

Physics-informed neural network simulation of two-phase flow in heterogeneous and fractured porous media



Xia Yan ^{a,b}, Jingqi Lin ^{a,b}, Sheng Wang ^{a,b}, Zhao Zhang ^c, Piyang Liu ^d, Shuyu Sun ^{d,e}, Jun Yao ^{a,b}, Kai Zhang ^{a,d,*}

^a School of Petroleum Engineering, China University of Petroleum (East China), Qingdao 266580, China

^b National Key Laboratory of Deep Oil and Gas, China University of Petroleum (East China), Qingdao 266580, China

^c Research Centre for Mathematics and Interdisciplinary Sciences, Shandong University, Qingdao 266237, China

^d Civil Engineering School, Qingdao University of Technology, Qingdao 266520, China

^e Computational Transport Phenomena Laboratory (CTPL), Division of Physical Sciences and Engineering (PSE), King Abdullah University of Science and Technology (KAUST), Thuwal 23955-6900, Saudi Arabia

ARTICLE INFO

Key words:

Deep learning
Physics informed
Two-phase flow
Heterogeneous
Fractured porous media

ABSTRACT

Physics-informed neural networks (PINNs) have received great attention as a promising paradigm for forward, inverse, and surrogate modeling of various physical processes with limited or no labeled data. However, PINNs are rarely used to predict two-phase flow in heterogeneous and fractured porous media, which is critical to lots of subsurface applications, due to the significant challenges in their training. In this work, we present an Enriched Physics-Informed Neural Network (E-PINN) to overcome these barriers and realize the simulation of such flow. Specifically, the Embedded Discrete Fracture Model (EDFM) is adopted to explicitly represent fractures, and then the finite volume method (FVM) instead of the Automatic Differentiation (AD) is used to evaluate spatial derivatives and construct the physics-informed loss function, so that the flux continuity between neighboring elements with different properties (e.g. matrix and fracture) can be defined rigorously. Besides, we develop a novel physics-informed neural network (NN) architecture adopting the adjacency-location anchoring, adaptive activation function, skip connection and gated updating to enrich the pressure information and enhance the learning ability of NN. Additionally, the initial and boundary conditions are constrained through a hard approach, which encodes them into network design, to improve the accuracy and efficiency of network training. In order to further reduce the difficulty of training, the Implicit-Pressure Explicit-Saturation (IMPES) scheme is used to calculate pressure and saturation, in which only the pressure needs to be solved by training NN. Finally, the superiority and applicability of E-PINN to complex practical problems is demonstrated through the simulations of immiscible displacement in 2D/3D heterogeneous and fractured reservoirs.

1. Introduction

Traditional ML typically requires feature engineering, where humans manually select and extract features from raw data and assign weights to them. Conversely, deep learning solutions perform feature engineering with minimal human intervention.

Recently, deep learning (LeCun et al., 2015; Goodfellow et al., 2016; Matsuo et al., 2022) (DL) as an important branch of artificial intelligence has been successfully applied in various engineering applications, such as computer vision (Voulodimos et al., 2018), natural language (Otter et al., 2020), and signal classification (Altaheri et al., 2023). Different

from traditional machine learning (Wang et al., 2021), which typically requires manual feature design and using classifiers (e.g., support vector machine (Pisner and Schnyer, 2020) and decision tree (Charbuty and Abdulazeez, 2021)) to complete learning tasks, DL extracts features with minimal human intervention by designing complex neural networks including convolutional neural network (Li et al., 2021) (CNN), recurrent neural network (Yu et al., 2019) (RNN), graph neural network (Zhou et al., 2020) (GNN), etc., can achieve end-to-end learning mode and greatly improve network performance. For scientific computing applications, most existing DL approaches could be categorized into two types: (1) data-driven approaches (Wu and Xiu, 2020; Reichstein et al.,

* Corresponding author at: No. 66 , West Changjiang Road, Huangdao District 266580, Qingdao, China.

E-mail address: zhangkai@upc.edu.cn (K. Zhang).

