



A Fully Implicit Reservoir Simulation Using Physics Informed Neural Networks

Agus Wahyudi^{1,2}, Tutuka Ariadji¹, Taufan Marhaendrajana¹, Kuntjoro Adji Sidarto¹, and
Zuher Syihab¹

¹Institut Teknologi Bandung
Ganesha Street No. 10 Bandung, West Java, 40132, Indonesia.

²Politeknik Energi dan Mineral Akamigas
Gajah Mada Street Karangboyo, Cepu, Central Java, 58315, Indonesia.

Corresponding author: awahyudi@esdm.go.id.

Manuscript received: June 27th, 2025; Revised: July 23th, 2025
Approved: August 07th, 2025; Available online: October 15th, 2025; Published: October 15th, 2025.

ABSTRACT - The accuracy of multiphase flow simulation in porous media is critical for reservoir management but the achievement is challenging due to the nonlinear, coupled nature of the governing equations and truncation errors inherent in mesh-based numerical solvers. Therefore, this study aims to present a mesh-free, fully implicit Physics-Informed Neural Network (PINN) framework for two-phase immiscible oil–water flow. The framework has feedforward neural networks which simultaneously approximate the continuous pressure and saturation fields while embedding the governing partial differential equations (PDEs), boundary conditions, and initial conditions directly into the loss function. Moreover, three network topologies in the form of single-row (N1), dual-row (N2), and branched-layer (NY) are evaluated across nine configurations with different architectural variants. The key novelty is in the fully implicit formulation of branched network architectures which effectively minimizes interference between pressure and saturation predictions. The benchmarking against the commercial simulator Eclipse[®] shows that NY configuration achieves the best performance with a mean squared error below 1.0e-10. N1 architecture also provides superior stability across successive time steps, while N2 exhibits slower convergence. Deep and narrow architectures produce higher accuracy but require approximately twice the computational cost per iteration. The results show that the proposed PINN-based framework produces high-fidelity solutions for complex reservoir problems without depending on spatial meshing despite higher computational demands. This offers a promising alternative to conventional numerical methods for both regular and irregular geometries.

Keywords: reservoir simulation, fully implicit, physics-informed neural networks.

© SCOG - 2025

How to cite this article:

Agus Wahyudi, Tutuka Ariadji, Taufan Marhaendrajana, Kuntjoro Adji Sidarto, and Zuher Syihab, 2025, A Fully Implicit Reservoir Simulation Using Physics Informed Neural Networks, Scientific Contributions Oil and Gas, 48 (3) pp. 131-147. DOI [org/10.29017/scog.v48i3.1858](https://doi.org/10.29017/scog.v48i3.1858).

INTRODUCTION

The flow of fluids through porous media is a fundamental phenomenon in several natural and industrial processes, including groundwater hydrology, oil recovery, and chemical engineering. Porous media flow is the movement of fluids through materials containing interconnected voids or pores such as soil, sandstone, or synthetic membranes. In petroleum engineering and hydrogeology, the understanding of multiphase flow with a specific focus on the simultaneous flow of oil and water is very important for effective reservoir management and environmental remediation (Cai & Berg 2025). The flow of two immiscible fluids such as oil and water through a porous medium causes an interaction with the solid matrix and each other (Buckley & Leverett 1942). This behavior is governed by several key phenomena. For example, the interfacial tension and wettability often experienced lead to the tendency of one fluid, typically water, to wet the surface of the pores and produce capillary forces that influence phase distribution (Alizadeh & Fatemi 2021). The ability of each fluid to flow is reduced due to the presence of the other phase which is quantified by relative permeability functions (Corey et al., 1956; Goda & Behrenbruch 2004). This shows that each fluid occupies a fraction of the pore volume defined as saturation. Moreover, the sum of the saturations of oil and water is required to equal unity.

The equations governing fluid flow in porous media, including single-phase and multiphase, are typically described by nonlinear and coupled partial differential equations (PDEs). However, analytical solutions are generally not feasible due to the nonlinear and coupled nature of the equations, except in highly simplified cases (Abou-Kassem et al., 2020). This leads to the preference for numerical methods. For example, Finite Difference Method (FDM) and Finite Volume Method (FVM) are common discretization methods for spatial domains. Implicit Pressure, Explicit Saturation (IMPES) and Fully Implicit Method (FIM) schemes are widely used time-integration methods (Jenny et al., 2006; Moncorgé et al., 2018; Younis et al., 2010). These numerical solutions are approximations that introduce errors due to truncation of the Taylor series during discretization and linearization compared to the exact analytical solutions (Tijink & Cottier 2019). Therefore, boundary (Dirichlet or Neumann) and initial conditions (initial saturation

and pressure distributions) need to be specified in the process of solving the models. Two-phase flow of oil and water in porous media is a complex but well-studied process governed by conservation laws and Darcy's law with important applications in energy and environmental fields. The accurate modeling requires detailed characterization of the porous medium and fluid properties as well as the adoption of robust mathematical methods for solving the coupled nonlinear equations.

Artificial Neural Networks (ANNs) have shown considerable potential in solving PDEs by leveraging their capacity as universal function approximators (Long et al., 2019; Sirignano & Spiliopoulos 2018). This is associated with the ability to offer a mesh-free framework in providing solutions compared to traditional numerical methods which depend on discretization techniques such as finite difference or finite element (Berrone & Pintore 2024; Diab et al., 2022). The process is achieved by approximating the solution function over the domain directly through training a neural network to satisfy the governing equations as well as the boundary and initial conditions (Lu et al., 2021).

The key advantage of using ANNs for PDEs is in the ability to generalize over the input domain and represent complex nonlinear mappings with a relatively compact model (Raissi et al., 2019). This leads to the specific usefulness for high-dimensional problems and scenarios related to sparse or scattered data where conventional methods are computationally expensive or unstable. Moreover, ANNs can be trained to provide continuous, differentiable solutions across the entire domain which is advantageous for problems requiring gradient information.

A foundational work in this domain was conducted by Lagaris et al. who introduced a method for solving both ordinary and PDEs using feedforward neural networks (Lagaris et al., 1998). The method focused on training the network to minimize the residual of PDEs and inherently satisfied boundary conditions through a carefully designed trial function. This marked an important step in integrating machine learning methods into numerical analysis.

More recent developments include Physics-Informed Neural Network (PINN) framework where the loss function of ANNs is constructed to include PDEs residuals and any relevant physical constraints using automatic differentiation (Raissi et al., 2019).

This method has shown success in different forward and inverse problems associated with complex PDEs in fields such as fluid dynamics, elasticity, and heat conduction. ANNs have shown promise in approximating the solutions of such complex systems due to the ability to learn from data and capture nonlinear relationships without depending on mesh-based discretization. In the context of porous media, this capability becomes specifically valuable when focusing on heterogeneous properties, irregular geometries, and limited observational data. Recent studies have successfully applied PINN to simulate fluid flow in porous structures. For example, Zhu et al. (2021) applied PINN to model single-phase flow through heterogeneous porous media by encoding the governing PDEs and boundary conditions into the neural network loss function. Zhang et al. (2024) extended the method to multiphase flow scenarios to show the ability in capturing saturation fronts and pressure distributions effectively. ANNs were also useful in building the model for the prediction of wax deposition rate in two-phase flow, total organic carbon and forecasting reservoir performances during the carbon capture, utilization, and storage (Iskandar & Kurihara 2022; Septiano et al., 2021; Wardhana et al., 2021). The results show that ANNs and PINN are promising alternatives to traditional numerical solvers for simulating porous media flows with advantages in flexibility, generalizability, and the ability to incorporate sparse data (Fraces et al., 2020; Fraces & Tchelepi 2021; Fuks & Tchelepi 2020; Shukla et al., 2021).

The application of PINN to solve PDEs has experienced rapid growth but the adoption for strongly coupled nonlinear systems such as two-phase flow in porous media remains challenging. A critical and underexplored aspect is the design of optimal neural network architectures for those types of systems. Specifically, the standard Fully Connected Neural Network (FNN) with multiple outputs is often adopted to simultaneously predict pressure and saturation distributions. However, the variables exhibit fundamentally different characteristics. This is observed from the fact that pressure is typically elliptic in nature and tends to be smooth and global while saturation is hyperbolic, often exhibits sharp fronts, and is more locally influenced. The efforts to force a single, shared-parameter backbone network to learn the disparate physical behaviors can introduce a high risk of feature interference and gradient conflict during training. The process is capable of leading to

unstable convergence or failure to find a solution that satisfies both governing equations simultaneously. Therefore, a significant study gap exists in the development and evaluation of specialized FNN topologies with a specific focus on branched or multi-path architectures that can effectively minimize interference between the pressure and saturation solutions to enhance the robustness and accuracy of PINN for fully implicit, two-phase flow simulations.

