

Quantum-Classical Physics-Informed Neural Networks for Solving Reservoir Seepage Equations

Xiang Rao^{1,2,3,5*}(饶翔), Yina Liu^{4*}(刘怡娜), Yuxuan Shen²(申毓萱)

¹School of Petroleum Engineering, Yangtze University, Wuhan 430100, China

²College of Future Technology, Yangtze University, Wuhan 430100, China

³State Key Laboratory of Low Carbon Catalysis and Carbon Dioxide Utilization (Yangtze University), Wuhan 430100, China

⁴King Abdullah University of Technology, Physical Science and Engineering Division, 23955-6900, Saudi Arabia

⁵Western Research Institute, Yangtze University, Karamay 834000, China

*Corresponding author: Prof. Xiang Rao (raoxiang0103@163.com, raoxiang@yangtzeu.edu.cn), Miss. Yina Liu (yina.liu@kaust.edu.sa)

Abstract: Solving partial differential equations (PDEs) for reservoir seepage is critical for optimizing oil and gas field development and predicting production performance. Traditional numerical methods suffer from mesh-dependent errors and high computational costs, while classical Physics-Informed Neural Networks (PINNs) face bottlenecks in parameter efficiency, high-dimensional expression, and strong nonlinear fitting. To address these limitations, we propose a Discrete Variable (DV)-Circuit Quantum-Classical Physics-Informed Neural Network (QCPINN) and apply it to three typical reservoir seepage models for the first time: the pressure diffusion equation for heterogeneous single-phase flow, the nonlinear Buckley-Leverett (BL) equation for two-phase waterflooding, and the convection-diffusion equation for compositional flow considering adsorption. The QCPINN integrates classical preprocessing/postprocessing networks with a DV quantum core, leveraging quantum superposition and entanglement to enhance high-dimensional feature mapping while embedding physical constraints to ensure solution consistency. We test three quantum circuit topologies (Cascade, Cross-mesh, Alternate) and demonstrate through numerical experiments that QCPINNs achieve high prediction accuracy with fewer parameters than classical PINNs. Specifically, the Alternate topology outperforms others in heterogeneous single-phase flow and two-phase BL equation simulations, while the Cascade topology excels in compositional flow with convection-dispersion-adsorption coupling. Our work verifies the feasibility of QCPINN for reservoir engineering applications, bridging the gap between quantum computing research and industrial practice in oil and gas engineering.

Keywords: Quantum-Classical Physics-Informed Neural Network; Quantum Computing; Reservoir Numerical Simulation; Physics-Informed Neural Network; Flow in Porous Media; Discrete-Variable Quantum Circuit

1 Introduction

Solving partial differential equations (PDEs) is a core problem in the fields of science and engineering. The accurate solution of PDEs for reservoir seepage directly determines the reliability of oil and gas field development plan optimization and production performance

prediction. Traditional numerical methods, such as the finite volume method (Rao et al., 2024), finite element method (Jackson et al., 2015), meshless method (Rao et al., 2022), and others (Wang et al., 2023; Rao et al., 2025a), rely on mesh or point cloud discretization. When dealing with strong reservoir heterogeneity, strong flow nonlinearity, and convection-dominated problems, these methods are prone to mesh-dependent errors or numerical dispersion. Furthermore, the computational cost increases exponentially with the complexity of the system, making it difficult to meet the demand for efficient simulation of complex oil and gas reservoirs.

Physics-Informed Neural Networks (PINNs) embed physical laws into the neural network architecture (Raissi et al., 2019) and achieve PDE solving through a combination of data-driven and physics-constrained paradigms. They can ensure the physical consistency of solutions without requiring large amounts of labeled data, providing a new approach for reservoir forward simulation and inverse problem solving (Almajid & Abu-Al-Saud, 2022; Hanna et al., 2022). Recently, Physics-Informed Kolmogorov-Arnold Networks (PIKAN) have also been developed for reservoir flow simulation (Rao et al., 2025b; Kalesh et al., 2025). For instance, Rao et al. (2025b) applied PIKAN to single-phase seepage simulation of heterogeneous permeability based on a mixed pressure-velocity formulation and found that PIKAN outperformed PINNs in simulation performance. However, classical PINNs and PIKANs face three core limitations when addressing complex reservoir flow scenarios: low parameter efficiency, insufficient high-dimensional expression capability, and restricted fitting of strong nonlinearity. First, to capture the heterogeneity of permeability or the high-dimensional characteristics of governing equations, it is necessary to design deep fully connected networks or increase the number of neurons, leading to a surge in the scale of trainable parameters (usually ranging from thousands to tens of thousands). This not only increases the memory and time costs of training but also easily causes overfitting and unstable convergence. Second, when simulating the steep shock front of convection-dominated equations, the introduction of artificial viscosity terms (Fuks & Tchelepi, 2020) and the gradient propagation characteristics of classical activation functions (e.g., Tanh, ReLU) tend to result in gradient vanishing or local oscillations, making it difficult to balance accuracy and convergence speed. These limitations prevent classical PINNs from achieving both efficiency and accuracy in complex reservoir simulations, becoming a key bottleneck restricting their industrial application.

As a revolutionary paradigm to break through the bottlenecks of classical computing, quantum computing can trace its development back to the 1980s. Feynman (1982) proposed the core idea of quantum system simulation, and Deutsch (1985) confirmed the universal Turing machine property of quantum computers, laying a theoretical foundation for quantum computing. Subsequent landmark achievements, such as the Deutsch-Jozsa algorithm (1992), Shor's large integer factorization algorithm (1994), Grover's database search algorithm (1996), and the Harrow-Hassidim-Lloyd (HHL) linear system solving algorithm (2009), have theoretically verified the exponential acceleration advantages of quantum computing in specific problems. At the experimental level, Cirac and Zoller (1995) proposed the ion trap qubit scheme, Monroe et al. (1995) implemented the first CNOT quantum logic gate, and Chuang et al. (1998) verified the Deutsch-Jozsa algorithm based on nuclear magnetic resonance technology, promoting quantum computing from theory to experiment.

Currently, quantum computing is in the Noisy Intermediate-Scale Quantum (NISQ) era (Bharti et al., 2022), and variational quantum algorithms tailored for NISQ hardware have become a research focus (Cerezo et al., 2021), including the Variational Quantum Eigensolver (VQE) (Peruzzo et al., 2014; Tilly et al., 2022), Variational Quantum Linear Solver (VQLS) (Bravo-Prieto et al., 2023), Variational Quantum Classifier (VQC) (Havlíček et al., 2019; Jäger and Krems., 2023), and Variational Quantum Phase Estimation (VQPE) (Liu et al., 2024). These algorithms generate quantum states through Parameterized Quantum Circuits (PQC) and iteratively optimize parameters using classical optimizers. While avoiding noise accumulation in deep quantum circuits, they achieve approximate solutions to complex problems. In the field of PDE solving, VQLS has been attempted for solving Poisson equations (Ali and Kabel, 2023; Govindugari and Wong, 2024; Ayoub and Baeder, 2025), heat conduction equations (Liu et al., 2022), reservoir seepage equations (Rao., 2024a; Rao., 2024b), and others. By decomposing the coefficient matrix of the linear system (obtained from PDE numerical discretization) into a superposition of Pauli bases, VQLS computes the cost function using quantum state preparation and Hadamard testing to obtain the solution of the linear system. Therefore, the application of VQLS to PDE computation requires prior numerical discretization of the PDE, rather than obtaining the PDE solution purely through quantum computing. Quantum neural networks (QNNs) based on variational quantum algorithms have also been used as pure machine learning methods (Tüysüz et al., 2021; Rao et al., 2024b); for example, Rao et al. (2024b) applied QNNs to predict carbon dioxide (CO_2) sequestration in saline aquifers. However, in general, due to the lack of physical information and the more challenging nonlinear processing compared to classical networks, the prediction performance of pure QNNs is relatively limited.

A natural idea is to develop pure Quantum Physics-Informed Neural Networks (QPINNs). For example, Trahan et al. applied QPINNs to spring-mass systems and Poisson equations, while Berger et al. (2025) applied QPINNs to Poisson, Burger's, and Navier-Stokes equations. Xiao et al. (2024) used QPINNs to solve forward and inverse problems of certain PDEs. Theoretically, however, pure quantum neural networks require more layers than hybrid quantum-classical neural networks, which significantly increases the hardware threshold for physical implementation of the network and makes it more susceptible to noise interference. Moreover, hybrid quantum-classical neural networks can leverage the advantage that classical networks achieve nonlinear complexity at a much lower cost than quantum networks, thereby enhancing the nonlinear fitting capability of the neural network.

Thus, hybrid Quantum-Classical Physics-Informed Neural Networks (QCPINNs) have been further developed. For instance, Sedykh et al. (2024) proposed a QCPINN for simulating laminar fluid flow in 3D Y-shaped mixers, and Dehaghani et al. (2024) applied QCPINNs to solve quantum optimal control problems. Fernandez et al. (2025) integrated a quantum layer into a classical PINN to form a QCPINN and conducted preliminary tests on its computational performance for wave equations. Trahan et al. (2024) and Leong et al. (2025) applied QCPINNs to 1D Burger's equations and 2D inviscid compressible flows, respectively. Farea et al. (2025) applied QCPINNs to Helmholtz equations, wave equations, Klein-Gordon equations, and other PDEs. Through numerical experiments, both Trahan et al. (2024) and Farea et al. (2025) found that compared with classical PINNs, QCPINNs can construct high-

dimensional feature spaces using the properties of quantum superposition and entanglement. This enables PDE solving with far fewer parameters than classical PINNs, significantly improving parameter efficiency. QCPINNs provide a breakthrough solution to the core limitations of classical PINNs: through a hybrid architecture of classical preprocessing, quantum core computing, and classical postprocessing, they achieve deep integration of quantum advantages and physical constraints.

As a newly developed method in the past two years, QCPINNs have so far been applied primarily to fundamental physical PDEs and have not been extended to the field of reservoir engineering. Therefore, this study for the first time applies QCPINN technology to reservoir engineering, systematically simulating three typical seepage models in reservoir seepage: the pressure diffusion equation for single-phase reservoir flow with heterogeneous permeability, the nonlinear Buckley-Leverett (BL) equation for two-phase waterflooding, and the convection-diffusion equation for component concentration considering adsorption in compositional flow. It accurately captures core features such as permeability heterogeneity, flow nonlinearity, and convection dominance, providing an early test case for the future development of reservoir numerical simulators or machine learning surrogate models based on quantum computing. This work promotes the translation of quantum computing from basic physics research to industrial application in oil and gas engineering.

2 Typical Partial Differential Equations in Reservoir Numerical Simulation

This section briefly presents the corresponding typical partial differential equations for three classic reservoir seepage models: single-phase flow, two-phase flow, and compositional flow. Specifically, these equations are the pressure diffusion equation for heterogeneous permeability in single-phase flow, the nonlinear Buckley-Leverett (BL) equation for two-phase flow, and the convection-diffusion equation for component concentration considering adsorption effects.

2.1 Pressure Diffusion Equation with Heterogeneous Permeability in Single-Phase Flow

The core of single-phase flow (e.g., pure oil-phase flow in the early stage of reservoir development) lies in the diffusion process of the pressure field in a medium with heterogeneous permeability. Its governing equation is derived based on Darcy's law and the law of mass conservation. For a two-dimensional (2D) reservoir domain $\mathbf{x} = (x, y)$ and time domain $t \in [0, T]$, the general form of the pressure diffusion equation for heterogeneous single-phase flow is:

$$\frac{\partial}{\partial t}(\phi(\mathbf{x})\rho) = \nabla \cdot \left(\frac{k(\mathbf{x})}{\mu} \rho \nabla p \right) + q_s(\mathbf{x}, t), \quad (1)$$

where $p(\mathbf{x}, t)$ denotes reservoir pressure, the unknown function to be solved; $\phi(\mathbf{x})$ is porosity; $k(\mathbf{x})$ represents the absolute permeability of the rock; ρ is fluid density, assumed constant under weakly compressible conditions; μ denotes fluid viscosity; $q_s(\mathbf{x}, t)$ is the source/sink term (negative for production wells where fluid flows out, positive for injection wells where fluid flows in). Under steady-state and weakly compressible fluid conditions, Eq.