2019; Yin et al., 2023), and (2) physics-based approaches (Zhang et al., 2023; Xu et al., 2023; Lv et al., 2024). In the data-driven approaches, sufficient labeled data are necessary for training NN to find the non-linear relationships between inputs and outputs, in which the differences between outputs and labels are represented via the loss function. These approaches have achieved notable success in rapid forecasting. However, the labeled data generation is a big problem that hinders their further applications (Cai et al., 2021). To address this challenge, physics-based approaches are proposed, which directly take into account the underlying physical principles and require little or no labeled data for training NN. Among these approaches, PINNs presented by Raissi et al. (2019) are the most representative, and have been applied for forward, inverse, and surrogate modeling of various physical processes (e.g., solid mechanics (Bai et al., 2023), heat transfer (Xu et al., 2023), and fluid mechanics (Cai et al., 2021)). PINNs approximatively obtain the solution of partial differential equations (PDEs) by training NN to minimize the physics-informed loss function including the terms of PDE residual, initial and boundary conditions. Several open-source libraries have been developed to implement PINNs, for instance, Deep-XDE (Lu et al., 2021), SciANN (Haghigat and Juanes, 2021), and SimNet (Hennigh et al., 2021).

Over the past few years, PINNs and PINNs-like approaches have been applied to simulate and predict fluid flow in porous media, which is critical to lots of subsurface applications (He et al., 2020; Tartakovsky et al., 2020; Sun et al., 2022) like oil and gas extraction, geothermal utilization, and CO₂ geological sequestration. Wang et al. (2020) developed a theory-guided neural network (TgNN), which is trained with a few label data, physics principles (PDEs, initial and boundary conditions), and practical engineering theories (engineering controls, expert knowledge), to predict subsurface flow. Daolun et al. (2021) added the pressure gradient feature as a signpost into PINNs to improve its accuracy in solving the single phase seepage problem with source-sink terms. Fuks and Tchelepi (2020) pointed out the limitations of PINNs for two-phase flow (i.e., it fails to provide reasonable approximations to the solution in the presence of shocks in the saturation field), and added an artificial diffusion term to the primary governing equation of two-phase flow for enabling NN to learn the true solutions. Rodriguez-Torrado et al. (2021) incorporated attention mechanism into PINNs to accurately predict 1D two-phase flow without adding diffusion terms. Almajid and Abu-Al-Saud (2022) investigated the performance of PINNs for simulating the 1D two-phase flow with the availability of label data, the trainability of the multiphase flow parameters, and the addition of a diffusive term, and their results indicated that PINNs can capture the overall trend of true solution without label data, while incorporating label data greatly improves its accuracy. Besides, adding a small amount of diffusion to the PDE-constrained loss function improved the solution slightly only when label data were used. Li et al. (2022) extended TgNN for predicting two-phase seepage by using two independent NNs to approximate pressure and saturation, respectively. Kashefi and Mukerji (2023) included sparse observational data in physics-informed PointNet to predict pore-scale fluid flow within porous media. Du et al. (2023) implemented PINNs with a scheme based on time-domain decomposition to study the flow process of CO₂ in homogeneous formations. Lehmann et al. (2023) developed a new implementation of PINNs, in which the training loss function is based on the decoupled continuity equation and Darcy's law, for simulating the flow in heterogeneous porous media. The numerical experiments of single-phase steady flow show that compared with a technique suggested in the literature (He et al., 2020) where the hydraulic conductivity field is interpolated by a NN, their proposed method could provide more accurate automatic differentiation of the neural networks for pressure and velocity fields. Different from their work, we will solve the heterogeneous two-phase transient flow problem via the numerical discretization with FVM. In above mentioned studies, the AD technique is adopted to calculate the PDE residual, which is unsuited for the deep learning of fluid flow in heterogeneous porous media considering

source/sink terms without labeled data (Zhang, 2022). This is because that the remarkable gradient changes caused by source/sink terms and the heterogeneous physical properties impose challenges to the performance of AD (Zhang, 2022). Hereto, we developed a physics-informed convolutional neural network (Zhang et al., 2023) (PICNN), in which the flux continuity between neighboring elements with different properties is defined strictly by using the FVM (Eymard et al., 2000), and the large pressure gradient near source/sink is approximated using a well model. Although some methods (e.g. the implementation suggested by Lehmann et al.) can be extended to simulate heterogeneous two-phase flow, PICNN is currently the only physics-informed neural network that has realized the simulation of two-phase transient flow in heterogeneous porous media containing source/sink terms without labeled data to our best knowledge. However, PICNN has only been applied to 2D heterogeneous problems, and its performance is not well enough in training efficiency. This is because that the physical information of PICNN, as well as other PINNs, is only reflected in the loss function, while the NN architecture is still lack of physical information. In addition, CNN used in PICNN is hard to deal with the irregular models with unstructured grids (Gao et al., 2021), thus PICNN cannot be straightly used for fractured porous media with discrete fracture model.