This study investigates the application of PINN for mesh-free simulation of two-phase flow in porous media. The primary objective is to solve coupled nonlinear PDEs governing fluid flow to obtain simultaneous, fully implicit solutions for the spatial distributions of pressure and saturation. A critical secondary objective is to determine the optimal FNN architecture for the multi-physics problem. This requires a comparative analysis of network topologies with an emphasis on the efficacy of monolithic versus branched designs to mitigate gradient conflict between the distinct physical variables and ensure robust convergence. The model incorporates well constraints with source and sink terms representing an injector and a producer. The spatial domain is treated as continuous to eliminate the need for a mesh-grid method as required in conventional reservoir simulations. Therefore, this method avoids truncation errors and improves solution accuracy. The predictive accuracy and convergence performance of different architectures evaluated based on hyperparameters such as depth and width are rigorously benchmarked against solutions from the industry-standard numerical simulator known as Eclipse.

Background

The governing equation for two-phase flow in porous media is achieved by combining mass conservation and Darcy's law (Abou-Kassem et al., 2020). The fluid phases of oil and water as well as the capillary pressure are considered to be neglected. Therefore, Darcy's equations and the standard black oil fluid model are substituted into the continuity equations to produce the following mass balance for each phase as in Equations 1 and 2.

$$\nabla \cdot \left(K \frac{\rho_l k_{rl}(S_l)}{\mu_l} \nabla p \right) = \frac{\partial(\rho_l \phi S_l)}{\partial t} \quad (1)$$

$$\sum_{l=o,w} S_l = 1 \quad (2)$$

where, K represents the absolute permeability vector of the reservoir rock and the relative permeability of a fluid phase is dependent on its saturation. Density viscosity, and porosity ϕ varies with pressure. The subscript l denotes the liquid phase which includes oil o and water w . Moreover, ∇ signifies the spatial gradient and $\partial/\partial t$ is the partial time derivative. In a displacement process where water replaces oil in a fully oil-saturated porous medium, two-phase flow behavior produced is characterized by the drainage relative permeability curve. The relative permeability of oil and water was empirically formulated by Corey (Corey et al., 1956), (Goda & Behrenbruch 2004) and presented in Equations 3 and 4:

$$k_{ro} = k_{ro,max} \left(\frac{S_o - S_{or}}{1 - S_{or} - S_{wc}} \right)^{n_o} \quad (3)$$

$$k_{rw} = k_{rw,max} \left(\frac{S_w - S_{wc}}{1 - S_{or} - S_{wc}} \right)^{n_w} \quad (4)$$

A standard method in petroleum reservoir simulation known as fluid compressibility is integrated in the black oil model to account for pressure-dependent changes in fluid density and volume. Each fluid in a two-phase oil-water is considered slightly compressible and is characterized using pressure-dependent properties.

$$c_l = \frac{1}{B_l} \left(\frac{\partial B_l}{\partial P} \right)_T \quad (5)$$

The isothermal compressibility of a fluid phase c_l is the relative volume change with pressure at constant temperature. Meanwhile, the formation volume factor B_l varies with pressure P .

$$c_r = \frac{1}{\phi} \left(\frac{\partial \phi}{\partial P} \right)_T \quad (6)$$

The porosity of a rock which is represented by ϕ describes the fraction of the rock volume considered pore space and varies as a function of pressure. The change in porosity due to pressure is governed by the rock compressibility c_r .

The governing equations for fluid flow in porous media are typically formulated as a combination of boundary and initial value problems. An initial value problem refers to a situation where the solution of the governing equations depends on specified conditions

at an initial time. This often shows that the initial distributions of pressure, saturation, or other relevant state variables need to be known in the spatial domain at the start of the simulation. Meanwhile, a boundary value problem focuses on specifying the behavior of the solution along the spatial boundaries of the domain. The existence in porous media includes prescribed pressures, flow rates, or no-flow conditions at the reservoir boundaries, injection wells, or production wells. The boundary conditions are important for determining how the fluid interacts with the surrounding environment and govern the movement in the domain.

Fully Implicit Method (FIM) in reservoir simulation is a numerical method often used in reservoir simulation to solve the coupled systems of nonlinear PDEs governing multiphase fluid flow in porous media. It is called fully implicit because all variables such as pressure, saturation, composition, etc. are treated implicitly in time to ensure unconditional stability for large timesteps but the process is computationally expensive. All terms are evaluated at the next timestep $t+1$

$$\begin{aligned} & \nabla \cdot \left(K \frac{\rho_l k_{rl}(S_w)^{t+1}}{\mu_l^t} \nabla P^{t+1} \right) \\ &= \frac{(\rho_l \phi S_l)^{t+1} - (\rho_l \phi S_l)^t}{\Delta t} \end{aligned} \quad (7)$$

The domain is discretized into grid blocks and the equations are integrated over each control volume. Therefore, the discretized equations are written as residuals:

$$F(\mathbf{b}^{t+1}) = 0, \quad \mathbf{b} = [P, S_w, S_o] \quad (8)$$

where, is the residual form of the governing equation of fluid flow through porous media and is a vector of variables of pressure and saturation. The equation is solved through Newton-Raphson as follows:

$$J^k \delta \mathbf{b}^k = -F^k \quad (9)$$

$$\mathbf{b}^{k+1} = \mathbf{b}^k + \delta \mathbf{b}^k \quad (10)$$

Where, J is the Jacobian matrix containing partial derivatives of all equations with respect to all variables.

METHODOLOGY

A conceptual model was proposed to develop fluid flow solutions using PINN based on the five-spot waterflooding injection pattern in a two-dimensional (2D) plane as presented in Figure 1. The simulation area included a single sector with production-injection pairs positioned at opposite corners of a square-shaped area. The reservoir fluid consisted of a two-phase immiscible fluid in the form of oil and water which were treated as slightly compressible. The model assumed there was no dissolved gas in either the oil or water phases. Moreover, gravitational acceleration was neglected in the 2D aerial model.

The pattern of a fluid flow around a well is radial due to the movement from the surrounding radius toward the wellbore. Consequently, the well blocks were modeled separately from the reservoir domain and used to represent the surrounding area with homogeneous physical properties. This model was adapted from a similar block in numerical reservoir simulation which also had a steady-state radial flow. A previous study developed a model in numerical simulation from a vertical well in a reservoir with a focus on the inflow performance relationship (Peaceman 1983). The fluid flowed towards a radial center at one corner of the reservoir domain. The Darcy velocity in the radial direction at the block interface was subsequently described as a linear velocity along the axis.

Neural networks were trained to approximate the pressure and saturation field (11) at a given time step. The inputs are subsequently defined by 2D spatial variables as follows:

$$\hat{P}, \hat{S}_w \approx \hat{u}(x, y, \theta) \quad (11)$$

The spatial domain was treated as continuous to eliminate the need for a mesh-grid method observed in conventional reservoir simulations. Moreover, the temporal domain was discretized in adherence to standard reservoir simulation frameworks where solutions were computed at predefined time steps. This discretization reduced the dimensions and computing time of the problem. The governing equation for each liquid phase was formulated into a residual function as presented in Equation 12.

$$\mathcal{R}(P, S_l) = \nabla \cdot \left(\frac{k_{rl}(S_l^{t+1})}{B_l^t \mu_l} \nabla P^t \right) - \frac{\left(\frac{\phi S_l}{B_l} \right)^{t+1} - \left(\frac{\phi S_l}{B_l} \right)^t}{\Delta t} \quad (12)$$

The residual value was evaluated at collocation points (x_j, y_j) .

$$\mathcal{L}_{PDE,l} = \frac{1}{N_d} \sum_{j=1}^{N_d} \left(\mathcal{R}(\hat{P}(x_j, y_j), \hat{S}_l(x_j, y_j)) \right) \quad (13)$$

$(x_j, y_j) \in \Omega$

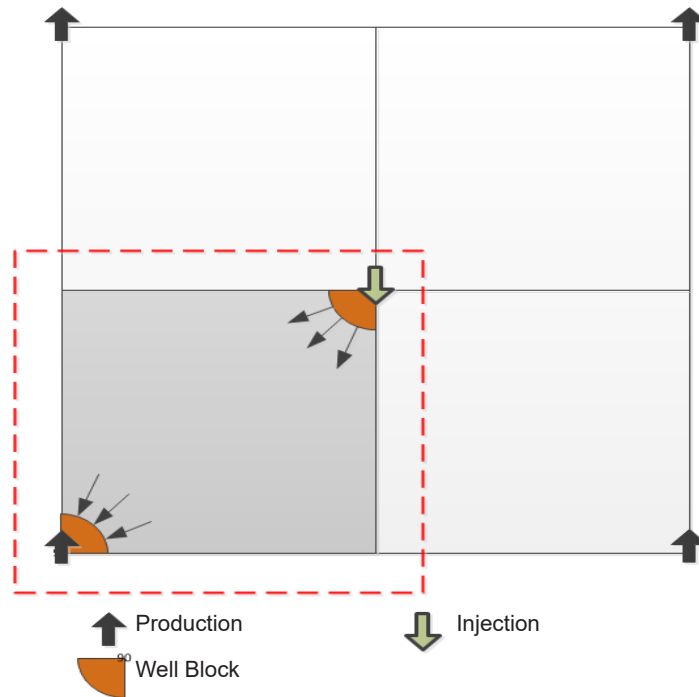


Figure 1. Conceptual reservoir model for two-dimensional linear flow

PDEs loss function of the fluid phase was formulated as the mean square error computed from the residuals of the governing equations. It was systematically evaluated at a set of predefined collocation points in the domain Ω .