(1) can be further simplified to an elliptic equation for pressure:

$$\nabla \cdot \left(\frac{k(\mathbf{x})}{\mu} \nabla p \right) + \frac{q_s(\mathbf{x})}{\rho} = 0. \quad (2)$$

Boundary conditions include Dirichlet boundaries (constant pressure boundaries, such as edge water or bottom water boundaries) and Neumann boundaries (constant flux boundaries, such as closed faults or impermeable boundaries), which satisfy $p(\mathbf{x}) = p_b$, $(\mathbf{x} \in \Gamma_D, t \in [0, T])$ and $\nabla p(\mathbf{x}) \cdot \mathbf{n} = 0$, $(\mathbf{x} \in \Gamma_N, t \in [0, T])$, respectively. Here, $\Gamma_D \subset \partial\Omega$ and $\Gamma_N \subset \partial\Omega$ are the constant pressure and constant flux boundaries, p_b is the boundary pressure, and \mathbf{n} is the boundary normal vector. The spatial heterogeneity of permeability $k(\mathbf{x})$ is the key factor leading to the nonlinearity of pressure distribution. Theoretically, classical PINNs require increasing the number of neurons or stacking more network layers to enhance nonlinear simulation capabilities. In contrast, the QCPINN can efficiently fit heterogeneous characteristics with fewer parameters by leveraging quantum entanglement.

2.2 Nonlinear Buckley-Leverett (BL) Equation in Two-Phase Flow

The waterflooding process is a typical two-phase flow, and the BL equation describes the nonlinear convection process of saturation through simplified mass conservation relationships, requiring special handling of the steep gradient at the shock front. Under the one-dimensional (1D) flow assumption (along the main streamline direction x), ignoring gravity and capillary forces, the conservative form of the BL equation is:

$$\frac{\partial S_w}{\partial t} + v \frac{\partial f_w(S_w)}{\partial x} = 0, \quad (3)$$

where $S_w(x, t)$ is water saturation; v denotes the total liquid flow velocity; $f_w(S_w)$ is the water cut, representing the ratio of water flow rate to total flow rate, typically a strongly nonlinear function of saturation with the expression:

$$f_w(S_w) = \frac{1}{1 + \frac{\mu_w}{\mu_o} \frac{k_{ro}(S_w)}{k_{rw}(S_w)}}, \quad (4)$$

where $k_{rw}(S_w)$ and $k_{ro}(S_w)$ are the relative permeabilities of the water and oil phases, respectively, usually described by the Corey model:

$$k_{rw}(S_w) = \begin{cases} 0, & S_w \leq S_{wc} \\ \left(\frac{S_w - S_{wc}}{1 - S_{wc} - S_{or}} \right)^{n_w}, & S_{wc} < S_w < 1 - S_{or} \\ 1, & S_w \geq 1 - S_{or} \end{cases}, \quad k_{ro}(S_w) = \begin{cases} 1, & S_w \leq S_{wc} \\ \left(\frac{1 - S_w - S_{or}}{1 - S_{wc} - S_{or}} \right)^{n_o}, & S_{wc} < S_w < 1 - S_{or} \\ 0, & S_w \geq 1 - S_{or} \end{cases}. \quad (5)$$

in which, S_{wc} is the irreducible water saturation, S_{or} is the residual oil saturation, and n_w , n_o are Corey exponents, all determined by core experiments.

The initial condition of the BL equation is generally that the reservoir is saturated with pure

oil before waterflooding, and the water saturation equals the irreducible water saturation: $S_w(x, 0) = S_{wc}$ for $(x \in [0, L])$, where L is the length of the main streamline in the reservoir. The boundary condition generally adopts a constant saturation boundary at the injection end ($x=0$): $S_w(0, t) = 1 - S_{or}$, $(t \in [0, T])$. The core challenge of the BL equation is the shock front induced by nonlinear convection: when the fractional flow function $f_w(S_w)$ is convex, the saturation distribution forms a steep discontinuous front whose position advances with time. When fitting such fronts, classical PINNs tend to suffer from “numerical diffusion” due to the local linearity of Tanh or ReLU activation functions, resulting in an excessively wide predicted saturation transition zone at the front. This issue can only be mitigated by increasing the sampling density, leading to a sharp surge in computational cost. In theory, The DV-Circuit QCPINN enhances local fitting capabilities for steep gradients through nonlinear combinations of controllable rotation gates, thereby reducing diffusion errors.

2.3 Convection-Diffusion Equation for Concentration in Compositional Flow

Chemical flooding (e.g., polymer flooding) or CO₂ sequestration processes in reservoir development involve convection, diffusion, and adsorption of multicomponent substances on rock surfaces. Taking a 2D reservoir domain $\Omega \subset \mathbb{R}^2$ as an example, the convection-diffusion equation for the concentration of a single component (e.g., polymer) is:

$$\phi \frac{\partial c}{\partial t} + \rho_b \frac{\partial q}{\partial t} = \nabla \cdot (D \nabla c) - \nabla \cdot (\mathbf{v} c), \quad (6)$$

where $c = c(\mathbf{x}, t)$ denotes the component concentration; t represents time; ϕ is the porosity; ρ_b stands for the rock density; q denotes the mass of solute adsorbed per unit mass of rock. It satisfies the linear adsorption law (Henry's law) with the concentration c , i.e., $q = K_d c$, where K_d is the distribution coefficient reflecting the rock's adsorption capacity for the component; D represents the diffusion coefficient of the component; $\mathbf{v} = (v_x, v_y)$ denotes the seepage velocity distribution.

The initial condition of the multicomponent convection-diffusion equation is that no injected components are present in the reservoir at the early development stage, with a concentration of zero: $c(\mathbf{x}, 0) = 0, (\mathbf{x} \in \Omega)$. Boundary conditions generally include: a constant concentration injection at the injection boundary $\mathbf{x} \in \Gamma_{inj}$ denoted by $c(\mathbf{x}, t) = c_{inj}$, $(t \in [0, T])$; a convection-dominated free outflow at the production boundary $\mathbf{x} \in \Gamma_{out}$ denoted by $\nabla c(\mathbf{x}, t) \cdot \mathbf{n} = 0$, $(t \in [0, T])$; and a closed boundary $\mathbf{x} \in \Gamma_{close}$ denoted by $\nabla c(\mathbf{x}, t) \cdot \mathbf{n} = 0$ and $c(\mathbf{x}, t) \mathbf{u}(\mathbf{x}, t) \cdot \mathbf{n} = 0$, $(t \in [0, T])$. In reservoir seepage, the convection velocity is much higher than the diffusion velocity, resulting in convection-dominated equations, sharp concentration fronts, and a high tendency for numerical oscillations. Classical PINNs require introducing artificial viscosity or other methods to suppress oscillations, which inevitably compromises front accuracy. In contrast, the high-dimensional feature mapping capability of the DV-Circuit QCPINN enables accurate fitting of convection-diffusion coupling

without artificial viscosity.

3 Discrete Variable (DV)-Circuit QCPINN Method

The core idea of the DV-Circuit QCPINN (Farea et al., 2025) is to leverage classical networks for input dimension adaptation and feature preprocessing, utilize the superposition/entanglement properties of DV quantum circuits to achieve high-dimensional feature mapping, then employ classical networks again to decode quantum outputs and calculate physical losses, and ultimately optimize both classical and quantum parameters through backpropagation.

Targeting the characteristics of reservoir PDEs, this study designs a tailored DV-Circuit QCPINN framework to ensure stability and accuracy under scenarios of heterogeneity, strong nonlinearity, and convection dominance. As illustrated in Fig. 1, this framework comprises four components: a classical preprocessor, a DV quantum core, a classical postprocessor, and a physical loss calculation unit. Specifically, the preprocessor maps reservoir inputs (spatiotemporal coordinate parameters of pressure, saturation, or concentration) to quantum input dimensions; the quantum core generates quantum states via parameterized DV circuits and performs measurements; the postprocessor decodes quantum measurement results into physical variables (pressure, saturation, or concentration); and the loss unit computes PDE residual loss as well as boundary/initial condition losses, with all parameters optimized through backpropagation.

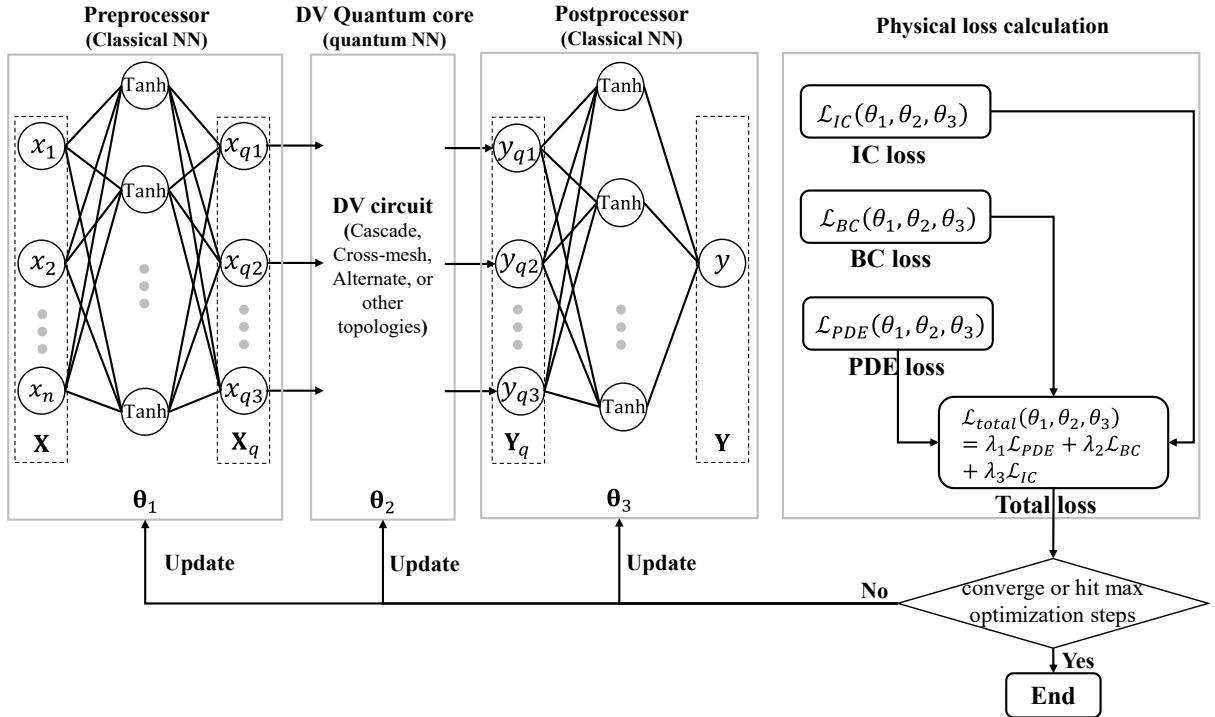


Fig. 1 Schematic diagram of the QCPINN workflow proposed in this study.

Next, we elaborate on the technical details of each component.