In this study, an enriched physics-informed neural network named E-PINN is first proposed to simulate two-phase flow in 2D/3D heterogeneous and fractured porous media without labeled data. Specifically, the EDFM (Yan et al., 2016; Yan et al., 2018; Yan et al., 2019; Losapio and Scotti, 2023) is adopted to explicitly represent fractures, and the physics-informed loss function is constructed by means of FVM. Furthermore, a novel physics-informed NN architecture adopting the adjacency-location anchoring, adaptive activation function, skip connection and gated updating is developed to enrich the pressure information and enhance the learning ability of NN. Because this novel NN is based on fully connected neural network (Sainath et al., 2015) (FNN), it can be applied to both regular and irregular models. Additionally, the initial and boundary conditions are constrained through a hard approach, which encodes them into network design, to improve the accuracy and efficiency of network training. The simulation of two-phase flow in porous media is usually implemented by using fully implicit scheme (Saad and Saad, 2013) and IMPES scheme (Chen et al., 2019). The fully implicit scheme has the advantage of unconditional stability, but its calculation at each time step is extremely time-consuming since the pressure and saturation need to be solved simultaneously. Different from the fully implicit scheme, the main idea of the IMPES scheme is to separate the computation of pressure from that of saturation, then the pressure and saturation equations are solved using implicit and explicit time approximation schemes respectively. Although the IMPES scheme is not unconditionally stable (limitations on the time step) (Redondo et al., 2018), it achieved great popularity because that it requires less computing time at each time step and is easier to implement. In this study, we adopt the IMPES scheme to decouple the pressure and saturation equations. After that, we can solve the pressure first and then the saturation. As the pressure is known when calculating the saturation, the saturation can be updated with an explicit numerical scheme. The advantage of doing this is that only the pressure needs to be solved using the NN during the entire simulation period, which can reduce the difficulty of NN training and improve the efficiency of numerical simulation.

The structure of this paper is organized as follows: First, we introduce the governing equations for the two-phase flow in porous media; Second, we describe the network structure and training strategy of E-PINN; Third, we perform case studies to demonstrate the superiority and applicability of E-PINN. Finally, we conclude this study and point out further work.

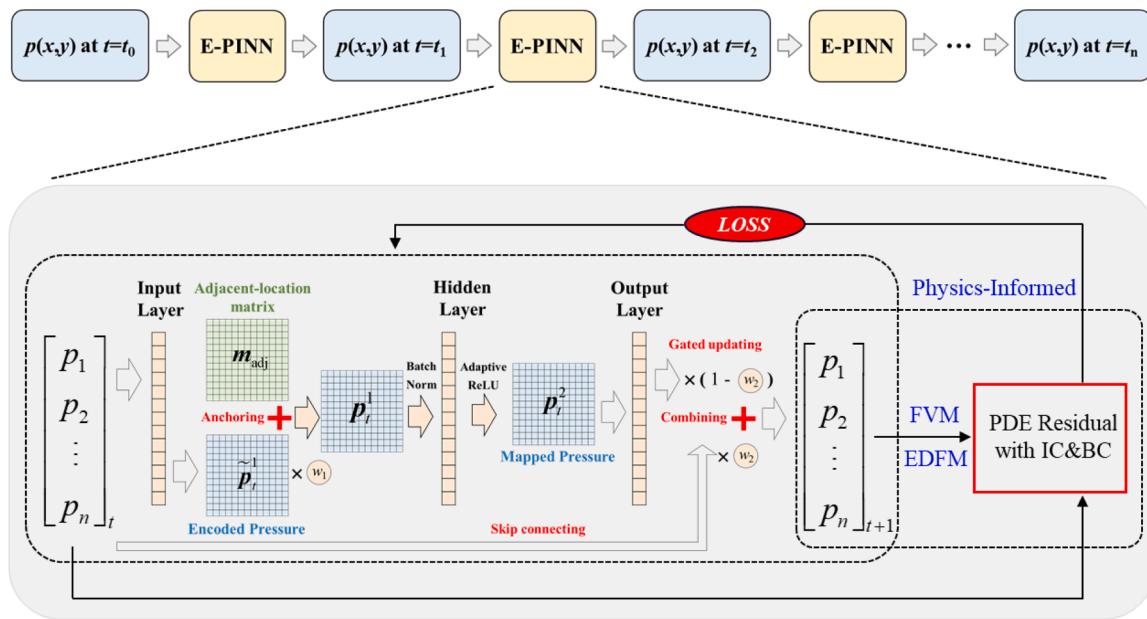


Fig. 1. The procedure of E-PINN-based pressure prediction.