The initial and boundary conditions were necessary to complete PDEs fluid flow equation. ANNs were trained on labeled data representing the initial conditions in order to estimate the pressure distribution and the saturation distributions $\hat{S}_w^0$.

$$\hat{P}^0, \hat{S}_w^0 \approx \hat{u}^0(x, y, \theta) \quad (14)$$

At the reservoir boundary, Dirichlet boundary conditions were applied to each liquid phase as presented in Equation (15). The two types of boundary conditions applied were constant flow rate at the wells (inner boundary) and no-flow at the reservoir boundaries (outer boundary).

$$\mathcal{L}_{BC,n,l} = \frac{1}{N_b} \sum_{k=1}^{N_b} \left(\frac{\partial \hat{P}}{\partial n}(x_k, y_k) + \frac{\mu_l v_{n,l}}{K_n k_{rl}} \Big|_k \right)^2 \quad (15)$$

$(x_k, y_k) \in \partial\Omega$

The boundary condition loss function of the fluid phase in a linear direction presented as in was formulated as the mean square error computed from the residual values of the boundary conditions equations from collocation points within the domain boundary. Moreover, the radial well blocks had a constant flow rate normal to the arc of the well block which allowed the flow velocity vector to be decomposed along the Cartesian axes. The constant flow rate boundary conditions were applied to the well block or inner boundary with the loss function expressed in the production well as presented in Equation (16) and the injection well in Equation (17). The inner boundary loss function corresponded to the fluid phase of each well with the oil phase for production wells and the water phase for injection wells.

$$\mathcal{L}_{BC,PRD} = \mathcal{L}_{PRD,o_x} + \mathcal{L}_{PRD,o_y} \quad (16)$$

$$\mathcal{L}_{BC,INJ} = \mathcal{L}_{INJ,w_x} + \mathcal{L}_{INJ,w_y} \quad (17)$$

No-flow boundary conditions were applied on all four sides of the reservoir boundary as a loss function of the outer boundary, including the west, north, east

and south. At zero flow rates, the pressure gradient was equal to zero. The outer boundary loss function, was the sum of the mean square errors of the four boundary conditions as presented in (18). This was in line with the normal flow direction towards the reservoir boundary for each phase. The west and east sides were in the x direction while the north and south were in the y direction.

$$\begin{aligned} \mathcal{L}_{BC,OB} = & \mathcal{L}_{w,o_x} + \mathcal{L}_{w,w_x} + \mathcal{L}_{n,o_y} + \mathcal{L}_{n,w_y} \\ & + \mathcal{L}_{e,o_x} + \mathcal{L}_{e,w_x} + \mathcal{L}_{s,o_y} + \mathcal{L}_{s,w_y} \end{aligned} \quad (18)$$

The steps to determine a solution using PINN algorithm are outlined in Figure 2. First, a neural network $\hat{u}(x, y, \theta)$ was constructed to approximate the solution $P(x, y)$ and $S_w(x, y)$. The network parameters $\theta = \{w^\ell, b^\ell\}$ for $1 \leq \ell \leq L$ consisted of the weight matrices and bias vectors of the neural network used in the approximation $\hat{u}$. The next step was to constrain the neural network in order to satisfy the physical laws governed by PDEs and the boundary conditions. The training dataset consisted of two subsets which were the points in the interior domain and those on the boundary. These were generated using randomly sampled collocation points. Furthermore, an optimization procedure was executed to identify optimal parameters θ by minimizing the loss function $\mathcal{L}(x, y, \theta)$ through the process known as training. The highly nonlinear and non-convex nature of the loss landscape with respect to θ led to minimization which was performed using gradient-based optimization methods such as Adam, BFGS and L-BFGS (Lu et al., 2021).

The fully implicit method requires the neural network to simultaneously solve the coupled distributions of pressure and saturation at the same time. ANNs shared the same input layers but had separate output layers for pressure and saturation distribution, respectively. These two outputs represented distinct problem-solving characteristics which included one for flow problems and the other for transport problems. Meanwhile, the solution search for the two problems has the possibility of interfering with each other depending on the network topology adopted. This study presents three distinct ANNs topologies based on hidden layer configurations presented in Figure 3 and explained as follows:

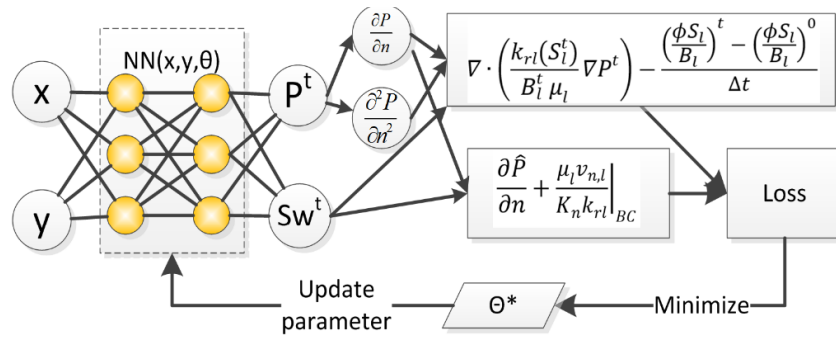


Figure 2. Workflow to solve two-phase fluid flow through porous media equations using PINN

1). Single-row layers (N1): There was no separation between layers for solving pressure and saturation distributions because both were computed in the same layer; 2). Dual-rows layers (N2): The layer for pressure distribution solution was separated from the layer for saturation solution which led to two sets of hidden networks with the same input; 3). Branched layers (NY): A common initial layer followed the input layer and subsequently diverged into two dedicated branches for pressure and saturation solutions.

The achievement of optimal training of PINN generally requires normalizing inputs to a smaller range (e.g., $[0,1]$ or $[-1,1]$). This enhanced numerical stability, accelerated convergence, and mitigated risks such as vanishing or exploding gradients. PINN combined the losses from physical data, PDEs residuals and observational data. This was necessary because unnormalized input variables could lead to imbalanced residual scales to destabilize the optimization.

Each of the topologies includes additional layers after the input layer and before the output layer as shown in Figure 3(a). The layers rescaled the values to be in line with ANNs characteristics as follows: 1). Input rescaling layer, transformed the input coordinates (x, y) into the range $(0,1)$; 2). Output rescaling layers mapped the $(0,1)$ output range back to the physical scales of solution distributions where was for pressure distribution and for saturation distribution.

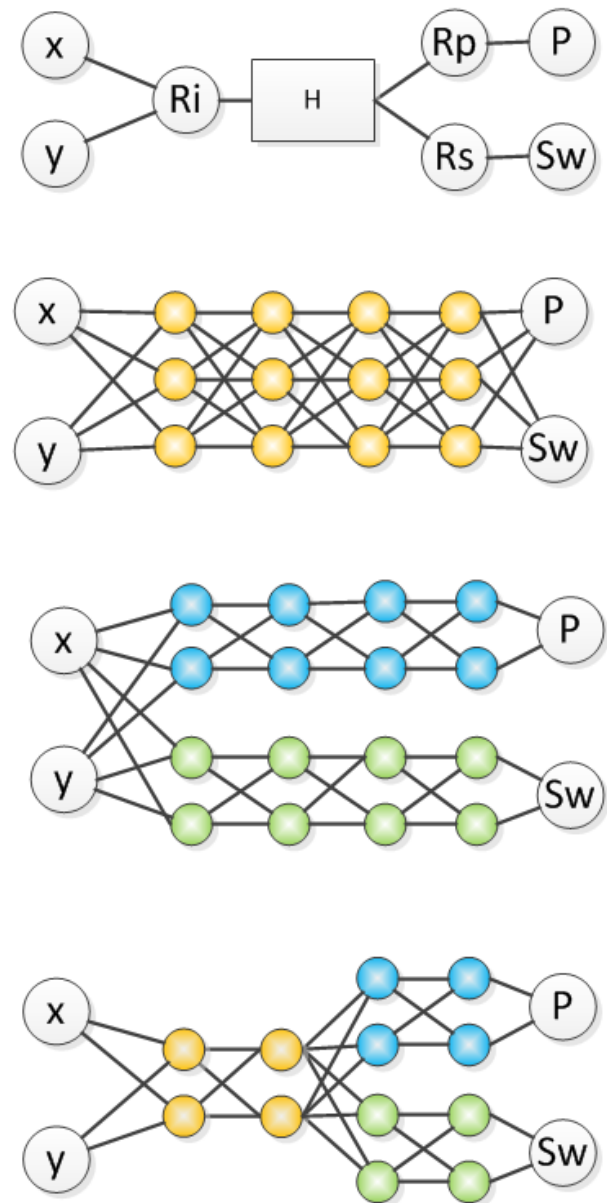


Figure 3. Neural Network topology: (a) Basic network configuration, (b) single row layer N1, (c) dual row layer N2, (d) branched layer NY

RESULT AND DISCUSSION

The three topologies mentioned were implemented into nine neural network configurations with each having varying numbers of hidden layers and neurons per layer. The number of layers and neurons differed but the total number of weights and biases did not vary significantly. The smallest number of weights and biases was in model N2-1 with 4,284 variables while the largest was in N2-3 with 4,366.