The core functions of the classical preprocessor are input dimension adaptation and feature normalization. It converts inputs of reservoir seepage PDEs (e.g., spatiotemporal coordinates

x, y, t for 2D pressure equations) into low-dimensional quantum inputs acceptable to quantum circuits, while reducing sensitivity to parameter initialization. Specifically, it is designed as a fully connected network with a single hidden layer (i.e., one input layer, one hidden layer, and one output layer): the input is the feature vector $\mathbf{X} \in \mathbb{R}^{d_m}$ of the reservoir PDE (e.g., for 2D pressure equations, including x, y), and the number of neurons in the hidden layer is d_h ($d_h = 50$ in the numerical examples of this paper). The linear transformation from the input layer to this hidden layer is given by $\mathbf{h}_1 = W_1 \mathbf{X} + b_1$, where $W_1 \in \mathbb{R}^{d_h \times d_m}$ is the weight matrix, $b_1 \in \mathbb{R}^{d_h}$ is the bias vector, and Xavier initialization is adopted (to avoid gradient vanishing). A Tanh activation function is introduced in the hidden layer to enhance the nonlinear expression capability of the network. Subsequently, the output layer data $\mathbf{X}_q \in \mathbb{R}^{d_q}$ is obtained through full connection between the hidden layer and the output layer (where d_q is the number of qubits used in the quantum circuit), serving as the input to the subsequent quantum circuit, i.e., $\mathbf{X}_q = W_2 \text{Tanh}(\mathbf{h}_1) + b_2$. Thus, Eq. (7) presents the computation result of the preprocessor for the feature vector \mathbf{X} of the reservoir PDE. For convenience of description, the set of parameters W_1, W_2, b_1, b_2 involved in the classical preprocessor is denoted as $\boldsymbol{\theta}_1$.

$$\mathbf{X}_q = W_2 \text{Tanh}(W_1 \mathbf{X} + b_1) + b_2 = f_1(\mathbf{X}, \boldsymbol{\theta}_1), \quad (7)$$

where f_1 represents the mapping from the input \mathbf{X} of the preprocessing network and the preprocessing network parameters $\boldsymbol{\theta}_1$ to the input \mathbf{X}_q of the quantum network.

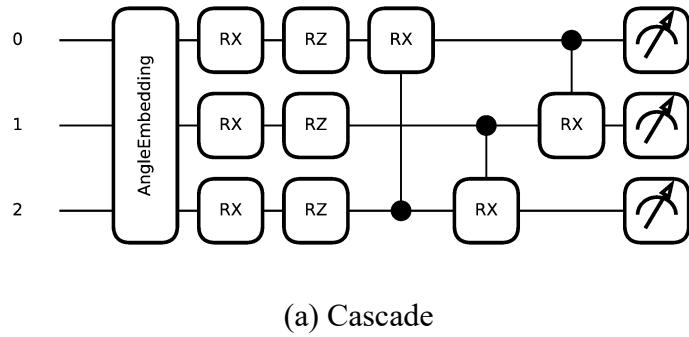
The DV quantum core is the computational engine of QCPINN, responsible for encoding classical preprocessed features into quantum states, achieving high-dimensional feature mapping via Parameterized Quantum Circuits (PQL), and ultimately obtaining classical outputs through measurements. This paper adopts Angle Embedding to map classical features to quantum states, in which each classical feature independently controls the rotation angle of one qubit, ensuring stable gradient propagation. For quantum input $\mathbf{X}_q = [x_{q1}, x_{q2}, \dots, x_{qd_q}]$, each element x_{qi} is treated as a rotation angle θ_{qi} , and then an RX rotation gate is sequentially applied to each qubit i to implement feature encoding, i.e., $U_{\text{embed}}(i) = \text{RX}(\theta_{qi})$. Through local rotation operations, Angle Embedding directly encodes classical features into the phase of qubits, avoiding coupling errors caused by global normalization.

The topology of the quantum circuit determines the entanglement pattern between qubits and directly affects the feature mapping capability. In each subsequent numerical example, three topologies (Cascade, Cross-mesh, and Alternate) are tested. Fig. 2(a) shows a single-layer Cascade topology for three qubits: it adopts a ring connection (qubit 0-2-1-0), where each qubit is only entangled with adjacent qubits, belonging to a hardware-efficient topology. Each layer of the circuit structure includes single-qubit rotation gates and two-qubit Controlled Rotation gates (CRX). The single-qubit layer applies $\text{RX}(\alpha_i) - \text{RZ}(\beta_i) - \text{RX}(\gamma_i)$ to each qubit, and the

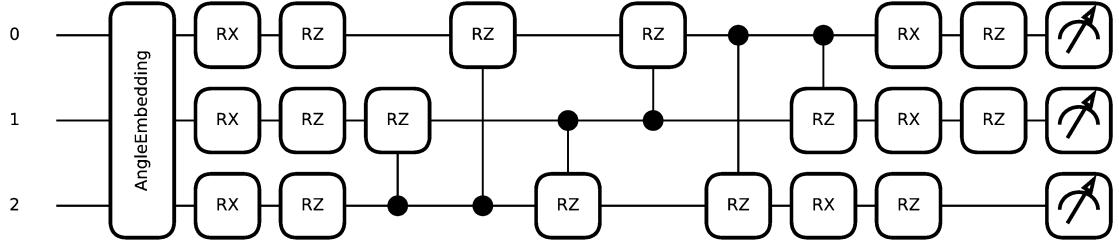
entanglement layer applies $\text{CRX}(\delta_{i,i+1})$ gates along the ring connection. For $d_q = 3$ qubits and $L=1$ layer (a constraint of NISQ-era hardware), the circuit depth is $(d_q + 2)L = 5$, and the number of parameters is $3 \times d_q \times L = 9$, including $2d_q \times L = 6$ single-qubit parameters and $d_q \times L = 3$ entanglement parameters, far fewer than the thousands of parameters in classical PINNs.

Fig. 2(b) depicts a single-layer Cross-mesh topology for three qubits: it features a fully connected structure (each qubit is entangled with all other qubits), categorized as a highly expressive topology suitable for multi-variable coupling scenarios. The entanglement layer of the circuit structure applies $\text{CRZ}(\delta_{i,j})$ gates to all qubit pairs (i,j) ($i \neq j$). For $d_q = 3$ and $L=1$, the circuit depth is $(d_q^2 - d_q + 4)L = 10$ layers, and the number of parameters is $(d_q^2 + 3d_q)L = 18$, including $4d_q \times L = 12$ single-qubit parameters and $d_q(d_q - 1)L = 6$ entanglement parameters, enabling efficient characterization of the coupling correlation between concentration and velocity fields.

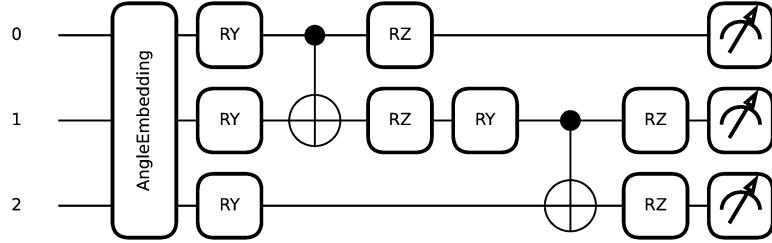
Fig. 2(c) illustrates a single-layer Alternate topology for three qubits: it employs a nearest-neighbor connection (entanglement only between adjacent qubits), serving as a lightweight and efficient topology that balances circuit complexity control and basic local correlation characterization. The circuit structure alternately deploys parameterized single-qubit rotation layers and nearest-neighbor entanglement layers: the single-qubit layer sequentially applies optimizable rotation gates such as RY and RZ to each qubit, while the entanglement layer performs entangling operations like CNOT on adjacent qubit pairs (e.g., 1-2, 2-3). For $d_q = 3$ qubits and $L=1$ layer, according to topological feature calculations, the circuit depth is $6L=6$ layers, and the number of parameters is $4(d_q - 1)L = 8$ (all single-qubit parameters), with $d_q - 1 = 2$ two-qubit CNOT gates. This design not only captures local dependencies in physical systems through alternating rotation and entanglement operations but also reduces resource consumption and noise impacts of quantum hardware via its concise connection scheme.



(a) Cascade



(b) Cross-mesh



(c) Alternate

Fig. 2 Three Quantum Circuit Topologies used in this study.

For ease of description, the set of all parameters involved in the quantum circuits shown in Fig. 2 is denoted as $\boldsymbol{\theta}_2$. Quantum measurement is a key step in mapping quantum states to classical outputs. This paper adopts Pauli-Z expectation measurement: for each qubit i , the expected value $\langle \hat{Z}_i \rangle$ under the Pauli-Z basis is measured, yielding a classical vector $\mathbf{Y}_q \in \mathbb{R}^{d_q}$, where $\langle \hat{Z}_i \rangle = \text{Tr}(\rho(\boldsymbol{\theta}_2) \hat{Z}_i)$, and $\rho(\boldsymbol{\theta}_2)$ is the density matrix output by the quantum circuit, and \hat{Z} is the Pauli-Z operator. The measurement results range from [-1, 1], which need to be mapped to the actual range of physical variables via the postprocessor. Thus, the quantum circuit can be abstractly expressed as Eq. (8).

$$\mathbf{Y}_q = f_2(\mathbf{X}_q, \boldsymbol{\theta}_2), \quad (8)$$

where f_2 represents the mapping from the input \mathbf{X}_q of the quantum network and the network parameters $\boldsymbol{\theta}_2$ to the output \mathbf{Y}_q of the quantum network.

The classical postprocessor is responsible for quantum output decoding and physical variable mapping, converting the low-dimensional vector from quantum measurements into the target physical variables of reservoir seepage PDEs (pressure p , saturation S_w , or component concentration c). It is designed as a fully connected network with a single hidden layer, symmetric to the preprocessor: the input of the first layer (quantum output decoding) is the quantum measurement vector $\mathbf{Y}_q \in \mathbb{R}^{d_q}$, which is linearly transformed to the hidden layer vector $\mathbf{h}_3 \in \mathbb{R}^{d_h}$, i.e., $\mathbf{h}_3 = W_3 \mathbf{Y}_q + b_3$, where $W_3 \in \mathbb{R}^{d_h \times d_q}$ and $b_3 \in \mathbb{R}^{d_h}$ are trainable parameters initialized via Xavier. The Tanh activation function is introduced in this hidden

layer, followed by a linear transformation through full connection to obtain the output layer data \mathbf{Y} . Thus, Eq. (9) summarizes the operation process of the classical postprocessor. For convenience, the set of parameters W_3 , W_4 , b_3 , b_4 involved in the classical postprocessor is denoted as $\boldsymbol{\theta}_3$.

$$\mathbf{Y} = W_4 \text{Tanh}(W_3 \mathbf{Y}_q + b_3) + b_4 = f_3(\mathbf{Y}_q, \boldsymbol{\theta}_3), \quad (9)$$

where $W_4 \in \mathbb{R}^{1 \times d_h}$ and $b_3 \in \mathbb{R}^1$. f_3 represents the mapping from the output \mathbf{Y}_q of the quantum network and the postprocessing network parameters $\boldsymbol{\theta}_3$ to the output \mathbf{Y} of the postprocessing network.

Combining Eqs. (7), (8), and (9), the mapping of input $\mathbf{X} \in \mathbb{R}^{d_{in}}$ to output $\mathbf{Y} \in \mathbb{R}^1$ for the hybrid quantum-classical network in Fig. 1 can be denoted as $\mathbf{Y} = f(\mathbf{X}, \boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3)$.

Physical loss is the core guarantee of QCPINN’s physical consistency requiring simultaneous minimization of PDE residual loss (to ensure the solution satisfies the governing equation) and boundary/initial condition losses (to ensure the solution satisfies constraints). The total loss function is:

$$\mathcal{L}_{\text{total}}(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3) = \lambda_1 \mathcal{L}_{PDE}(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3) + \lambda_2 \mathcal{L}_{BC}(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3) + \lambda_3 \mathcal{L}_{IC}(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3), \quad (10)$$

where λ_1 , λ_2 , and λ_3 are loss weights (we set $\lambda_1=1$, $\lambda_2=1$, $\lambda_3=1$ in this paper). The PDE residual loss is calculated using Mean Squared Error (MSE) for the residuals of the PDE governing equation based on automatic differentiation at sampling points. The boundary condition loss computes the MSE between predicted values and boundary conditions at boundary sampling points, covering both Dirichlet and Neumann boundaries. The initial condition loss calculates the MSE between predicted values and initial conditions at sampling points.

The parameters of the DV-circuit QCPINN include classical parameters $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_3$ as well as quantum parameters $\boldsymbol{\theta}_2$. The optimization process updates both types of parameters simultaneously. The Adam optimizer is adopted, combined with a ReduceLROnPlateau scheduler to dynamically decay the learning rate during loss plateaus. This reduces sensitivity to quantum parameter initialization and helps escape local optima. Additionally, gradient clipping is applied to limit the gradient norm, preventing “gradient explosion” of parameter gradients.