2. Methodology

2.1. Governing equations of two-phase flow in porous media

In this study, we consider the immiscible and slightly compressible two-phase flow in porous media. The mass conservation equation for each phase is (Zhang et al., 2023; Moortgat and Firoozabadi, 2013)

$$\frac{\partial}{\partial t}(\phi\rho_\beta S_\beta) + \nabla \cdot (\rho_\beta \mathbf{v}_\beta) = \rho_\beta q_\beta \quad (1)$$

where subscript β indicates fluid phases, $\beta=n$ for non-wetting phase and $\beta=w$ for wetting phase; ϕ is porosity; ρ , S , \mathbf{v} , and q indicate density, saturation, velocity, and source/sink of the fluid, respectively. The flow of each phase in porous media conforms to the Darcy's law, as (Moortgat and Firoozabadi, 2013)

$$\mathbf{v}_\beta = -\frac{k_{r\beta}}{\mu_\beta} k (\nabla p_\beta - \rho_\beta g \nabla D) \quad (2)$$

where k and k_r represent absolute permeability and relative permeability, respectively; μ and p are viscosity and pressure of fluid; g and D indicate the gravity acceleration and depth. Additionally, the following auxiliary equations need to be provided

$$S_w + S_n = 1 \quad (3)$$

$$p_c = p_n - p_w \quad (4)$$

where p_c indicates the capillary force, which is neglected in this study as its influence on fluid flow is considerably lower than other factors (Zhang et al., 2023), thus $p_n = p_w = p$. Consider the compressibilities of fluid and porous media, i.e., $\rho_\beta = \rho_{\beta 0} \exp[C_\beta(p-p_0)]$ and $\phi = \phi_0 \exp[C_r(p-p_0)]$, subscript 0 represents the initial state, C_r is the isothermal compressibility of porous media, and C_β is the isothermal compressibility of phase β . By defining $\Phi_\beta = p - \rho_\beta g D$, substituting Eq. (2) into Eq. (1) and ignoring the term introduced by spatial gradient of fluid density as the fluid is slightly compressible (Martin, 1959; Coats, 2000), we can approximate Eq. (1) as

$$\phi \frac{\partial S_\beta}{\partial t} + S_\beta \phi (C_\beta + C_r) \frac{\partial p}{\partial t} = \nabla \cdot (\lambda_\beta k \nabla \Phi_\beta) + q_\beta \quad (5)$$

where $\lambda = k_r/\mu$ is fluid mobility. To implement the IMPES scheme, we sum up the equations of non-wetting phase and wetting phase to yield the following pressure equation

$$\phi(S_n C_n + S_w C_w + C_r) \frac{\partial p}{\partial t} = \nabla \cdot (\lambda_n k \nabla \Phi_n) + \nabla \cdot (\lambda_w k \nabla \Phi_w) + q_n + q_w \quad (6)$$

Since the densities of non-wetting phase and wetting phase are usually different, their flow potential is also different, i.e., $\Phi_n \neq \Phi_w$. By defining $\Phi_c = \Phi_n - \Phi_w = (\rho_w - \rho_n)gD$, Eq. (6) can be written as

$$\phi(S_n C_n + S_w C_w + C_r) \frac{\partial p}{\partial t} = \nabla \cdot (\lambda_t k \nabla \Phi_w) + \nabla \cdot (\lambda_n k \nabla \Phi_c) + q_n + q_w \quad (7)$$

where $\lambda_t = \lambda_w + \lambda_n$ is the total mobility. Therefore, the two-phase flow simulation can be carried out by solving Eqs. (5) and (7) sequentially. Specifically, Eq. (7) is solved implicitly for pressure at current timestep, using the saturation at previous timestep to calculate the relevant parameters. Then, the calculated pressure is substituted into Eq. (5) to explicitly calculate saturation. Please note that above governing equations hold in both matrix and fracture. In fractured porous media, different properties are applied to matrix and fracture.