In N1 topology, the network chain was not separated into the network for solving pressure and saturation distributions. This led to the production of two configurations in the form of models N1-1 and N1-2. Meanwhile, N2 separated the network chain into distinct networks which led to the production of three configurations, including models N2-1, N2-2, and N2-3.

Table 1. Number of hidden layers and variables

No	Model	No. of Hidden Layer			$\sum$ var
		Com	Pres	Sat	
1	N1-1	4	-	-	4365
2	N1-2	6	-	-	4349
3	N2-1	-	3	3	4284
4	N2-2	-	4	4	4364
5	N2-3	-	5	5	4366
6	NY-1	1	5	5	4302
7	NY-2	1	3	3	4344
8	NY-3	2	2	2	4293
9	NY-4	3	1	1	4287

The final NY topology featured a common layer that branched into pressure and saturation layers. This led to the design of four neural network configurations including NY-1, NY-2, NY-3, and NY-4. The number of layers and neurons per layer is presented in Table 1. The differences in the number of layers and neurons while a relatively similar number of variables was maintained led to different network architectures, including wide-shallow, narrow-deep, or intermediate structures. The performance of each topology and its configurations was subsequently evaluated based on convergence speed and accuracy.

Initial condition training

Initial conditions define the state of the reservoir at time zero before any production or injection starts in a two-phase oil-water reservoir simulation. The settings are the critical starting point for the

simulation to predict the behavior of the reservoir over time. The two most fundamental initial conditions are pressure and saturation. The fluids in the reservoir are typically assumed to be in a state of hydrostatic equilibrium which is uniform in a 2D aerial model. This shows that the initial saturation is not uniform. The water zone with a specific focus on the area surrounding an injection well is assigned a water saturation equal to . Meanwhile, the oil zone is initialized with the connate water saturation as presented in Figure 4.

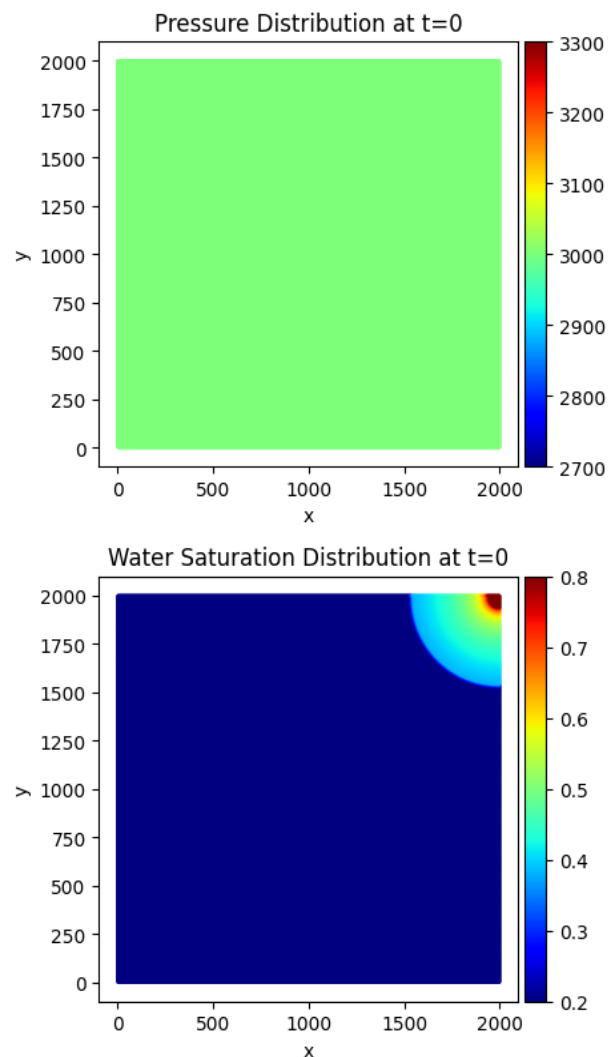


Figure 4. Initial conditions for pressure and water saturation distribution

Each NN model from the three topologies was trained to approximate the initial conditions using BFGS optimizer with 20 iterative optimization steps. The loss function history of the initial conditions is presented in Figure 5 while the final loss values are summarized in Table 2.

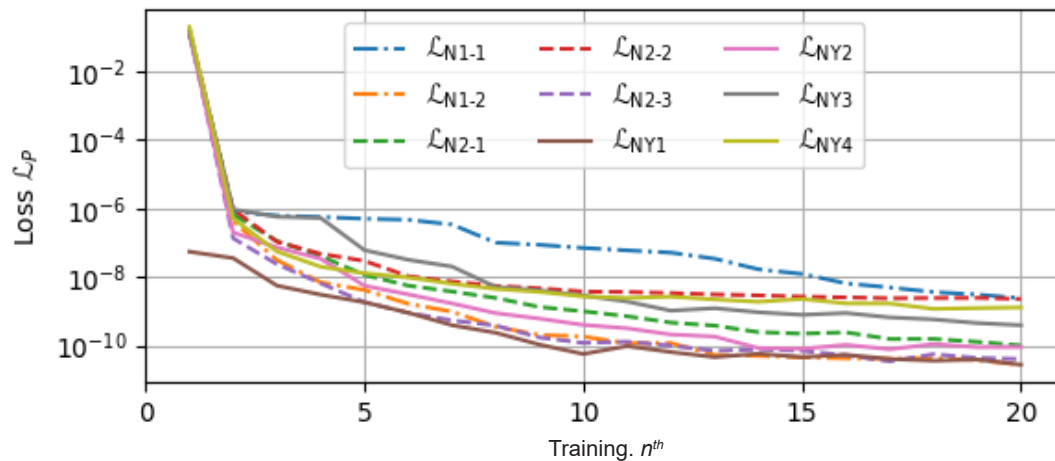


Figure 5. Initial BC Loss History

Table 2. Initial condition loss function values

Model	$\mathcal{L}_{p0}$	$\mathcal{L}_{Sw0}$	$\mathcal{L}_{p0} + \mathcal{L}_{Sw0}$
N1-1	7.934e-11	2.325e-09	2.404e-09
N1-2	1.181e-13	2.738e-11	2.750e-11
N2-1	5.657e-14	1.009e-10	1.009e-10
N2-2	2.793e-15	2.289e-09	2.289e-09
N2-3	6.311e-15	4.047e-11	4.048e-11
NY1	2.566e-15	2.771e-11	2.771e-11
NY2	2.706e-15	9.097e-11	9.098e-11
NY3	3.266e-15	3.916e-10	3.916e-10
NY4	1.580e-13	1.289e-09	1.289e-09

The loss curve shows that model NY-1 achieves the fastest convergence by maintaining relatively low loss values from the initial iterations. The information in Table 1 showed that three models exhibited significantly small loss values with N1-2 recording 2.750×10^{-11} and NY-1 had 2.771×10^{-11} primarily due to the saturation (S_w) component. However, the comparison of the pressure (P) losses showed that NY-1 outperformed N1-2 with a smaller value. The low initial BC loss value showed the effectiveness of ANNs as an effective approximator for the solution. The loss values in Table 2 reflected the ability of NY to produce better approximations compared to N1 and N2. All initial models were subsequently trained in PINN framework to solve the pressure and saturation distributions for the next timestep. The training was governed by the two-phase (oil-water) flow equations in porous media.

Solutions using N1 topology

The two ANNs in N1 topology were trained to solve two-phase flow equations using PINN framework. This numerical model solved the governing equations for pressure and saturation distributions simultaneously within a single row computational layer. N1-1 featured a relatively shallow and wide architecture with fewer hidden layers but more neurons per layer. Meanwhile, N1-2 adopted a deeper and narrower topology that comprised more hidden layers with fewer neurons per layer.

The training process was conducted by applying BFGS optimizer to 70 iterations with the loss function history presented in Figure 6. The solutions at the 70th iteration are shown in Figure 7 with the pressure and saturation distributions from both N1-1 and N1-2 models presented alongside the numerical reference for comparison.