The DV-circuit QCPINN code in this paper is developed based on the open-source code by Farea et al. (2025) and the PennyLane platform. Key modifications and adaptations include reservoir PDE-specific adjustments and DV quantum circuit topology adaptations. Similarly, PennyLane is configured with shots=None to enable analytical gradient calculation, avoiding gradient fluctuations caused by sampling noise.

4 Numerical experiments

4.1 Example 1

This example focuses on the heterogeneous-permeability single-phase flow problem

described in Section 2.1. Consider a 2D rectangular reservoir $\Omega = [0, 100]m \times [0, 100]m$. By setting $k(\mathbf{x}) = K_0 + y$, $K_0 = 10^{-2} \text{ m}^2 / (\text{MPa} \cdot \text{s})$, and $q_s(\mathbf{x}) = 0$ in Eq. (2), the seepage governing equation for this example can be rewritten as Eq. (11). As shown in Fig. 3, the upper and lower boundaries of the reservoir in this example are constant-pressure boundaries, with the upper boundary pressure being 5 MPa and the lower boundary pressure 10 MPa. The left and right boundaries are set as impermeable closed boundaries.

$$\nabla \cdot ((K_0 + y) \nabla p(x, y)) = 0, \quad (x, y) \in \Omega, \quad (11)$$

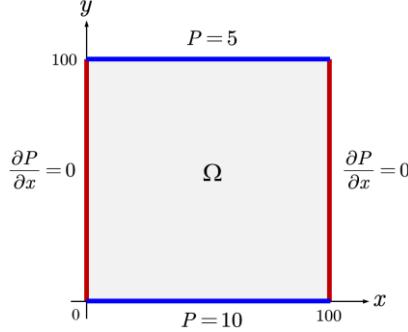


Fig. 3 Schematic of the reservoir domain and boundary conditions for Example 1.

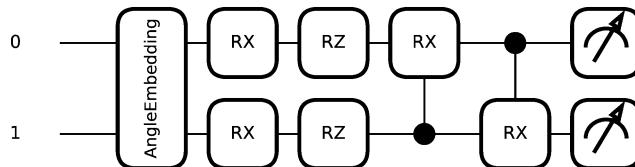
The QCPINN model in this example takes spatial coordinates (x, y) as input and outputs the pressure value p . Table 1 lists the detailed parameters and training configurations of the QCPINN model (classical network and DV quantum circuit) used in this example. To systematically evaluate the impact of different quantum circuit topologies on QCPINN performance, this study adopted three quantum circuit topologies: Cascade, Cross-mesh, and Alternate, as shown in Fig. 4. During training, the Adam optimizer was used with an initial learning rate of 0.005. The ReduceLROnPlateau scheduler was introduced to reduce the learning rate when the loss showed no improvement for multiple consecutive epochs. The patience value was set to 1000 epochs, the learning rate decay factor was 0.9 (a 10% reduction each time), and the minimum learning rate was 10^{-6} (to prevent training stagnation due to excessively low learning rates). Meanwhile, the maximum parameter gradient norm was set to 1.0, and gradient clipping was applied to limit the gradient norm. It should be noted that all numerical experiments in this study were performed on a CPU platform with a system configuration of two Intel Xeon E5-2680 v4 processors (28 physical cores in total, 2.40 GHz main frequency) and 128 GB of memory. Although quantum simulation has inherent overhead on classical hardware, given the small scale of the quantum circuit in this example (2 qubits) and a batch size of 64, this CPU environment can still ensure stable and efficient training.

Fig. 3 shows the loss function convergence performance of the three quantum circuit topologies (Cascade, Cross-mesh, and Alternate) during training. Overall, all three topologies rapidly decreased from an initial loss level of approximately 10^2 to a stable magnitude of 10^{-3} , indicating that the model has good trainability and convergence stability after embedding physical constraints. From the comparison results, the Cross-mesh topology decreased the fastest during training and achieved a slightly lower final loss than the other two structures. The Cascade and Alternate topologies had relatively slower convergence speeds but reached

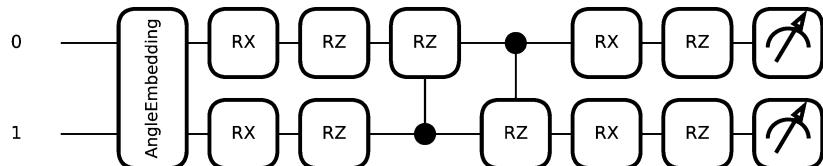
similar stable loss levels in the later stage.

Fig. 4 compares the exact solution with the QCPINN prediction results using the three quantum circuit topologies. As shown in Fig. 6, all QCPINN models successfully simulated the nonlinear pressure distribution in the y -direction, and all were highly consistent with the reference solution in terms of overall shape, with almost no visible differences. Fig. 7 presents the absolute error distributions of the pressure distribution for the three topologies, and their quantitative error indicators are listed in Table 2. We can see that the Cascade topology had a mean absolute error of 0.002860, which was at an intermediate level among the three, but the maximum absolute error was the highest (0.015921), with errors mainly concentrated in the high-gradient region near the lower boundary. The Cross-mesh topology had a slightly higher mean absolute error (0.003892) and a maximum absolute error of 0.008218, with mild error aggregation in the internal region and some boundaries. In contrast, the Alternate topology (Fig. 5c) had the best overall performance, with the lowest mean absolute error (0.001697) and the lowest maximum relative error (0.11%) among the three, with only slight error accumulation in the corner regions.

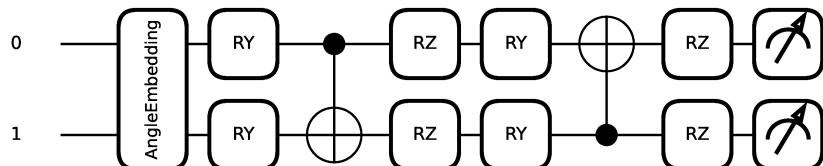
In summary, for the seepage problem in heterogeneous reservoir described by the elliptic equation in this example, the three quantum circuit topologies (Cascade, Cross-mesh, and Alternate) have small differences in loss function convergence speed and computational accuracy, and all can be used as effective QCPINN architectures. Among them, the Alternate topology performs better in terms of computational accuracy compared to the Cascade and Cross-mesh topologies.



(a) Cascade



(b) Cross-mesh



(c) Alternate

Fig. 4 Three quantum circuit topologies used in Example 1.

Table 1 Parameters of the neural networks and optimizers in Example 1.

Configuration Category	Parameter Name	Setting Value
Neural Network	Hidden Layers	1 layer with 50 neurons
	Activation Function	Tanh
DV Quantum circuit	Circuit Topology	Cascade
		Cross-mesh
		Alternate
DV Quantum circuit	Number of Qubits	2
	Number of Quantum Layers	1
	Encoding	angle
Training Hyperparameters	Optimizer	Adam
	Initial Learning Rate	0.005
	Batch Size	64
	Maximum Training Epochs	20000

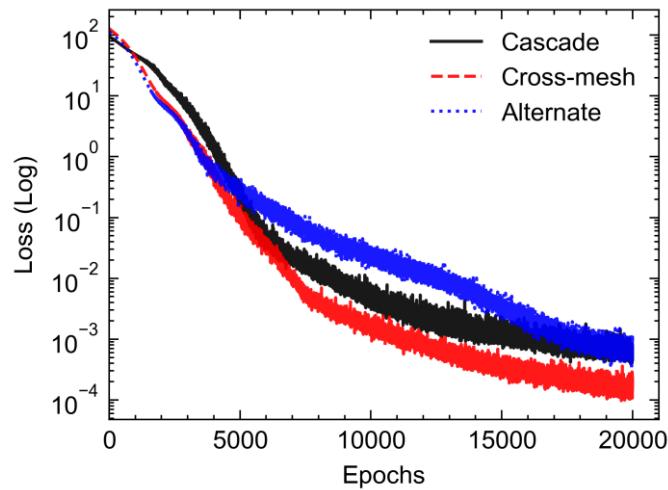


Fig. 5 Comparison of training loss convergence for three quantum circuit topologies in Example 1.

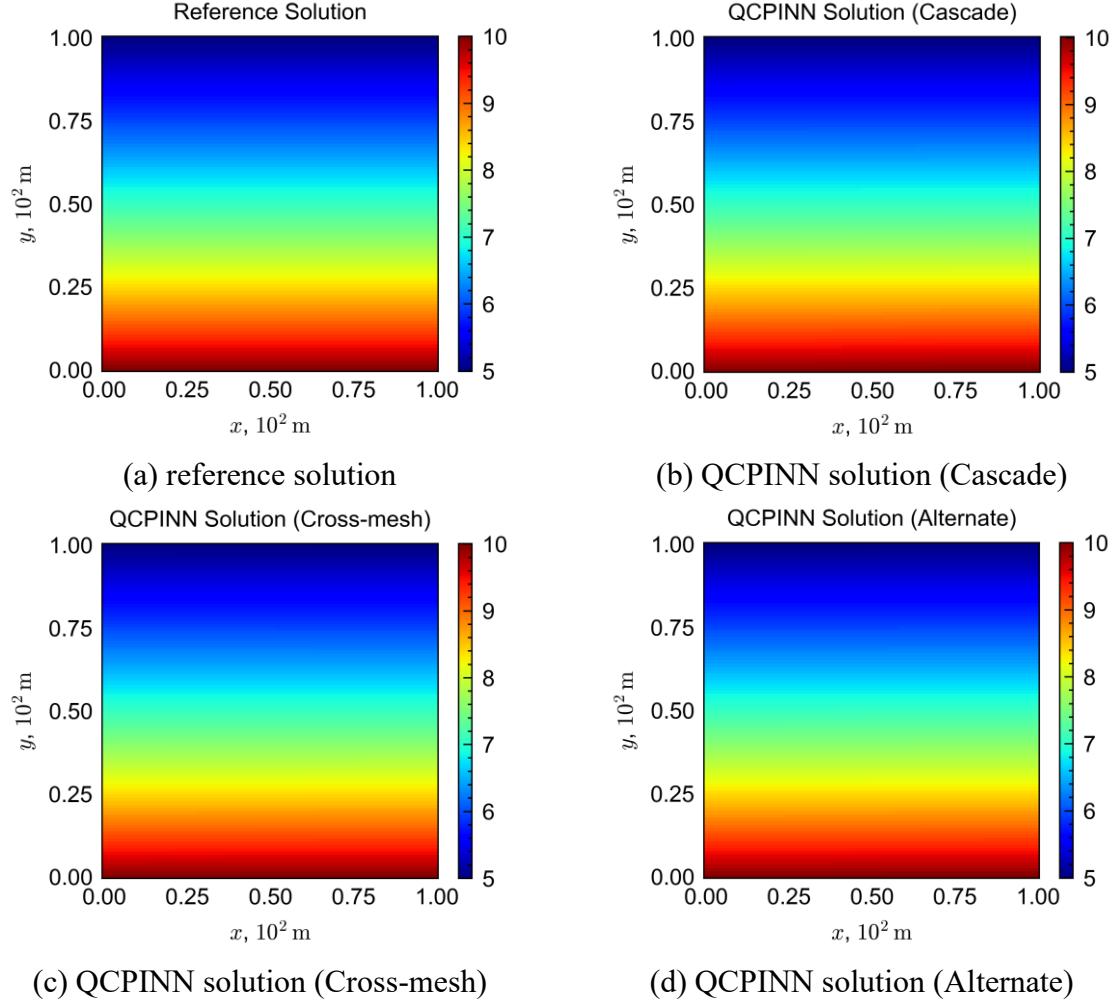


Fig. 6 Comparison of pressure distributions: (a) reference solution, and QCPINN predictions using (b) Cascade, (c) Cross-mesh, and (d) Layered topologies.