2.2. The structure of enriched physics-informed neural network

The procedure of E-PINN-based pressure prediction and the structure of E-PINN are illustrated in Fig. 1. The input and output of E-PINN are the pressure fields at time t and $t+1$, respectively. These two pressure fields are used for constructing the physics-informed loss function to train NN and update the pressure field at time $t+1$. After the pressure is solved by using E-PINN, the saturation can be updated explicitly.

A novel physics-informed NN architecture is designed to efficiently and accurately predict the variation of pressure with time. Specifically, the neural network mapping adopts the form of FNN, which can ensure the universal applicability for 2D, 3D, regular, and irregular models. The input of NN is the pressure of n elements at time t , which is stored as a column vector, and its output is the pressure of n elements at time $t+1$. Besides, we design and adopt the following special structures to enrich the pressure information and enhance the learning and memory ability of the network.

(1) Adjacency-location anchoring

Inspired by the anchor graph (Liu et al., 2022) and connectivity pattern (Benson et al., 2016), we design an anchoring structure to add the connection information between neighboring elements into NN architecture. Specifically, the feature dimension of input pressure vector \mathbf{p}_t is raised from 1 to n via the input layer mapping, thus the encoded pressure matrix $\tilde{\mathbf{p}}_t^1$ of $n \times n$ is obtained, then we multiply $\tilde{\mathbf{p}}_t^1$ with a learnable scaling weight w_1 and add them with the adjacent-location matrix \mathbf{m}_{adj} of $n \times n$ to get the anchored pressure matrix \mathbf{p}_t^1 . Therefore, the mapping process for this part is as follows

$$\begin{aligned}\tilde{\mathbf{p}}_t^1 &= \mathbf{W}_1 \cdot \mathbf{p}_t + \mathbf{b}_1 \\ \mathbf{p}_t^1 &= \tilde{\mathbf{p}}_t^1 \cdot w_1 + \mathbf{m}_{\text{adj}}\end{aligned}\quad (8)$$

where \mathbf{W}_1 and \mathbf{b}_1 are input-layer trainable weights and biases, respectively. The i th row of \mathbf{m}_{adj} represents the connection relationship of i th element, such as $\mathbf{m}_{\text{adj}}(i, j)$ indicates the connection relationship between the i th element and j -th element, and its value is 1 or 0 for connected and unconnected, respectively. In order to establish the mapping relationship between pressure fields, NN needs to learn both spatial topological information (i.e., connection relationship) and feature information (i.e., pressure value). As the connection relationship between neighboring elements is known for E-PINN, the anchoring structure utilizes this prior knowledge for guiding NN to focus on learning feature information. Therefore, it can reduce the learning difficulty of NN and improve its training accuracy.

(2) Adaptive activation function

Firstly, we apply BatchNorm (Santurkar et al., 2018) to normalize the anchored pressure matrix \mathbf{p}_t^1 to maintain the stability for gradient backpropagation. Then, we input the normalized pressure matrix into hidden-layer for mapping and activate it with the adaptive ReLU function (Jagtap et al., 2020), in which a learnable weight α is added.

$$\mathbf{p}_t^2 = \text{ReLU}((\mathbf{W}_2 \cdot \text{BatchNorm}(\mathbf{p}_t^1) + \mathbf{b}_2) \cdot \alpha) \quad (9)$$

where \mathbf{p}_t^2 indicates the nonlinear mapped pressure matrix; \mathbf{W}_2 and \mathbf{b}_2 are hidden-layer trainable weights and biases, respectively. Compared to conventional ReLU function, the adaptive one has more superior learning ability because of its improvements in the convergence rate.

(3) Skip connection and Gated updating

For the mapping transformation in output-layer, skip connection and gated updating are designed. Specifically, we use skip connection to directly transfer the input pressure vector \mathbf{p}_t to output-layer, and perform linear mapping on pressure matrix \mathbf{p}_t^2 , then the following gated updating strategy is applied to combine them as the network output

$$\mathbf{p}_{t+1} = \mathbf{w}_2 \cdot \mathbf{p}_t + (1 - \mathbf{w}_2) \cdot (\mathbf{W}_3 \cdot \mathbf{p}_t^2 + \mathbf{b}_3) \quad (10)$$

where \mathbf{p}_{t+1} indicates the pressure vector at time $t + 1$; \mathbf{w}_2 is a learnable weight coefficient; \mathbf{W}_3 and \mathbf{b}_3 are output-layer trainable weights and biases, respectively. By using skip connection and gated updating, it can be more reasonable to refer to previous pressure distribution during network training. Therefore, the learning difficulty of NN is further reduced, and the convergence of later training can be improved.