Both PINN solutions showed significantly similar pressure and saturation distributions which were in excellent agreement with the reference. N1-2 achieved faster convergence than N1-1 as evidenced by the lower loss function values particularly after the 13th iteration and production of solutions that more closely matched the numerical reference at the 70th iteration. The pressure solution from N1-1 exhibited minor discrepancies near the injection well area. The continuous optimization of N1-1 could achieve comparable convergence to N1-2 but would require additional iterations. The pressure distribution and water saturation from an aerial perspective of the final solutions for N1-2 are shown in Figure 8.

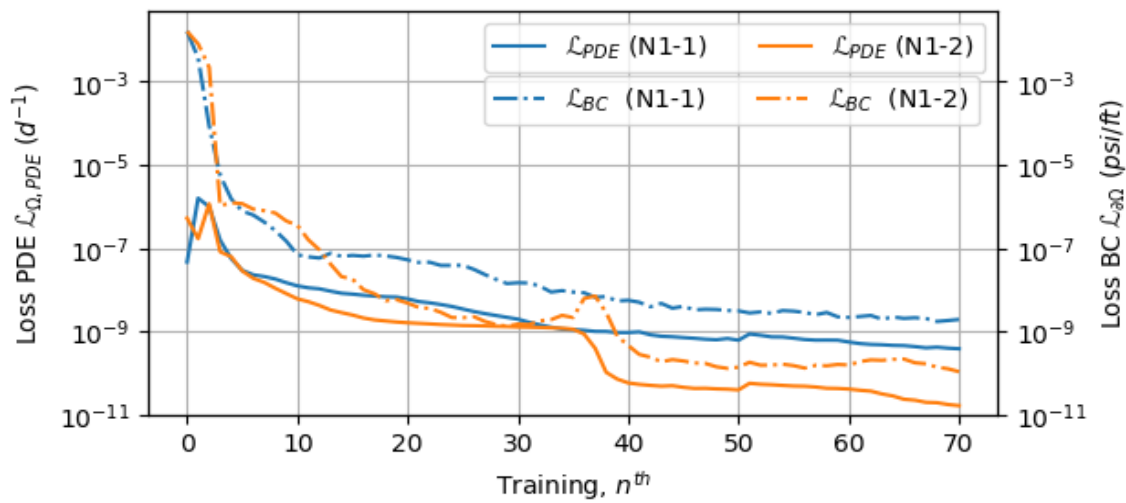


Figure 6. Loss history of N1 topology at timestep $t+I$

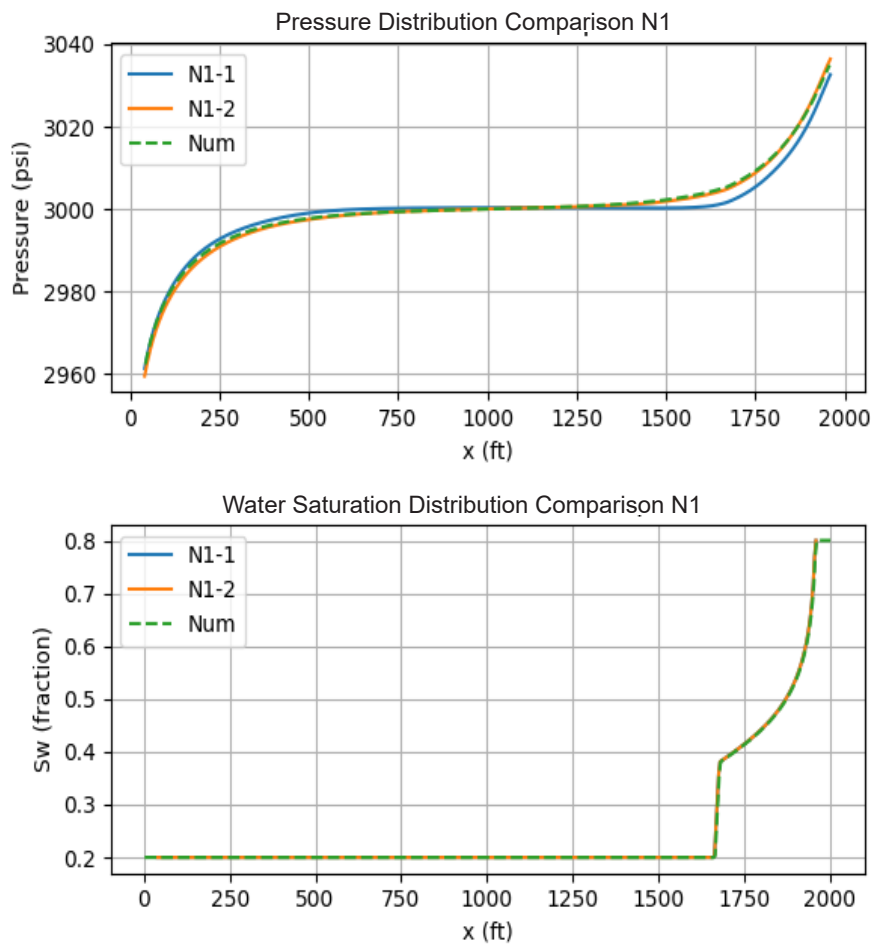


Figure 7. Solution of the N1 topology at timestep $t+I$

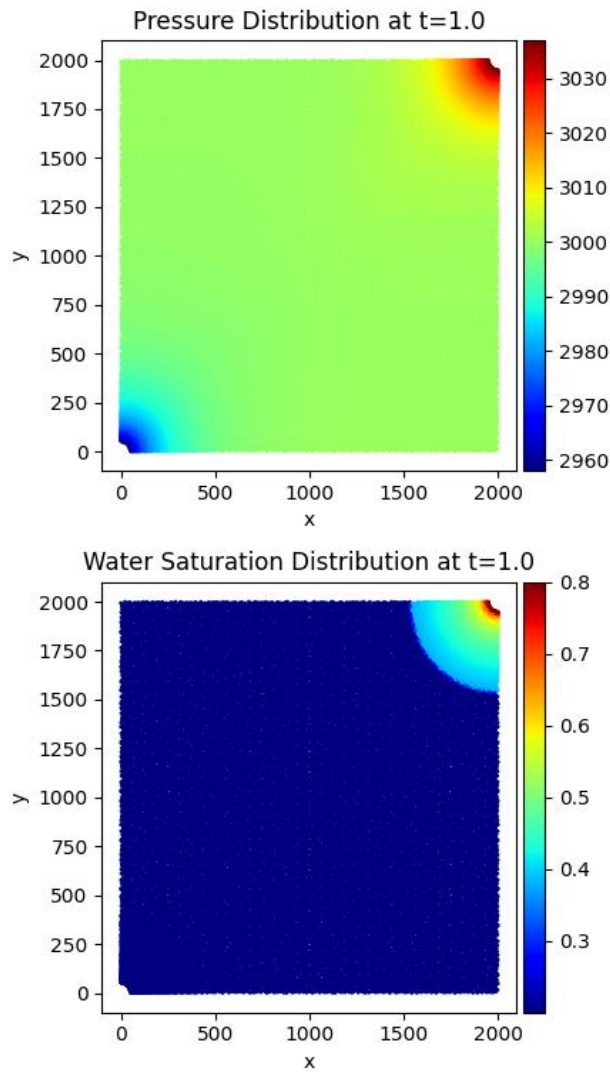


Figure 8. Solutions of the N1-2 model

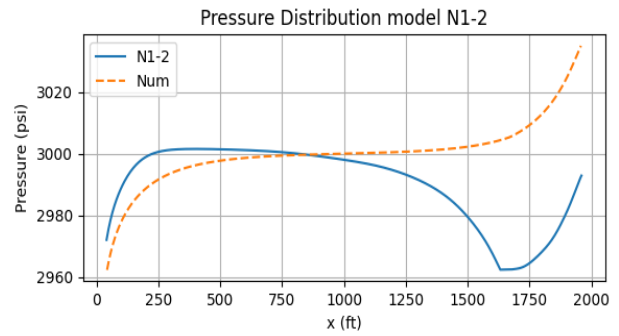
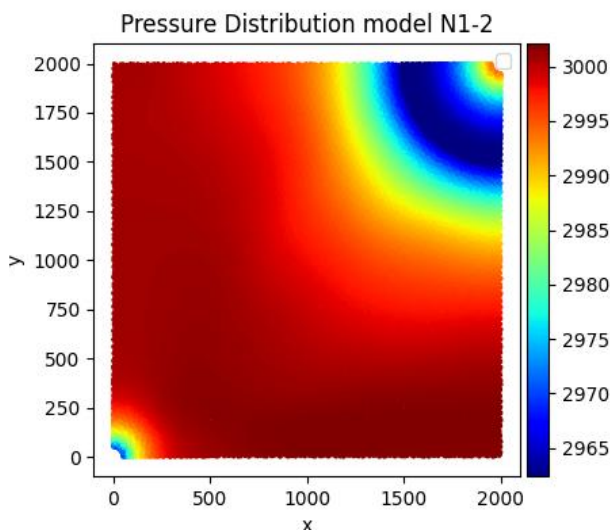


Figure 9. Interference on pressure distribution: (a). Effect of saturation distribution pattern (b). Comparison with reference solution

The results showed that the deeper and narrower architecture of N1-2 had superior performance in both accuracy and convergence speed for two-phase flow solutions compared to the shallower-wider N1-1 configuration. However, the model contained the risk of interference between the pressure and saturation solutions during training.