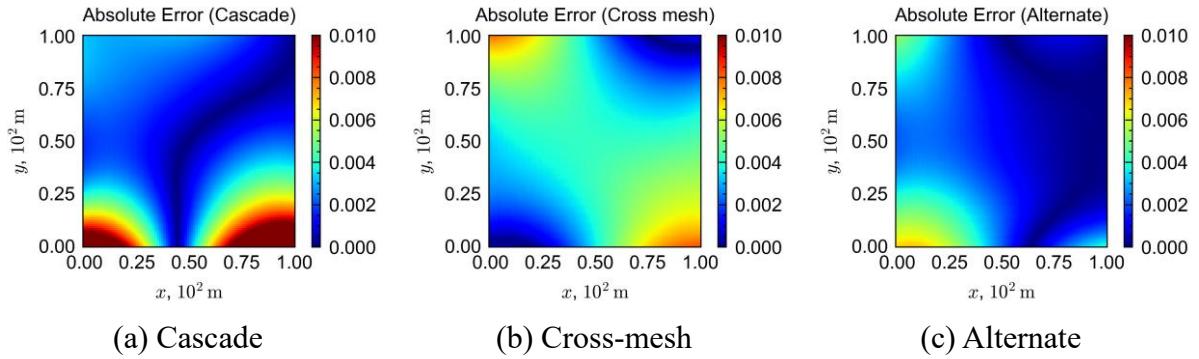


Fig. 7 Absolute error distributions of the pressure profiles predicted by QCPINN using the (a) Cascade, (b) Cross-mesh, and (c) Layered topologies.

Table 2 Comparison of errors for the pressure profiles predicted by QCPINN using three quantum circuit topologies in Example 1.

Topological Structure	Mean Absolute Error	Max Absolute Error	Mean Relative Error	Max Relative Error
-----------------------	---------------------	--------------------	---------------------	--------------------

Cascade	0.002860	0.015921	0.04%	0.16%
Cross mesh	0.003892	0.008218	0.06%	0.16%
Alternate	0.001697	0.006967	0.02%	0.11%

4.2 Example 2

This example focuses on the two-phase flow problem described in Section 2.2. By setting $S_{wc} = 0$, $S_{or} = 0$, $n_o = n_w = 1$, and a constant total oil-water two-phase velocity $v = 1$ in the x -direction (the units of x , t , and v are adjusted to be compatible, so they are not explicitly specified in this example), Eq. (3) can be concretized as the nonlinear Buckley-Leverett (BL) equation in Eq. (12).

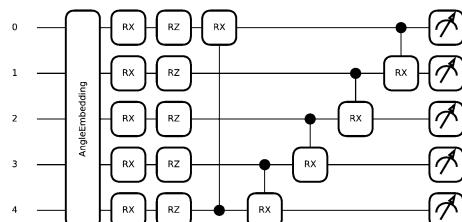
$$\frac{\partial S_w}{\partial t} + v \frac{\partial f_w}{\partial x} = 0, \quad 0 \leq x \leq 1, \quad 0 \leq t \leq 1 \quad (12)$$

where $f_w(S_w) = S_w / \left(S_w + \frac{1-S_w}{M} \right)$, $M = 2$ is the viscosity ratio of oil to water.

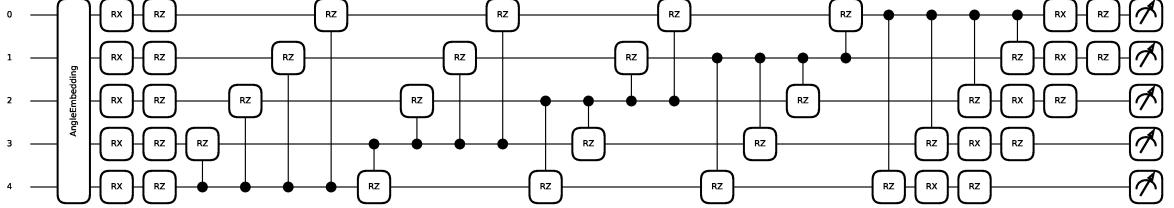
The initial water saturation in the reservoir calculation domain is 0. The left boundary is set as a constant water saturation boundary of 1 to simulate continuous water injection displacement conditions. The initial and boundary conditions of Eq. (12) are expressed as:

$$S_w(0, t) = 1, \quad 0 \leq t \leq 1, \quad S_w(x, 0) = 0, \quad 0 \leq x \leq 1. \quad (13)$$

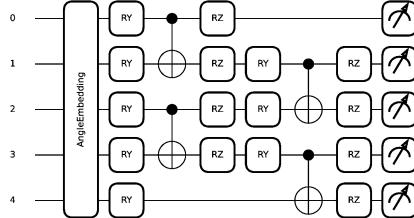
The QCPINN model in this example takes spatial coordinates and time variables (x, t) as input and outputs the water saturation S_w . To systematically evaluate the impact of different quantum circuit topologies on QCPINN performance, this example continues the design of Example 1 and uses the three topologies (Cascade, Cross-mesh, and Alternate) for calculations. Since this problem is a transient problem and the nonlinearity of the water cut function will theoretically increase the model learning difficulty, the number of qubits is increased to 5 while the number of quantum layers remains 1. The corresponding three quantum circuit topologies are shown in Fig. 6. The other network parameters, optimizer parameters, and operating hardware environment of the model are consistent with those in Example 1.



(a) Cascade



(b) Cross-mesh



(c) Alternate

Fig.6 Three quantum circuit topologies used in Example 2.

Fig. 7 shows the loss function convergence of the three quantum circuit topologies during training. Overall, all three topologies successfully reduced the loss value from an initial magnitude of 10^0 to around 10^{-2} , but continued to oscillate near 10^{-2} in the later stage, reflecting the convergence challenges posed by the nonlinearity and transient characteristics of the problem. Specifically, the Cross-mesh topology exhibits a rapid decline in the loss function during the initial training phase but, like the Cascade topology, shows significant fluctuations in the loss function in the later training stages. In contrast, the loss function curve of the Alternate topology displays a notably smaller oscillation amplitude, indicating superior convergence in this case. Eventually, the loss levels of the three models tend to converge to a comparable magnitude.

Fig. 8 compares the prediction results of the three QCPINN topologies at different time instants ($t=0.1, 0.3, 0.5, 0.7, 0.9$) with the reference solution from the upwind finite difference method based on an ultra-dense grid. The results demonstrate that all three topologies can accurately capture the steep water-flooding front and its migration behavior, with the overall morphology being basically consistent with the reference solution. Specifically, the Cascade topology shows slight deviations in the front and the water-unswept regions, but the overall curve trend remains consistent with the reference solution. The Cross-mesh topology is highly consistent with the reference solution in all regions at each time step except for a noticeable deviation from the reference solution at the front. The Alternate topology only exhibits minor deviations in the water-unswept regions, while fitting most closely with the reference solution in other regions, almost overlapping completely.

Fig. 9 presents the relative L2 errors between the saturation predictions of three quantum topological structures and the reference solution over the time interval of 0.1-0.9. As a quantitative evaluation metric, the L2 error is defined as follows:

$$L_2(t_k) = \sqrt{\frac{1}{N} \sum_{i=1}^N (S_{pred}(x_i, t_k) - S_{ref}(x_i, t_k))^2} \quad (14)$$

where N denotes the number of spatial sampling points, t_k represents a specific time point, S_{pred} and S_{ref} stand for the water saturation predicted by QCPINN and the reference solution, respectively.

We can see from Fig. 9 that the errors of all topological structures remain at a low magnitude, indicating the model exhibits excellent predictive accuracy. Specifically, in the early stage $t=0.1$, the Cross-mesh topology achieves the lowest error (approximately 0.008), followed by the Cascade topology, while the Alternate topology yields the highest error. With the progression of time, the accuracy performance of each topology undergoes significant changes. The error of the Cross-mesh topology gradually increases, whereas the error of the Alternate topology decreases markedly and remains consistently at a low level (around 0.006) for all time steps from $t=0.4$ to $t=0.9$, demonstrating significantly superior accuracy compared to the Cascade and Cross-mesh topologies. The Cascade topology exhibits relatively high errors across all time steps and fails to demonstrate obvious advantages at any stage.

Overall, for the 1D nonlinear Buckley-Leverett (BL) equation describing waterflooding, the QCPINN models constructed with the Cascade, Cross-mesh, and Alternate topologies all demonstrate excellent capabilities in capturing physical characteristics and high prediction accuracy. Among them, the Alternate topology stands out in terms of training convergence and computational accuracy, showing relatively superior performance.

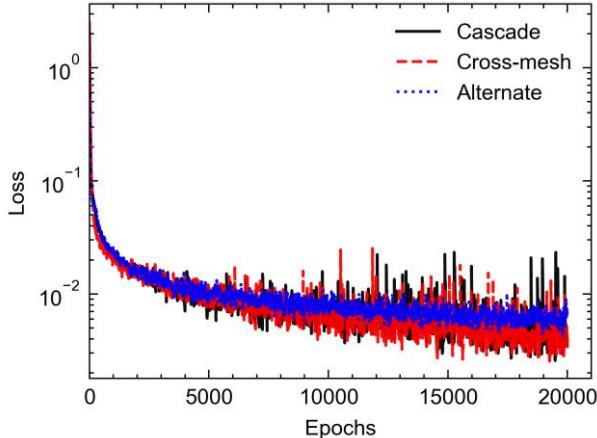


Fig. 7 Comparison of training loss convergence for three quantum circuit topologies in Example 2.

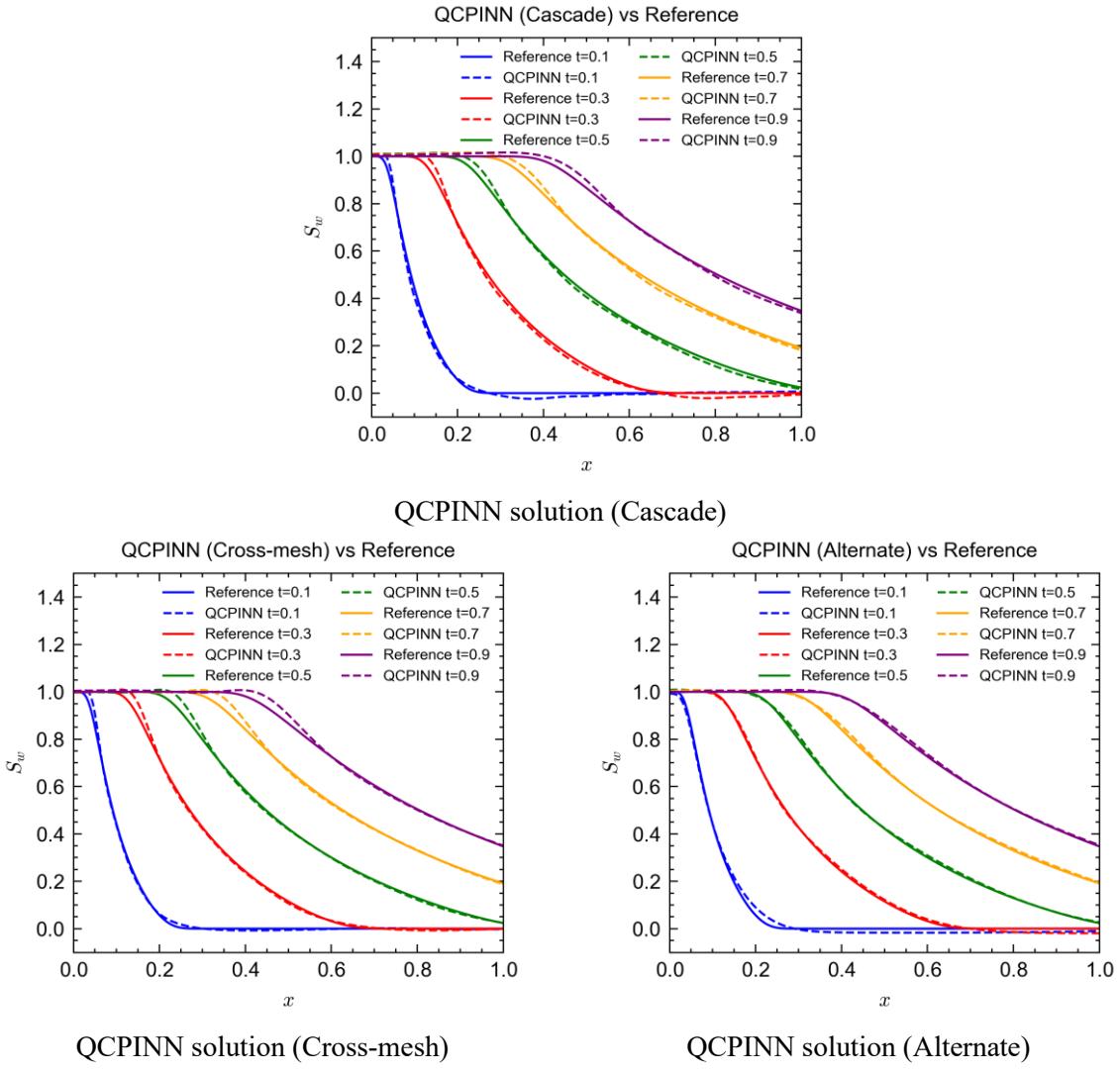


Fig. 8 Comparison of saturation distributions: QCPINN predictions using three topologies vs reference solution at different times.