The developed NN is implemented based on PyTorch (Paszke et al., 2019) (Version 1.12.1). The number of neurons in both input-layer and hidden-layer is n , which is the number of elements. Besides, the number of output neurons is 1. In this study, the initial values for w_1 , w_2 , and α are set to 0.01, 0.5, and 0.1, respectively. Note that these initial values

are obtained through numerical experiments, ensuring that the proposed method has a certain degree of universality for different problems in this study. To further improve the NN performance, hyperparameter optimization can be used to determine the optimal initial values for specific problems. Other network parameters are initialized by means of Xavier method (Datta, 2020). The training of NN is performed by using the Adam optimizer (Kingma and Adam, 2014), whose parameters are set to default values (Reyad et al., 2023) except for the learning rate.

2.3. The training strategy of enriched physics-informed neural network

In existing PINNs, the physics-informed training is implemented by minimizing a composite loss function taking the form (Cai et al., 2021; Cuomo et al., 2022)

$$MSE = MSE_r + MSE_i + MSE_b \quad (11)$$

where MSE is the mean square error loss function (Zhang et al., 2023) used to train network parameters; MSE_r , MSE_i , and MSE_b represent the loss associated with the PDE residual, the initial condition (IC), and the boundary condition (BC), respectively.

Usually, IC and BC are constrained through the soft enforcement (i.e., adding extra loss terms), as shown in Eq. (11). However, this approach faces challenges in terms of accuracy and efficiency (Cuomo et al., 2022). To solve this problem, we use a hard approach to constrain IC and BC by encoding them into the network design. On the other hand, MSE_r is commonly evaluated at the selected domain points and calculated by using AD. However, this approach suffers from unstable performance while handling flow problems in highly heterogeneous porous media with sources and sinks (Zhang, 2022). Hereto, we construct MSE_r via the FVM so that the flux continuity between neighboring elements with different properties can be defined strictly. Besides, the following well model (Peaceman, 1993) is utilized to approximate the large pressure gradient near source/sink

$$q_\beta = \lambda_\beta WI(p - p_{wf}) \quad (12)$$

where p and p_{wf} are the pressures of the grid where the well is located and the bottom hole pressure (BHP); WI indicates the well production index, defined as

$$WI = \frac{2\pi kh}{\ln(r_e/r_w) + S} \quad (13)$$

where h represents the thickness of connected-formation; r_w and r_e indicate wellbore radius and equivalent radius, respectively; S is well skin factor, which is a technical term in petroleum engineering (Yildiz, 2006). Here, we use Peaceman method (Peaceman, 1993) to calculate equivalent radius, as $r_e = 0.14\sqrt{dx^2 + dy^2}$.

To derive the FVM formulation for pressure equation, the divergence theorem is substituted into the integral formulation of Eq. (7), and then the time derivative is evaluated using a standard first-order approximation

$$\begin{aligned}\frac{\phi(S_n C_n + S_w C_w + C_r) V_i}{\Delta t} (p_i^{t+1} - p_i^t) &= \sum_{j \in G_i} \left(\lambda_{t,ij+1/2} T_{ij} \left(\Phi_{wj}^{t+1} - \Phi_{wi}^{t+1} \right) \right) \\ &\quad + \sum_{j \in G_i} \left(\lambda_{n,ij+1/2} T_{ij} \left(\Phi_{cj}^t - \Phi_{ci}^t \right) \right) \\ &\quad + (q_n + q_w) V_i\end{aligned}\quad (14)$$

where subscript i and j are element number, $ij+1/2$ represents the intermediate value between two adjacent elements, such as upstream weight and harmonic mean; superscript t is time level; Δt is time step; G_i is the adjacent element set of element i ; V is element volume; λ is evaluated explicitly and its upstream weight value is adopted; T_{ij} is the transmissibility between elements i and j , defined as