Solutions using N2 topology

N2 topology was implemented in three distinct ANNs architectures which were designated as N2-1, N2-2, and N2-3. N2-1 model featured a relatively shallow and wide architecture for both pressure and saturation chains while N2-3 adopted a deep and narrow topology for both network branches. N2-2 represented an intermediate configuration between these two extremes.

The training process was based on a two-stage method and each stage consisted of 100 iterative optimizations using BFGS algorithm. The loss function history is presented in Figure 10 while the solutions at the 100th iterations are in Figure 11 with a focus on the pressure and saturation distributions from all three N2 models alongside the numerical reference for comparative analysis.

The three N2 topology models produced PINN solutions with nearly identical pressure and saturation distributions. The curvature patterns of the solution distributions matched the reference but significant value discrepancies were observed. The loss function history curves showed slow convergence rates with minimal improvement evident when comparing solutions at iterations 50 and 100.

The achievement of converged solutions that properly matched the reference with the topology had a lower possibility. The process would require a substantially higher number of iterations to attain

the solutions. N2 architecture demanded significantly more iterations compared to the others but produced solutions with considerable deviations from the reference values.

Solutions using NY topology

This study developed four ANNs models with an NY topology to examine the effects of layer configuration, neuron count, and branching position. NY-1 had a deep, narrow architecture that consisted of one common layer followed by branching and five solution layers to produce a substantially longer solution network than the common model. NY-2 maintained a similar structure with one common layer and three solution layers but with a shorter, wider configuration. NY-3 showed a balanced layer distribution between common and solution networks. Meanwhile, NY-4 incorporated more common layers than solution layers. All the latter three models NY-2, NY-3, and NY-4 exhibited relatively shallow and wide architectural characteristics.

Training was conducted using BFGS optimizer for 50 iterations with the loss function history presented in Figure 12. The solutions obtained at the 50th iteration are shown in Figure 13 with a focus on the pressure and saturation distributions from all four NY models alongside the numerical reference.

Figure 13 shows that the curves of all the models exhibit excellent agreement with the reference from the numerical solution. The variations in the number

of layers and neurons had minimal influence on solution quality. Moreover, the deep-narrow NY-1 architecture produced curves which was most closely in line with the reference solution and represented the best-performing configuration. The model also achieved the lowest loss function values as reflected in the loss history. At timestep $t+1$, the NY topology generally showed faster convergence and higher accuracy compared to both N1 and N2.

The training results for all models at timestep $t+1$ are summarized in Figure 14. The boxplot in Figure 14(a) presents the average computation time per iteration while the bar chart shows the total number of iterations. Model N1-2 in N1 topology which produced a convergent solution required more computation time than N1-1. The same trend was observed in NY topology where NY-1 produced the highest accuracy but demanded longer training time than other comparable models. In N2 topology, the solution did not converge while NY-1 showed a higher average computation time per iteration than other models. Model configurations with deeper and narrower neural network layers (i.e., more layers with fewer units per layer) generally required longer training times. This was attributed to the extended computations in both the forward and backward propagation steps. Figure 14(b) further shows that deep-narrow architectures such as N1-2, N2-3, and NY-1 achieve higher accuracy compared to shallow-wide configurations.