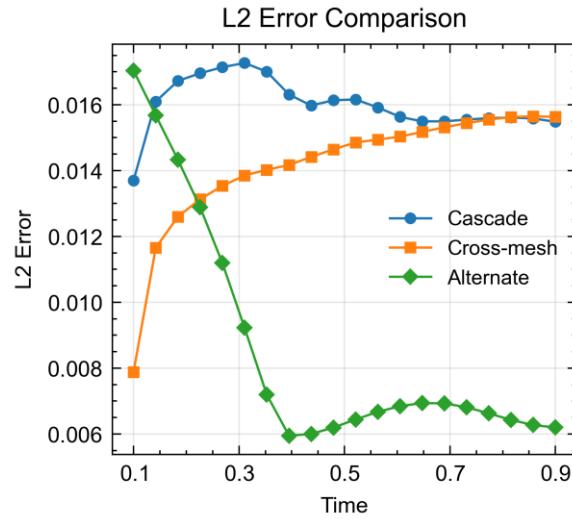


Fig. 9 Relative L2 error comparison of three topologies in Example 2.

4.3 Example 3

This example focuses on the component transport problem considering adsorption as discussed in Section 2.3. As illustrated in Fig. 10, a rectangular reservoir computational domain $\Omega=[0, 100]\text{m} \times [0, 100]\text{m}$ is still adopted. In Eq. (6), suppose $\phi = 0.3$, $\rho_b = 2000 \text{ kg/m}^3$, $K_d = 1.0 \times 10^{-4} \text{ m}^3/\text{kg}$, $D = 0.5 \text{ m}^2/\text{s}$, and $\mathbf{v} = (0.5, 0) \text{ m/s}$. At the initial moment, the component concentration throughout the reservoir is 0 kg/m^3 . A Dirichlet boundary condition is applied to the left boundary, where the solute concentration is maintained at 1 kg/m^3 ; the right boundary has a zero diffusion flux but no restriction on advective flux. The upper and lower boundaries are set as no-flux closed boundaries. The physical process in this example describes the migration behavior of components gradually propagating along the main flow direction under the combined effects of convection, dispersion, and linear adsorption in a uniform seepage field. Due to the retardation effect of linear adsorption, the effective migration velocity of components is lower than the seepage velocity. In the initial stage, components migrate from the high-concentration region of the left boundary to the low-concentration region, forming an adsorption-retarded concentration front that advances along the x -direction over time. Meanwhile, dispersion causes the front to gradually spread during propagation, overall reflecting the mass transport characteristics of convection dominance, adsorption retardation, and dispersion broadening, which fully couples the multi-physical processes of advective transport, molecular-mechanical dispersion, and solid-phase linear adsorption.

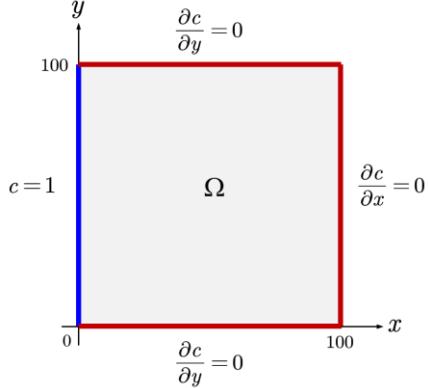


Fig. 10 Schematic of the computational domain and boundary conditions for Example 3.

The QCPINN model takes spatiotemporal coordinates (x, y, t) as input and outputs the target concentration c . Since this example involves multi-physical processes, to accurately characterize the component migration behavior, 6 qubits are used in this case, and three quantum topological structures (Cascade, Cross-mesh, and Alternate) are still adopted as shown in Fig. 11. The remaining parameters and training configurations of the model are the same as those in Example 1.

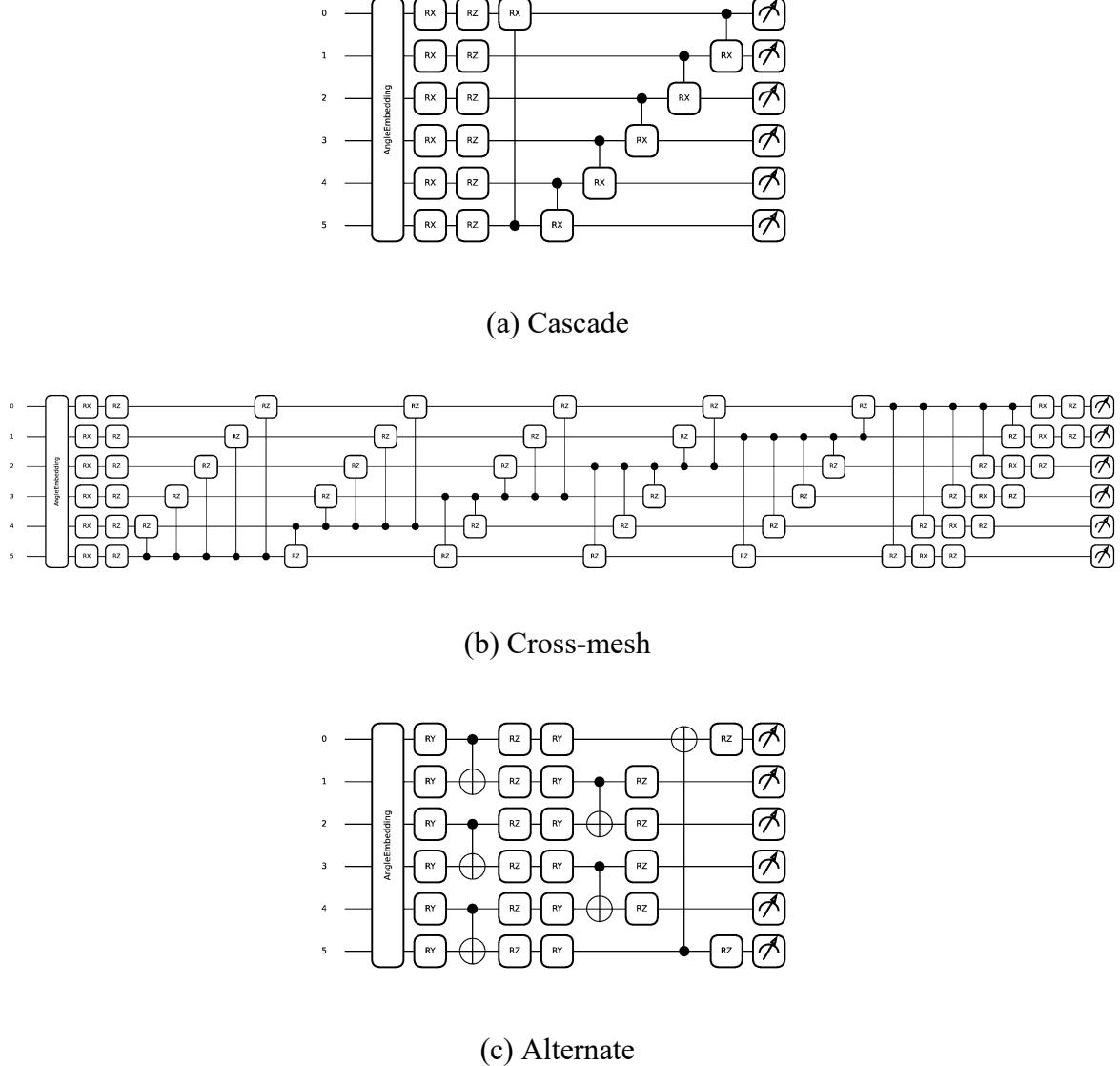


Fig. 11 Three quantum circuit topologies in Example 3.

Fig. 12 shows the convergence process of the loss function during training for the three quantum circuit topologies (Cascade, Cross-mesh, and Alternate). Overall, all three models exhibit excellent convergence performance: the loss value decreases rapidly with the increase in the number of iterations, dropping from the initial order of 10^0 to around the order of 10^{-2} . This indicates that the QCPINN method has strong adaptability in handling such multi-physics coupling problems. Specifically, the Cascade and Cross-mesh topologies show consistent descending trends, with faster convergence speeds and slightly lower final loss function values compared to the Alternate topology. In contrast, the Alternate topology has a relatively higher loss function value in the later stage of training, suggesting that it may face greater optimization resistance than the other two when approximating this complex physical process.

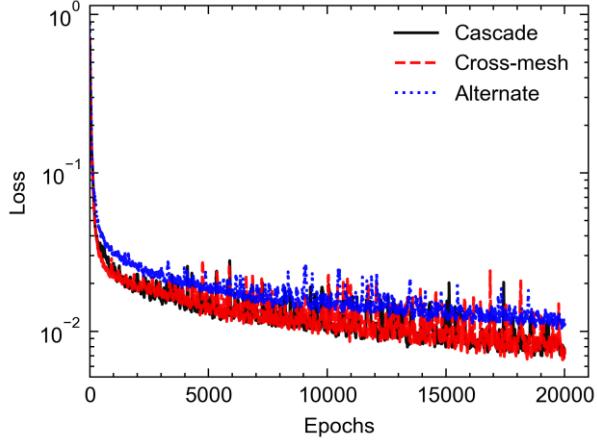


Fig. 12 Training loss over epochs of Example 3.

Figs. 13 and 14 show the comparisons of concentration distributions between the reference solution and the predictions of the three QCPINN topologies at $t=0.2$ and $t=0.6$, respectively. We can see that all QCPINN models successfully capture the concentration front migrating inward from the high-concentration left boundary and clearly reflect the front migration retardation caused by linear adsorption. At $t=0.2$, the front has not yet fully expanded, and the prediction results of the three topologies are highly consistent with the reference solution in terms of front position and shape. By $t=0.6$, the front further advances to the right and gradually broadens due to dispersion, and the QCPINN predictions still maintain good agreement with the reference solution, with no significant distortion especially in the regions where the front gradient changes sharply. Visually, there is no obvious difference between the prediction results of the three topologies and the reference solution, indicating that all models can characterize the convection-dispersion-adsorption coupling process.

Figs. 15 and 16 further present the absolute error distributions of the concentration fields predicted by the three topologies at $t=0.2$ and $t=0.6$, respectively, and the corresponding quantitative error indicators are summarized in Table 3. From the spatial distribution of errors, the main prediction errors show an obvious banded distribution, concentrated in the front region where the concentration changes drastically. This is because the large gradient change in this region increases the difficulty of neural network approximation. Combined with the data analysis in Table 3, the Cascade topology exhibits optimal prediction performance at both moments. Its mean absolute errors (MAE) at $t=0.2$ and $t=0.6$ are 0.018666 and 0.028057, respectively, and both the maximum absolute error and L2 error are significantly better than those of the other two structures. In comparison, the error levels of the Cross-mesh and Alternate topologies vary at $t=0.2$, the Alternate topology performs better than the Cross-mesh; at $t=0.6$, their error levels are relatively close but both slightly higher than that of the Cascade. This indicates that in this example, the Cascade topological structure can more effectively extract spatial features and map the physical laws of component migration.

In summary, for the 2D multi-physics coupling problem involving convection, dispersion, and adsorption, the three quantum topologies tested in this example all verify the effectiveness of the QCPINN method. All three can reproduce the spatiotemporal evolution characteristics

of component migration with high accuracy. In the specific performance comparison, the Cascade topology exhibits the most excellent performance in this example due to its faster convergence speed, and lower average error.

