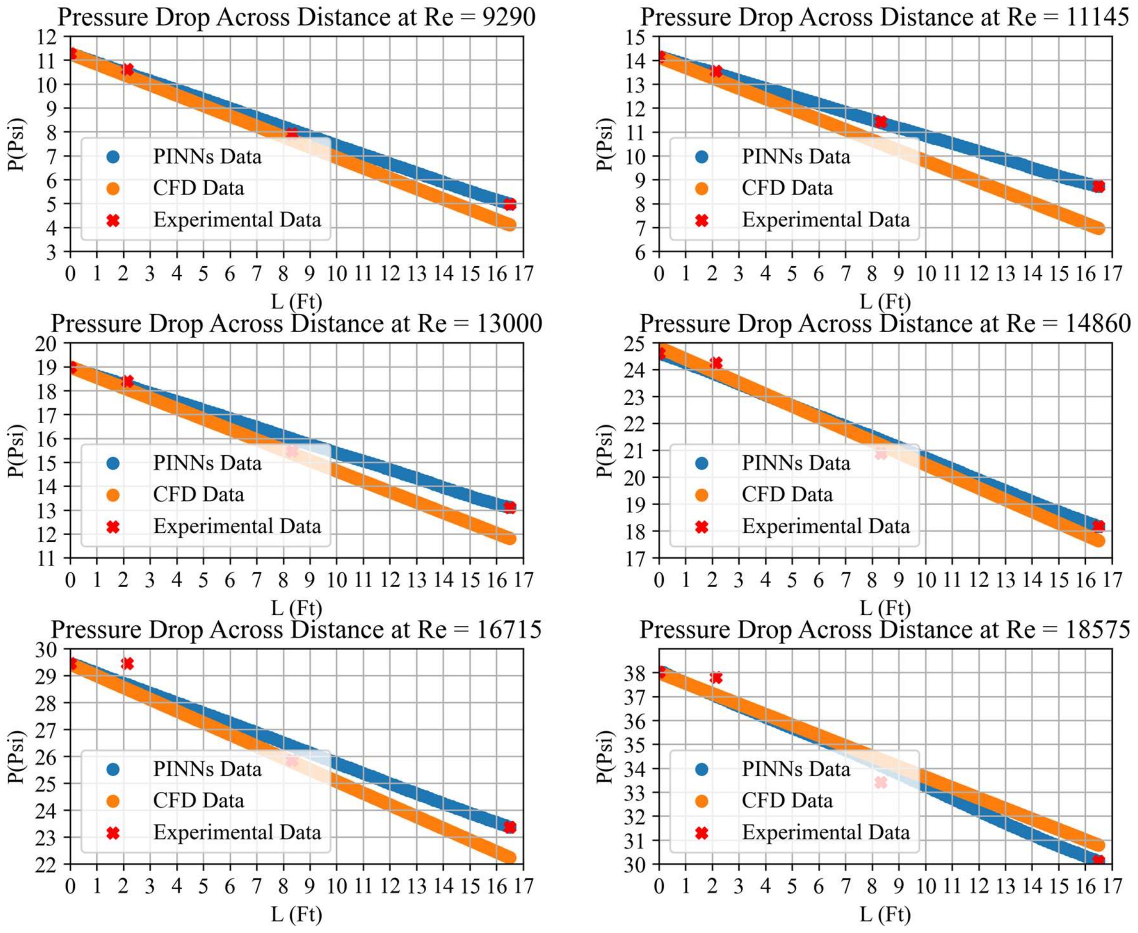


Figure 6. Pressure Drop at Turbulent Flow Region

IV. Conclusion

This study showcases the effectiveness of Physics-Informed Neural Networks (PINNs) in predicting pressure drops in turbulent annular flows, confirming their accuracy against both experimental data and computational fluid dynamics (CFD) simulations. Using the fundamental Reynolds-Averaged Navier-Stokes (RANS) equations, PINNs offer a simplified and accurate alternative to traditional CFD methods. At higher Reynolds numbers, the agreement between the data and experimental results improves notably. The regression coefficient (R^2) for the CFD lies between 0.78 and 0.98, while for the PINNs, it ranges from 0.97 to 0.99, highlighting a superior fit for the PINNs model, with the added benefit of being mesh-free. The experimental setup facilitated validation and demonstrated strong alignment between PINNs, ANSYS simulations, compare to actual measurements, even without including gravity terms in the PINNs. This suggests that PINNs are a promising alternative to conventional CFD models, offering significant advantages in computational efficiency and simplicity. Their successful validation against experimental data establishes PINNs as a reliable method for predicting pressure drops in turbulent annular flows, with potential applications in fields such as oil drilling and beyond.

References

[1] J. Ling and J. Templeton, "Reynolds Averaged Turbulence Modeling using Deep Neural," *Journal of Fluid Mechanics,* vol. 807, pp. 155-166, 2016. J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.

[2] C. Jiang, R. Vinuesa, R. Chen, J. Mi, S. Laima and H. Li , "An interpretable framework of data-driven turbulence modeling using deep neural networks," *Physics of Fluids,* vol. 5, p. 33, 2021.