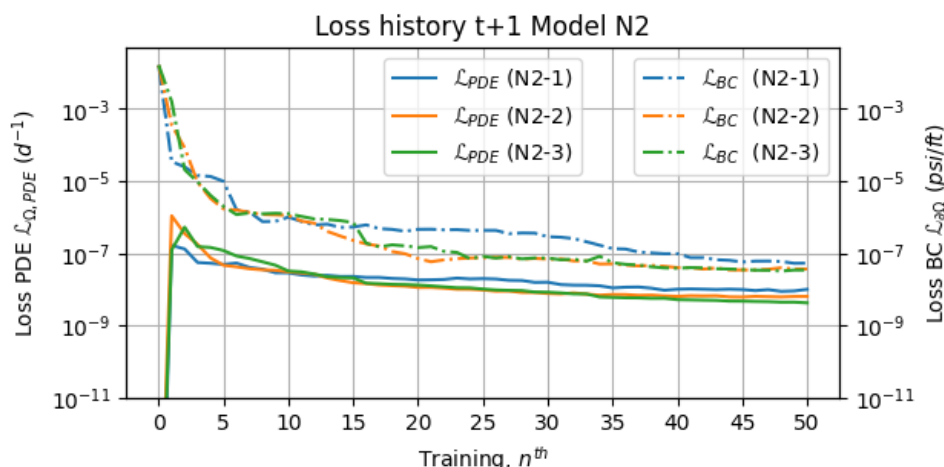


Figure 10. Loss history of the N2 topology at timestep $t+1$

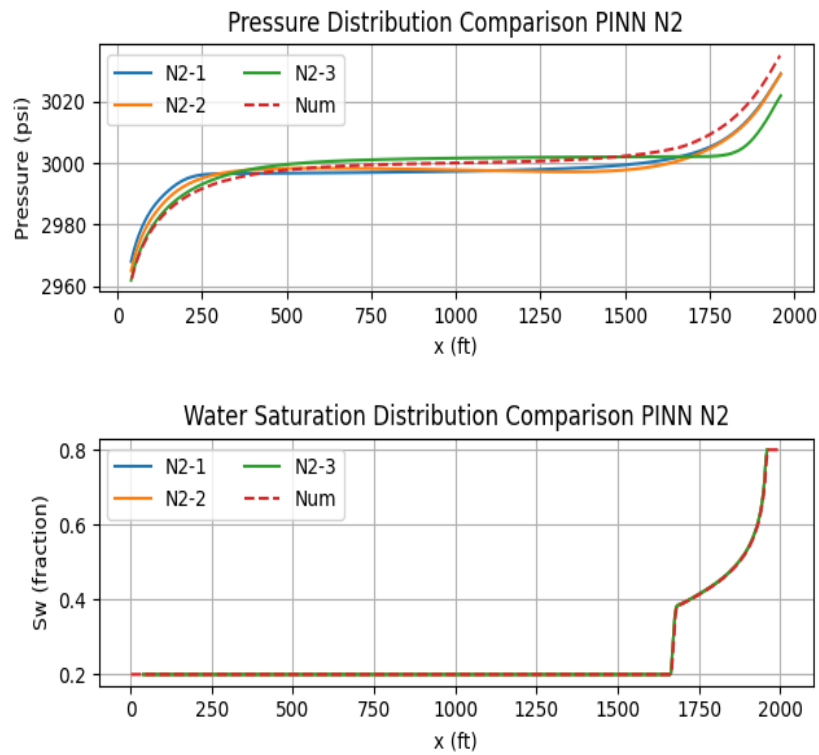


Figure 11. Solutions of the N2 topology at timestep $t+1$

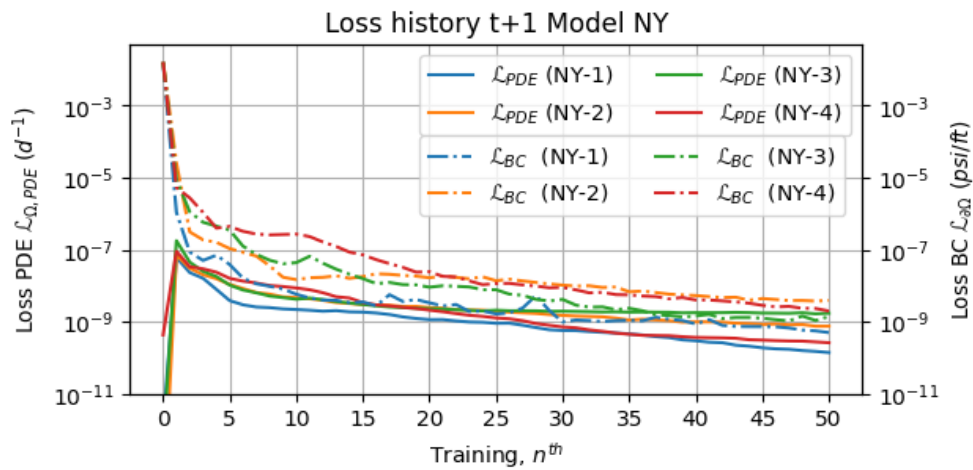


Figure 12. Loss history of the NY topology at timestep $t+1$

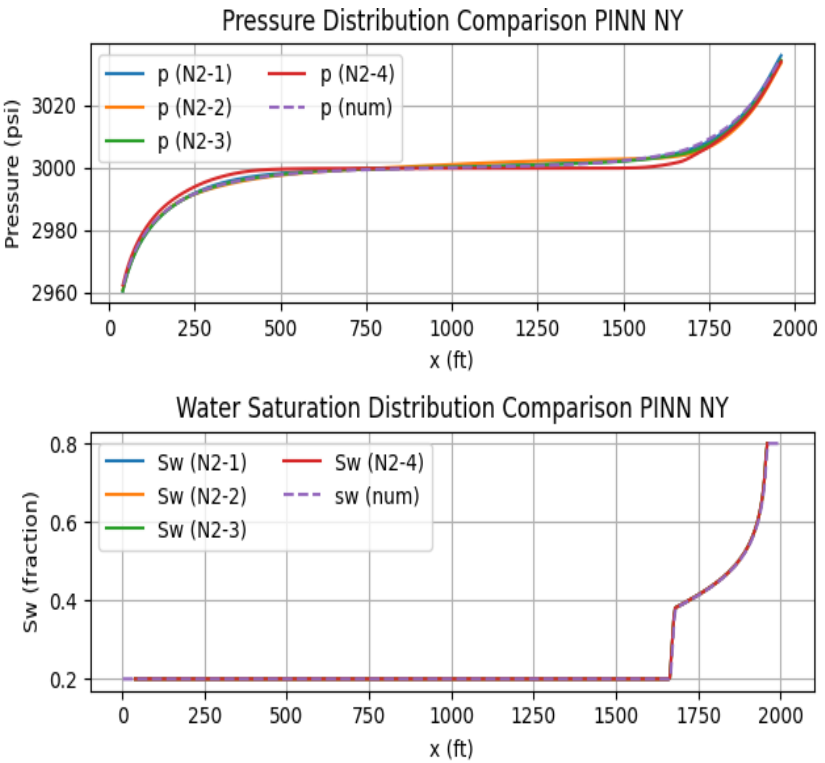


Figure 13. Solution of NY topology at timestep $t+1$

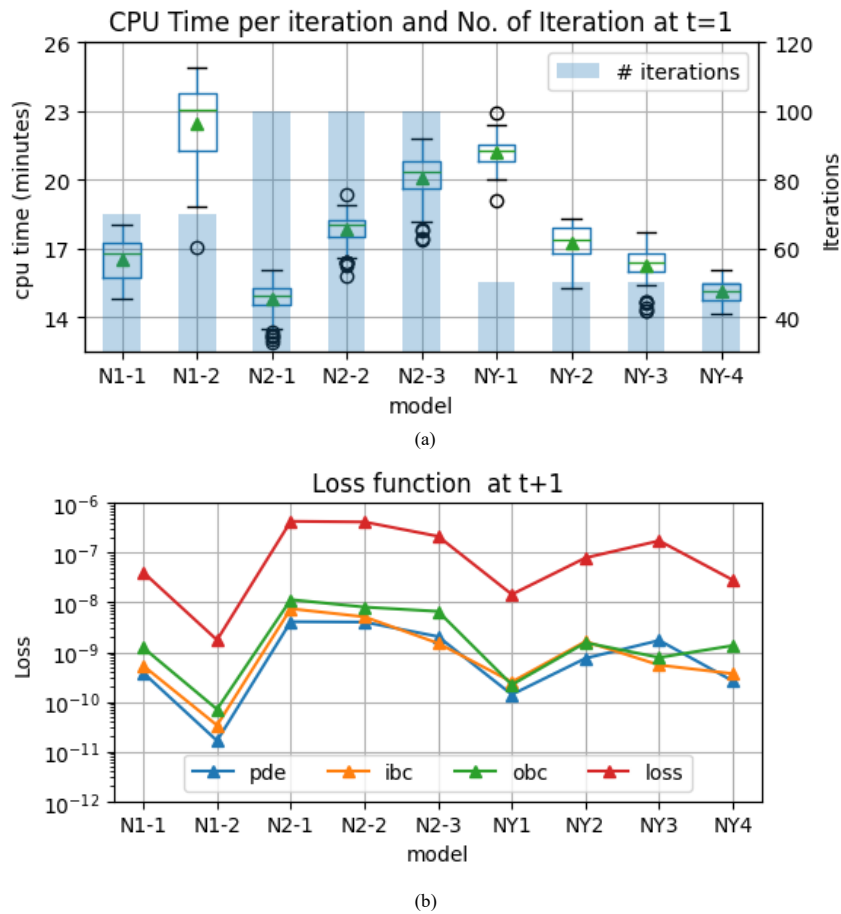


Figure 14. Training at timestep $t+1$: (a). Mean CPU time and number of iterations, (b). Loss fraction value

Subsequent training

The training was extended to solve the fluid flow equations at timestep $t+2$ using three selected models, including N1-2, NY-1, and NY-3. The models had previously solved the flow equations at timestep $t+1$ with high accuracy, thereby providing validated neural networks suitable for use as initial conditions in the subsequent timestep. The use of non-convergent or low-accuracy networks was also avoided to minimize the error propagation across timesteps. The neural network model obtained from

the previous training step was retrained to perform tasks at the subsequent timestep while an identical copy was used as the initial condition for pressure and saturation. As shown in Figure 16, the pressure and saturation distribution curves of N1-2 coincide with the reference solution which shows superior accuracy with the loss function values less than 1.0×10^{-10} . Meanwhile, NY-1 and NY-3 showed higher loss function values at timestep $t+1$ compared to other FNN models and further amplified in the subsequent timestep.

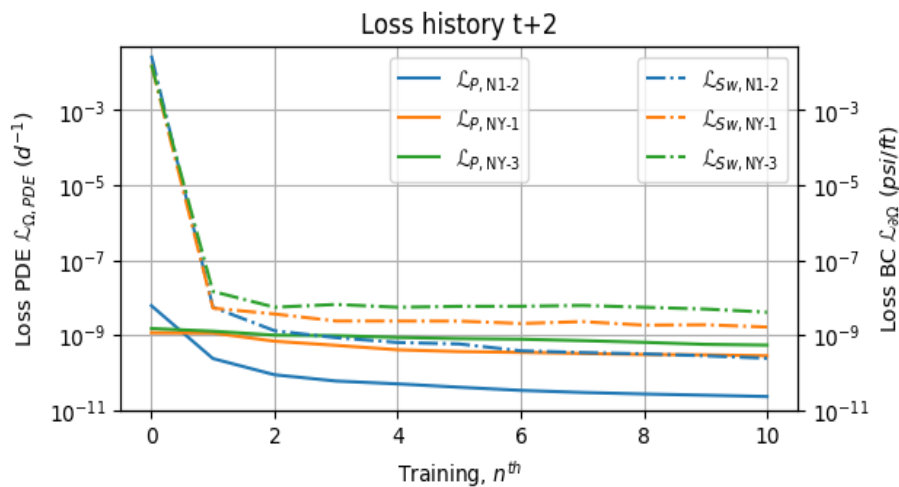


Figure 15. Loss history at timestep $t+2$

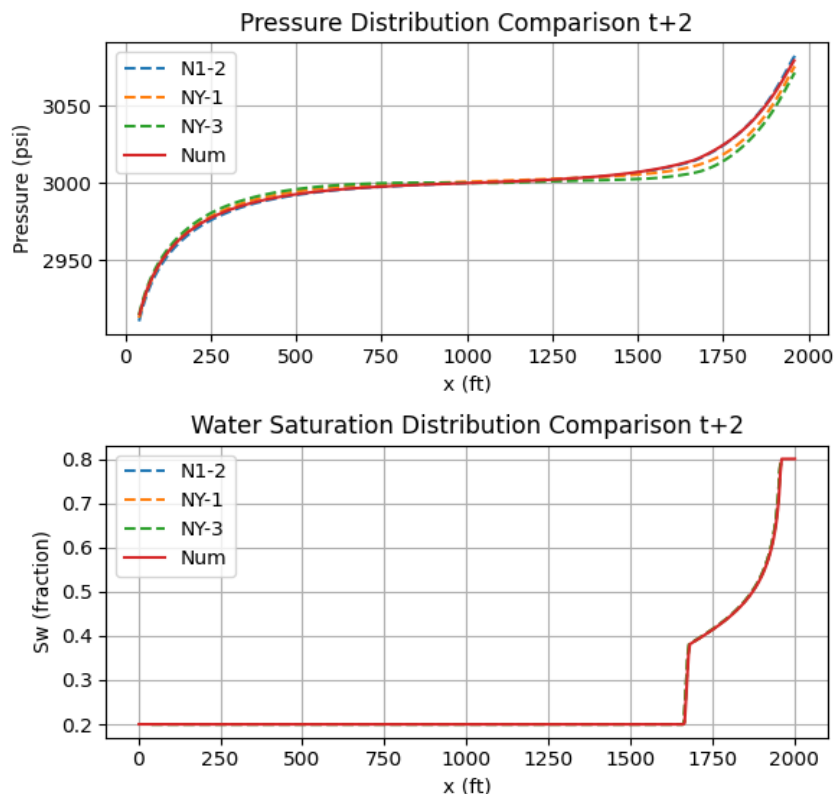


Figure 16. Solutions at timestep $t+2$

CONCLUSION

In conclusion, this study presented a PINN-based fully implicit method for solving two-phase immiscible fluid flow equations. FNN was effectively modeled to capture spatial and reservoir dynamics without requiring observational data or grid discretization. The branched-topology FNN architecture improved stability and convergence by minimizing interference between pressure and saturation distributions. It was also observed that the number of hidden layers critically influenced both convergence speed and accuracy. The proposed method showed strong potential for advancing machine learning-based PDEs solvers but required further improvements particularly in reducing computational costs.