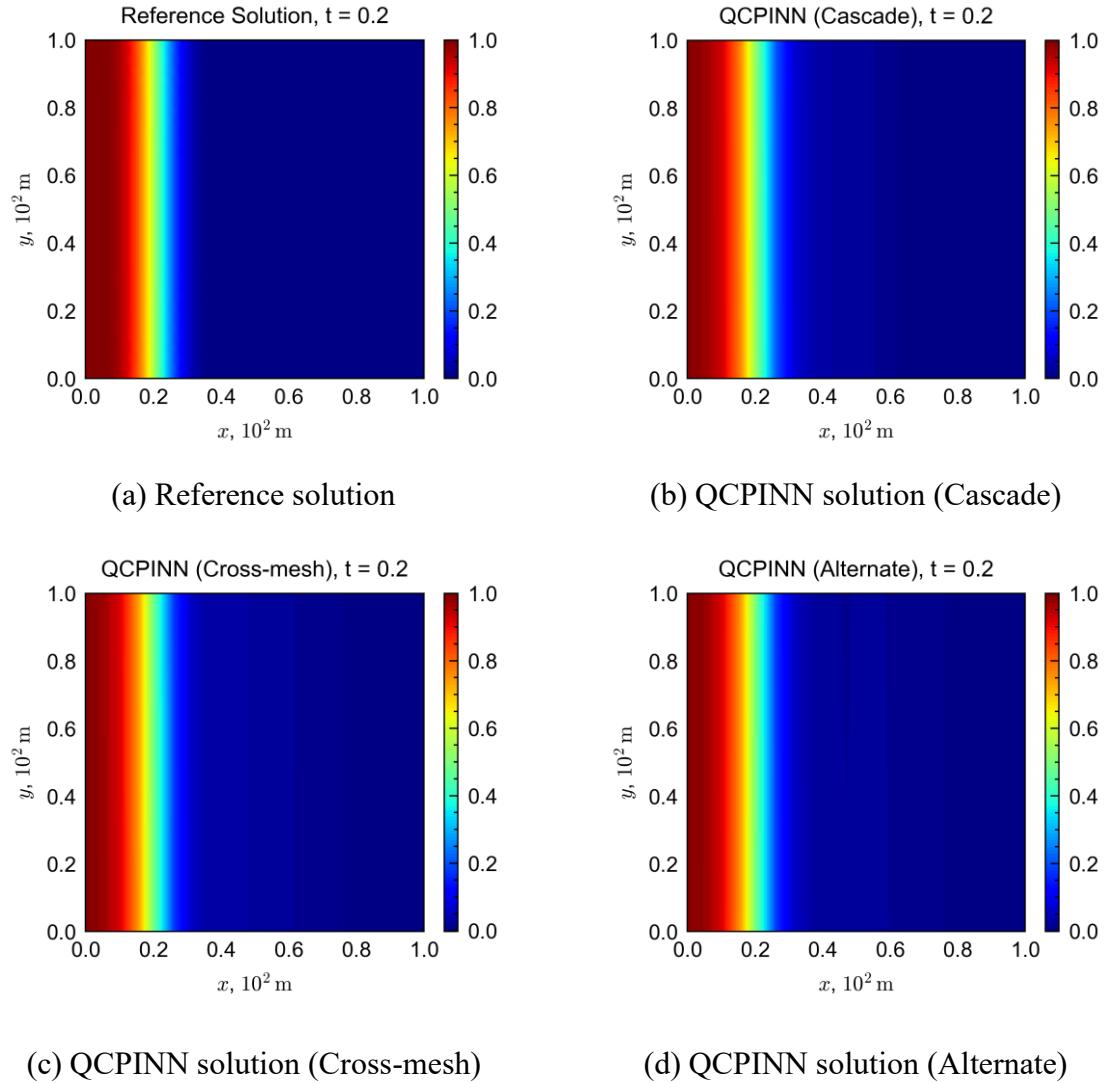


Fig. 13 Comparison of concentration distributions: QCPINN predictions using three topologies vs reference solution at $t=0.2$.

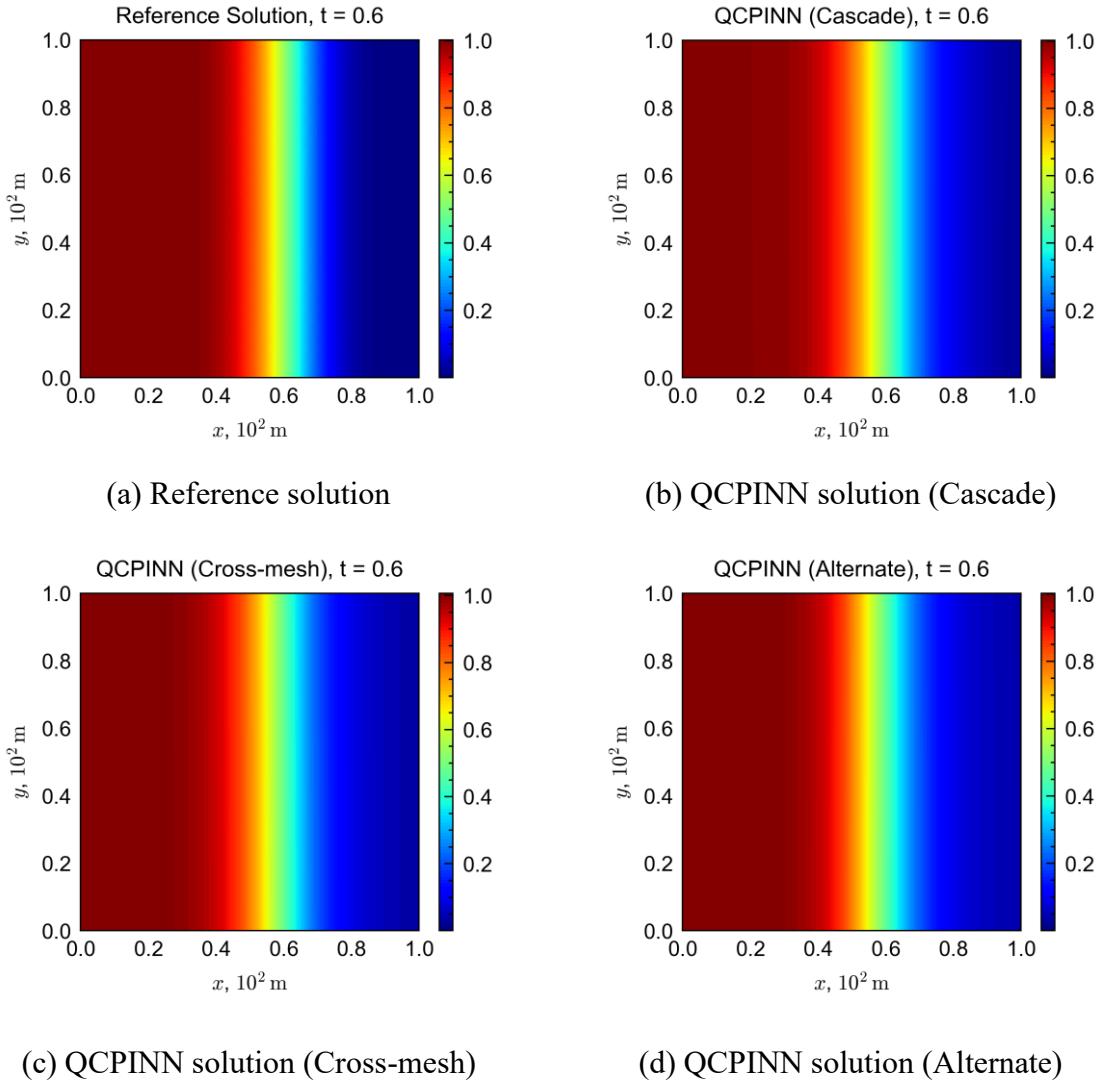


Fig. 14 Comparison of concentration distributions: QCPINN predictions using three topologies vs reference solution at $t=0.6$.

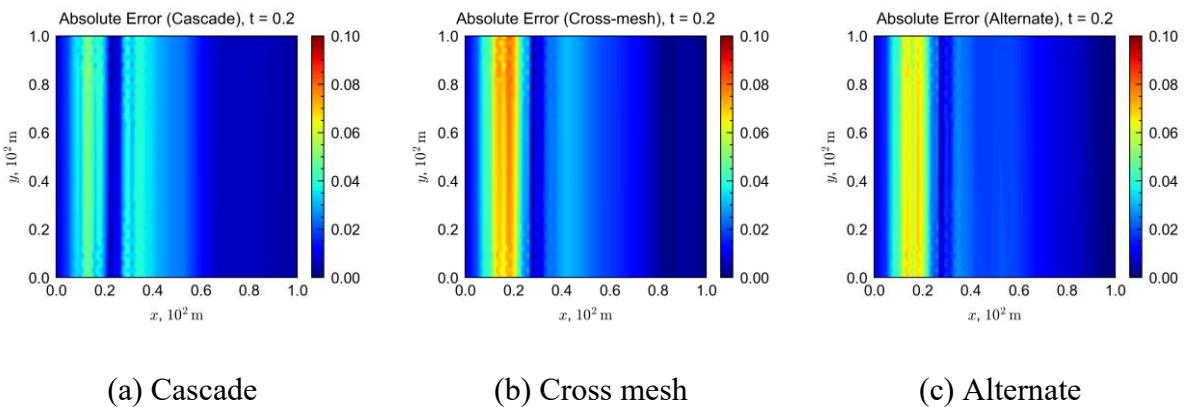


Fig. 15 Absolute error distributions of the concentration predictions for the (a) Cascade, (b) Cross-mesh, and (c) Alternate topologies at $t=0.2$.

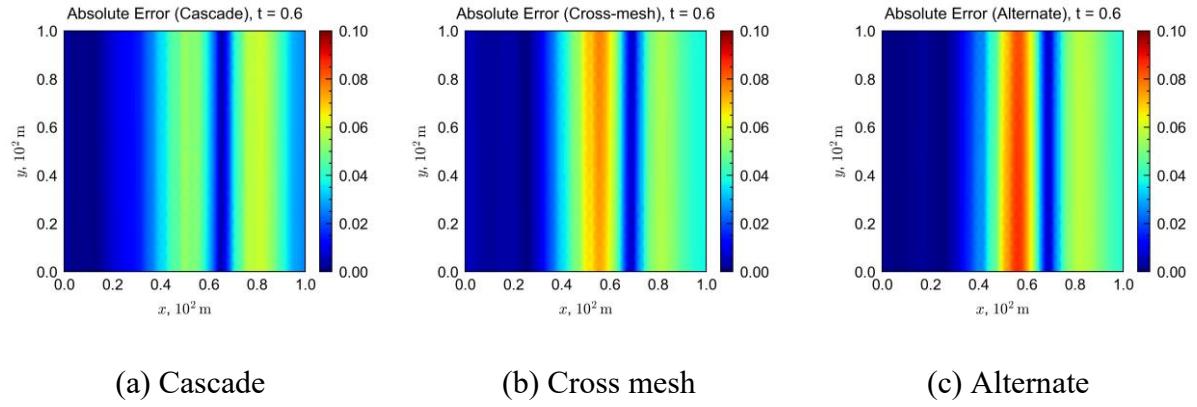


Fig. 15 Absolute error distributions of the concentration predictions for the (a) Cascade, (b) Cross-mesh, and (c) Alternate topologies at $t=0.6$.

Table 3 Comparison of prediction errors for the three quantum circuit topologies in Example 3.

Time	Quantum circuit topology	Mean Absolute Error	Max Absolute Error	L_2 Error
$t = 0.2$	Cascade	0.018666	0.054341	0.023108
	Cross mesh	0.021777	0.082090	0.029449
	Alternate	0.019940	0.071866	0.026243
$t = 0.6$	Cascade	0.028057	0.060663	0.034174
	Cross mesh	0.030991	0.078693	0.039053
	Alternate	0.031103	0.087238	0.040793

5 Conclusions

This study pioneers the application of Quantum-Classical Physics-Informed Neural Networks (QCPINNs) to reservoir numerical simulation, systematically addressing three core seepage PDEs with heterogeneity, strong nonlinearity, and convection dominance. Key conclusions are drawn as follows:

The proposed DV-Circuit QCPINN framework effectively integrates quantum computing advantages with physical constraints. By combining classical networks for feature preprocessing/decoding and a quantum core for high-dimensional mapping, it overcomes the parameter inefficiency and nonlinear fitting limitations of classical PINNs, achieving accurate PDE solutions with reduced computational complexity.