[3] P. Srinivasan, L. Guastoni, H. Azizpour, P. Schlatter and R. Vinuesa, "Predictions of turbulent shear flows using deep neural networks," *Physical Review Fluids,* no. 4, 054603.

[4] H. Eivazi, H. Veisi, M. Naderi and V. Esfahanian, "Deep neural networks for nonlinear model order reduction of unsteady flows," Physics of Fluids, vol. 32, no. 105104, 2020.

[5] "Recurrent neural networks and koopman-based frameworks for temporal predictions in a low-order model of turbulence," *International Journal of Heat and Fluid Flow,* no. 108816, 2012

[6] T. Murata, K. Fukami and K. Fukagata, "Nonlinear mode decomposition with convolutional neural networks for fluid dynamics," *Journal of Fluid Mechanics,* no. 882, A13, 2020

[7] L. Guastoni, . A. Güemes, A. Ianiro, S. Discetti, P. Schlatter, H. Azizpour and R. Vinuesa, "Convolutional-network models to predict wall-bounded turbulence from wall quantities," *Journal of Fluid Mechanics ,* no. 928: A27, 2021.

[8] A. Güemes, S. Discetti, A. Ianiro, B. Sirmacek, H. Azizpour and R. Vinuesa, "From coarse wall measurements to turbulent velocity fields through deep learning," *Physics of fluids,* no. 7, p. 33, 2021.

[9] H. Tang, J. Rabault, A. Kuhnle, Y. Wang and T. Wang, "Robust active flow control over a range of Reynolds numbers using an artificial neural network trained through deep reinforcement learning," *Physics of Fluids,* no. 32.5, 2020.

[10] F. Ren, J. Rabault and H. Tang, "Applying deep reinforcement learning to active flow control in weakly turbulent conditions," *Physics of Fluids,* no. 33.3.

[11] H. Taherdoost, "Deep Learning and Neural Networks: Decision-Making Implications," *Symmetry,* no. 9, 1723, 2023.

[12] R. Maziar, P. Perdikaris and G. Karniadakis, "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations," *Journal of Computational physics,* vol. 378, pp. 686-707, 2019.

[13] M. Raissi, "Deep Learning of Vortex Induced Vibrations," *Journal of Fluid Mechanics,* no. 861, pp. 119-137, 2019.

[14] M. Raissi, A. Yazdani and G. Karniadakis, "Hidden fluid mechanics: Learning velocity and pressure fields from flow visualizations," *Science,* no. 367.6481, pp. 1026-1030, 2020.

[15] M. Fathi, "Super-resolution and Denoising of 4D-Flow MRI Using Physics-Informed Deep Neural Nets," *Computer Methods and Programs in Biomedicine,* no. 197: 105729, 2020.

[16] A. Arzani, J.-X. Wang and R. DSouza, "Uncovering near-wall blood flow from sparse data with physics-informed neural networks," *Physics of Fluids,* no. 33.7, 2021.

[17] H. Eivazi, M. Tahani, P. Schlatter and R. Vinuesa, "Physics-informed neural networks for solving Reynolds-averaged Navier–Stokes equations," *Physics of Fluids ,* no. 34.7, 2022.

[18] A. Amin, " Design, development and control of a managed pressure drilling setup," in *Diss. Memorial University of Newfoundland*, 2017.

[19] A. Amin, S. Imtiaz, A. Rahman and F. Khan, "Nonlinear model predictive control of a Hammerstein Weiner model based experimental managed pressure drilling setup," *ISA transactions,* vol. 88, pp. 225-232, 2019.

[20] M. M. Huque, S. Butt, S. Zendehboudi and S. Imtiaz, "Systematic sensitivity analysis of cuttings transport in drilling operation using computational fluid dynamics approach," *Natural Gas Science and Engineering,* no. 103386, p. 81, 2020.

[21] Ansys, ANSYS 2022/R2, 2022 ANSYS, Inc.

[22] I. Celik, U. Ghia and R. Patrick, "Procedure for Estimation and Reporting of Uncertainty Due to Discretization," *Journal of Fluids Engineering,* p. 130(7), 2008.

[23] W. Graebel, Advanced fluid mechanics, Academic Press, 2007.

[24] H. K. Versteeg, An introduction to computational fluid dynamics: the finite volume method, Pearson education, 2007.

[25] F. White, Viscous Fluid Flow, McGraw Hill, 2011.

[26] ANSYS, ANSYS Theory Guide 2021 R2, ANSYS, Inc., 2021.

[27] H. Eivazi, M. Tahani, P. Schlatter and R. Vinuesa, "Physics-informed neural networks for solving," *Physics of Fluids,* vol. 7, p. 34, 2022.

[28] X. Jin, S. Cai, H. Li and G. Karniadakis, "NSFnets (Navier-Stokes flow nets): Physics-informed neural networks for the incompressible Navier-Stokes equations," *Journal of Computational Physics,* p. 426 : 109951., 2021.