ACKNOWLEDGEMENT

This study was financially supported by the Indonesia Endowment Fund for Education (LPDP), Ministry of Finance, Republic of Indonesia, Grant No. 0001935/TRP/D/BUDI-2019. The first author (Agus Wahyudi) is an employee of the Ministry of Energy and Mineral Resources (KESDM), Indonesia, and gratefully acknowledges the study leave opportunity provided to pursue this study.

GLOSSARY OF TERMS

Symbol	Definition	Unit
K	Permeability	mD
k_r	Relative permeability	
S	Fluid saturation	
S_{or}	Residual oil saturation	
S_{wc}	Connate water saturation	
μ	Viscosity	cp
ϕ	Porosity	
ρ	Density	lb/ft^3
c	Compressibility	$1/psi$
P	Pressure	psi
B	Formation volume factor	
v	Velocity	ft/d
t	Time	day
$\mathcal{R}$	Residual error	
$\mathcal{L}$	Loss function	
θ	Network parameter	
w	Layer weight	
b	Layer bias	

REFERENCES

- Abou-Kassem, Jamal H., M. Rafiqul Islam, & S. M. Farouq Ali. (2020). Petroleum Reservoir Simulation: The Engineering Approach, Second Edition. Petroleum Reservoir Simulation: The Engineering Approach, Second Edition pp. 1–491.
- Alizadeh, Mohammad Reza, & Mobeen Fatemi. (2021). Mechanistic Study of the Effects of Dynamic Fluid/Fluid and Fluid/Rock Interactions during Immiscible Displacement of Oil in Porous Media by Low Salinity Water: Direct Numerical Simulation'. Journal of Molecular Liquids vol. 322, pp. 114544.
- Berrone, Stefano, & Moreno Pintore. (2024). Meshfree Variational-Physics-Informed Neural Networks (MF-VPINN): An Adaptive Training Strategy. Algorithms vol. 17, no. 9.
- Buckley, S. E., & M. C. Leverett. (1942). Mechanism of Fluid Displacement in Sands. Transactions of the AIME vol. 146, no. 01, pp. 107–16.
- Cai, Jianchao, & Steffen Berg. (2025). Recent Advances in Flow Through Porous Media for Energy Exploitation. Energy & Fuels vol. 39, no. 20, pp. 9181–84.
- Corey, A. T., C. H. Rathjens, J. H. Henderson, & M. R. J. Wyllie. (1956). Three-Phase Relative Permeability. Journal of Petroleum Technology vol. 8, no. 11, pp. 63–65.
- Diab, Waleed, Omar Chaabi, Wenjuan Zhang, Muhammad Arif, Shayma Alkobaisi, & Mohammed Al Kobaisi. (2022). Data-Free and Data-Efficient Physics-Informed Neural Network Approaches to Solve the Buckley–Leverett Problem'. Energies vol. 15, no. 21.
- Fraces, Cedric G., Adrien Papaioannou, & Hamdi Tchelepi. (2020). Physics Informed Deep Learning for Transport in Porous Media. Buckley Leverett Problem.
- Fraces, Cedric G., & Hamdi Tchelepi. (2021). Physics Informed Deep Learning for Flow and Transport in Porous Media. SPE Reservoir Simulation Symposium Proceedings.
- Fuks, Olga, & Hamdi A. Tchelepi. (2020). Limitations Of Physics Informed Machine Learning For Nonlinear Two-Phase Transport In Porous Media'.

- Journal of Machine Learning for Modeling and Computing vol. 1, no. 1, pp. 19–37.
- Goda, H. M., & P. Behrenbruch. (2004). Using a Modified Brooks-Corey Model to Study Oil-Water Relative Permeability for Diverse Pore Structures.
- Iskandar, U. P., & Kurihara, M. (2022). Long Short-term Memory (LSTM) Networks for Forecasting Reservoir Performances in Carbon Capture, Utilisation, and Storage (CCUS) Operations. *Scientific Contributions Oil and Gas*, 45(1), 35–51. <https://doi.org/10.29017/SCOG.45.1.943>.
- Jenny, P., S. H. Lee, & H. A. Tchelepi. (2006). Adaptive Fully Implicit Multi-Scale Finite-Volume Method for Multi-Phase Flow and Transport in Heterogeneous Porous Media. *Journal of Computational Physics* vol. 217, no. 2, pp. 627–41.
- Lagaris, Isaac Elias, Aristidis Likas, & Dimitrios I. Fotiadis, 1998, 'Artificial Neural Networks for Solving Ordinary and Partial Differential Equations'. *IEEE Transactions on Neural Networks* vol. 9, no. 5, pp. 987–1000.
- Long, Zichao, Yiping Lu, & Bin Dong. (2019). PDE-Net 2.0: Learning PDEs from Data with a Numeric-Symbolic Hybrid Deep Network. *Journal of Computational Physics* vol. 399, pp. 108925.
- Lu, Lu, Xuhui Meng, Zhiping Mao, & George Em Karniadakis. (2021). DeepXDE: A Deep Learning Library for Solving Differential Equations. <https://doi.org/10.1137/19M1274067> vol. 63, no. 1, pp. 208–28.
- Moncorgé, A., H. A. Tchelepi, & P. Jenny. (2018). Sequential Fully Implicit Formulation for Compositional Simulation Using Natural Variables. *Journal of Computational Physics* vol. 371, pp. 690–711.
- Peaceman, Donald W. (1983). Interpretation of Well-Block Pressures in Numerical Reservoir Simulation With Nonsquare Grid Blocks and Anisotropic Permeability'. *Society of Petroleum Engineers Journal* vol. 23, no. 03, pp. 531–43.
- Raissi, M., P. Perdikaris, & G. E. Karniadakis. (2019). 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.
- Septiano, J., Yasutra, A., & Rahmawati, S. D. (2022). Build of Machine Learning Proxy Model for Prediction of Wax Deposition Rate in Two Phase Flow Water-Oil. *Scientific Contributions Oil and Gas*, 45(1), 34–48. <https://doi.org/10.29017/SCOG.45.1.922>.
- Shukla, Khemraj, Ameya D. Jagtap, & George Em Karniadakis. (2021). Parallel Physics-Informed Neural Networks via Domain Decomposition. *Journal of Computational Physics* vol. 447, pp. 110683.
- Sirignano, Justin, & Konstantinos Spiliopoulos. (2018). DGM: A Deep Learning Algorithm for Solving Partial Differential Equations'. *Journal of Computational Physics* vol. 375, pp. 1339–64.
- Tijink, Paul, & Juan Cottier. (2019). The Description and Quantification of the Truncation Errors Produced by Local-Grid Refinement in Reservoir Simulation'. *SPE Reservoir Evaluation & Engineering* vol. 22, no. 02, pp. 660–72.
- Wardhana, S. G., Pakpahan, H. J., Simarmata, K., Pranowo, W., & Purba, H. (2021). Algoritma komputasi machine learning untuk aplikasi prediksi nilai total organik carbon (TOC). *Lembaran Publikasi Minyak Dan Gas Bumi (LPMGB)*, 55(2), 75–87. <https://doi.org/10.29017/LPMGB.55.2.606>.
- Younis, R. M., H. A. Tchelepi, & K. Aziz. (2010). Adaptively Localized Continuation-Newton Method Nonlinear Solvers That Converge All the Time. *SPE Journal* vol. 15, no. 02, pp. 526–44.
- Zhang, Jingjing, Ulisses Braga-Neto, & Eduardo Gildin. (2024). Physics-Informed Neural Networks for Multiphase Flow in Porous Media Considering Dual Shocks and Interphase Solubility'. *Energy & Fuels* vol. 38, no. 18, pp. 17781–95.