Different quantum circuit topologies exhibit distinct performance in various seepage scenarios: the Alternate topology is optimal for heterogeneous single-phase flow and transient

nonlinear two-phase BL equation, while the Cascade topology excels in multi-physics coupled compositional flow. The Cross-mesh topology shows competitive early-stage accuracy but suffers from increasing errors in transient problems.

Numerical experiments confirm that QCPINNs can reliably capture core reservoir flow characteristics, including pressure distribution in heterogeneous media, steep saturation fronts in waterflooding, and adsorption-retarded concentration migration. Quantitative error analysis indicates that all topologies yield low magnitude errors (MAE as low as 0.0017, L2 error around 0.006), validating the method's accuracy and stability.

This work extends QCPINN from fundamental physics PDEs to industrial reservoir engineering, providing a novel technical route for future high-efficiency reservoir simulators and machine learning surrogate models. Future research can focus on scaling up quantum circuit complexity, optimizing topology design for specific reservoir scenarios, and integrating QCPINNs with field data for inverse problem solving (e.g., permeability inversion).

In summary, QCPINNs offer a breakthrough solution to the limitations of classical numerical methods and PINNs in complex reservoir simulation, laying a foundation for the industrial application of quantum computing in oil and gas field development.

6 Acknowledgements

Prof. Rao gratefully acknowledges the financial support from the General Program of the National Natural Science Foundation of China (Grant No. 52574028), and the National Science and Technology Major Project (Grant No. 2025ZD1401106), and the “Science and Technology Innovation Team” Program of the Xinjiang Uygur Autonomous Region (Grant No. 2024TSYCTD0018).

7 References

- Ali, M., & Kabel, M. (2023). Performance study of variational quantum algorithms for solving the Poisson equation on a quantum computer. *Physical Review Applied*, 20(1), 014054.
- Almajid, M. M., & Abu-Al-Saud, M. O. (2022). Prediction of porous media fluid flow using physics informed neural networks. *Journal of Petroleum Science and Engineering*, 208, 109205.
- Ayoub, F., & Baeder, J. D. (2025). High-entanglement capabilities for variational quantum algorithms: the Poisson equation case: F. Ayoub, JD Baeder. *Quantum Information Processing*, 24(8), 229.
- Bharti, K., Cervera-Lierta, A., Kyaw, T. H., Haug, T., Alperin-Lea, S., Anand, A., ... & Aspuru-Guzik, A. (2022). Noisy intermediate-scale quantum algorithms. *Reviews of Modern Physics*, 94(1), 015004.
- Berger, S., Hosters, N., & Möller, M. (2025). Trainable embedding quantum physics informed neural networks for solving nonlinear PDEs. *Scientific Reports*, 15(1), 18823.
- Bravo-Prieto, C., LaRose, R., Cerezo, M., Subasi, Y., Cincio, L., & Coles, P. J. (2023). Variational quantum linear solver. *Quantum*, 7, 1188.
- Cerezo, M., Arrasmith, A., Babbush, R., Benjamin, S. C., Endo, S., Fujii, K., ... & Coles, P. J. (2021). Variational quantum algorithms. *Nature Reviews Physics*, 3(9), 625-644.

- Chuang, I. L., Vandersypen, L. M., Zhou, X., Leung, D. W., & Lloyd, S. (1998). Experimental realization of a quantum algorithm. *Nature*, 393(6681), 143-146.
- Cirac, J. I., & Zoller, P. (1995). Quantum computations with cold trapped ions. *Physical Review Letters*, 74(20), 4091.
- Dehaghani, N. B., Aguiar, A. P., & Wisniewski, R. (2024, September). A hybrid quantum-classical physics-informed neural network architecture for solving quantum optimal control problems. In 2024 IEEE International Conference on Quantum Computing and Engineering (QCE) (Vol. 1, pp. 1378-1386). IEEE.
- Deutsch, D. (1985). Quantum theory, the Church–Turing principle and the universal quantum computer. *Proceedings of the Royal Society of London A*, 400(1818), 97-117.
- Deutsch, D., & Jozsa, R. (1992). Rapid solution of problems by quantum computation. *Proceedings of the Royal Society of London A*, 439(1907), 553-558.
- Fernandez, B. O., Leao, C., Revelo, D. E., & Fernandes, G. (2025, March). Exploring Hybrid Physics-Informed Neural Networks with Quantum Layers for Solving the Wave Equation. In Fifth EAGE Digitalization Conference & Exhibition (Vol. 2025, No. 1, pp. 1-5). European Association of Geoscientists & Engineers.
- Farea, A., Khan, S., & Celebi, M. S. (2025). QCPINN: Quantum-Classical Physics-Informed neural networks for solving PDEs. *Machine Learning: Science and Technology*.
- Feynman, R. P. (1982). Simulating physics with computers. *International Journal of Theoretical Physics*, 21(6-7), 467-488.
- Fuks, O., & Tchelepi, H. A. (2020). Limitations of physics informed machine learning for nonlinear two-phase transport in porous media. *Journal of Machine Learning for Modeling and Computing*, 1(1).
- Grover, L. K. (1996). A fast quantum mechanical algorithm for database search. In Proceedings of the 28th Annual ACM Symposium on Theory of Computing (pp. 212-219). ACM.
- Govindugari, N. R., & Wong, H. Y. (2024, September). Study of using variational quantum linear solver for solving poisson equation. In 2024 International Conference on Simulation of Semiconductor Processes and Devices (SISPAD) (pp. 01-04). IEEE.
- Hanna, J. M., Aguado, J. V., Comas-Cardona, S., Askri, R., & Borzacchiello, D. (2022). Residual-based adaptivity for two-phase flow simulation in porous media using physics-informed neural networks. *Computer Methods in Applied Mechanics and Engineering*, 396, 115100.
- Harrow, A. W., Hassidim, A., & Lloyd, S. (2009). Quantum algorithm for linear systems of equations. *Physical Review Letters*, 103(15), 150502.
- Havlíček, V., Cárcoles, A. D., Temme, K., Harrow, A. W., Kandala, A., Chow, J. M., & Gambetta, J. M. (2019). Supervised learning with quantum-enhanced feature spaces. *Nature*, 567(7747), 209-212.
- Jäger, J., & Krems, R. V. (2023). Universal expressiveness of variational quantum classifiers and quantum kernels for support vector machines. *Nature Communications*, 14(1), 576.
- Jackson, M. D., Percival, J. R., Mostaghimi, P., Tollit, B. S., Pavlidis, D., Pain, C. C., ... & Blunt, M. J. (2015). Reservoir modeling for flow simulation by use of surfaces, adaptive

- unstructured meshes, and an overlapping-control-volume finite-element method. *SPE Reservoir Evaluation & Engineering*, 18(02), 115-132.
- Kalesh, D., Merembayev, T., Omirbekov, S., & Amanbek, Y. (2025, May). Physics Informed Kolmogorov-Arnold Network for Two-Phase Flow Model with Experimental Data. In International Conference on Computational Science and Its Applications (pp. 447-460). Cham: Springer Nature Switzerland.
- Leong, F. Y., Ewe, W. B., Tran, S. B. Q., Zhang, Z., & Khoo, J. Y. (2025). Hybrid quantum physics-informed neural network: Towards efficient learning of high-speed flows. *Computers & Fluids*, 106782.
- Liu, C. Y., Chen, K. C., & Lin, C. H. A. (2024, June). Learning quantum phase estimation by variational quantum circuits. In 2024 International Joint Conference on Neural Networks (IJCNN) (pp. 1-6). IEEE.
- Liu, Y. Y., Chen, Z., Shu, C., Chew, S. C., Khoo, B. C., Zhao, X., & Cui, Y. D. (2022). Application of a variational hybrid quantum-classical algorithm to heat conduction equation and analysis of time complexity. *Physics of Fluids*, 34(11).
- Monroe, C., Meekhof, D. M., King, B. E., & Wineland, D. J. (1995). Demonstration of a fundamental quantum logic gate. *Physical Review Letters*, 75(25), 4714-4717.
- Peruzzo, A., McClean, J., Shadbolt, P., Yung, M. H., Zhou, X. Q., Love, P. J., ... & O'brien, J. L. (2014). A variational eigenvalue solver on a photonic quantum processor. *Nature communications*, 5(1), 4213.
- Raissi, M., Perdikaris, P., & Karniadakis, G. E. (2019). Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational physics*, 378, 686-707.
- Rao, X., Zhao, H., & Liu, Y. (2022). A meshless numerical modeling method for fractured reservoirs based on extended finite volume method. *SPE Journal*, 27(06), 3525-3564.
- Rao, X. (2024). Performance study of variational quantum linear solver with an improved ansatz for reservoir flow equations. *Physics of Fluids*, 36(4).
- Rao, X. (2024, November). The first application of quantum computing algorithm in streamline-based simulation of water-flooding reservoirs. In Abu Dhabi International Petroleum Exhibition and Conference (p. D011S003R006). SPE.
- Rao, X., He, X., Du, K., Kwak, H., Yousef, A., & Hoteit, H. (2024). A novel projection-based embedded discrete fracture model (pEDFM) for anisotropic two-phase flow simulation using hybrid of two-point flux approximation and mimetic finite difference (TPFA-MFD) methods. *Journal of Computational Physics*, 499, 112736.
- Rao, X., He, X., Kwak, H., & Hoteit, H. (2025a). A hybrid method combining mimetic finite difference and discontinuous Galerkin for two-phase reservoir flow problems. *International Journal for Numerical Methods in Fluids*, 97(4), 484-502.
- Rao, X., Liu, Y., He, X., & Hoteit, H. (2025b). Physics-informed Kolmogorov–Arnold networks to model flow in heterogeneous porous media with a mixed pressure-velocity formulation. *Physics of Fluids*, 37(7).

- Rao, X., Luo, C., He, X., & Hyung, K. (2024, November). An Efficient Quantum Neural Network Model for Prediction of Carbon Dioxide CO₂ Sequestration in Saline Aquifers. In Abu Dhabi International Petroleum Exhibition and Conference (p. D021S061R005). SPE.
- Sedykh, A., Podapaka, M., Sagingalieva, A., Pinto, K., Pflitsch, M., & Melnikov, A. (2024). Hybrid quantum physics-informed neural networks for simulating computational fluid dynamics in complex shapes. *Machine Learning: Science and Technology*, 5(2), 025045.
- Shor, P. W. (1994). Algorithms for quantum computation: discrete logarithms and factoring. In Proceedings 35th Annual Symposium on Foundations of Computer Science (pp. 124-134). IEEE.
- Tilly, J., Chen, H., Cao, S., Picozzi, D., Setia, K., Li, Y., ... & Tennyson, J. (2022). The variational quantum eigensolver: a review of methods and best practices. *Physics Reports*, 986, 1-128.
- Trahan, C., Loveland, M., & Dent, S. (2024). Quantum physics-informed neural networks. *Entropy*, 26(8), 649.
- Tüysüz, C., Rieger, C., Novotny, K., Demirköz, B., Dobos, D., Potamianos, K., ... & Forster, R. (2021). Hybrid quantum classical graph neural networks for particle track reconstruction. *Quantum Machine Intelligence*, 3, 1–20.
- Wang, Q., Moridis, G. J., & Blasingame, T. A. (2023). Wavelet transforms for the simulation of flow processes in porous geologic media. *Transport in Porous Media*, 146(3), 771-803.
- Xiao, Y., Yang, L. M., Shu, C., Chew, S. C., Khoo, B. C., Cui, Y. D., & Liu, Y. Y. (2024). Physics-informed quantum neural network for solving forward and inverse problems of partial differential equations. *Physics of Fluids*, 36(9).