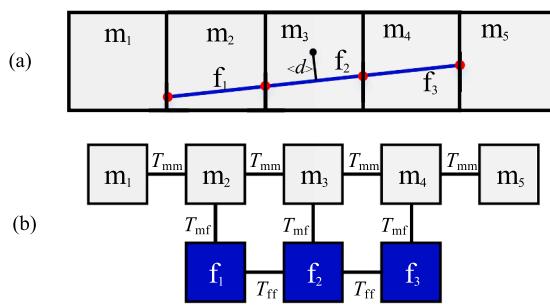


Fig. 2. Schematic of geometry discretization (a) and connection list (b) of EDFM (blue line indicates fracture).

$$T_{ij} = \frac{A_{ij} k_{ij+1/2}}{d_i + d_j} \quad (15)$$

where A is the interface area; d is the vertical distance from element center to interface; $k_{ij+1/2}$ is the harmonic mean absolute permeability. We simulate fluid flow in fractures via the EDFM, which does not need grids to match with fractures and can realize high precision with non-refined grids. Therefore, cartesian grids are used to discretize matrix, while the discretization of fractures is according to the intersections of fractures and grids, Fig. 2(a). Please note that although orthogonal grids are used for matrix, the grids of EDFM is still unstructured due to fracture elements. There are three types of transmissibility, i.e., T_{mm} (matrix to matrix), T_{mf} (matrix to fracture), and T_{ff} (fracture to fracture), as shown in Fig. 2(b). T_{mm} and T_{ff} can be calculated by using Eq. (15), whereas T_{mf} is defined as (Yan et al., 2016)

$$T_{mf} = \frac{A_{mf} k_{mf}}{\langle d \rangle} \quad (16)$$

where A_{mf} is the fracture area in grid; k_{mf} is the harmonic mean absolute permeability of matrix and fracture; $\langle d \rangle$ indicates the average normal distance between matrix and fracture, as shown in Fig. 2(a). For fractures that partially penetrate grid, T_{mf} is assumed linearly proportional to the fracture length within grid. If there are intersecting fractures, the star-delta transformation (Karimi-Fard et al., 2004) will be used.

We substitute NN output into Eq. (14) to approximate the residual of governing equations, and the residual of i th element is

$$\begin{aligned} R_i &= \frac{\phi(S_n C_n + S_w C_w + C_r) V_i}{\Delta t} (p_i^* - p_i^t) - (q_n + q_w) V_i \\ &- \sum_{j \in G_i} \left(\lambda_{t,ij+1/2} T_{ij} (\Phi_{w,j}^* - \Phi_{w,i}^*) \right) - \sum_{j \in G_i} \left(\lambda_{n,ij+1/2} T_{ij} (\Phi_{c,j}^t - \Phi_{c,i}^t) \right) \end{aligned} \quad (17)$$

where p^* is the NN output that represents p^{t+1} , and $\Phi_w^* = p^* - \rho_w g D$.

Algorithm 1

The training algorithm for simulating two-phase flow.

-
- 1: inputs: simulation time T , time step Δt , training tolerance ϵ , and maximum epoch number M
 - 2: **while** $t < T$ **do**
 - 3: if $t =$ first time: initialize NN by using Xavier method, and input IC
 - 4: if $t >$ first time: initialize NN by using the trained NN of previous time step
 - 5: update equation parameters (ϕ , λ , ρ , Φ_c , etc.) explicitly
 - 6: $m = 1$
 - 7: **while** $m < M$ and $MSE > \epsilon$ **do**
 - 8: input pressure at previous time step p^t into NN to get current approximate pressure p^*
 - 9: calculate residuals R_i and then evaluate loss MSE by using Eqs. (17) and (18)
 - 10: optimize NN parameters according to MSE
 - 11: $m = m + 1$
 - 12: **end while**
 - 13: take p^* associated with the lowest loss as p^{t+1}
 - 14: update wetting phase saturation S_w^{t+1} by using Eq. (19)
 - 15: $t = t + \Delta t$
 - 16: **end while**
-

In E-PINN, the IC (i.e., initial pressure and wetting phase saturation) is used as the NN input at first time step and straightly imposed where it is needed in Eqs. (17) and (19), besides the BC can also be straightly imposed in Eq. 17 by replacing p^* at boundary points with observed value, thus MSE is equal to MSE_r

$$MSE = MSE_r = \frac{1}{N_e} \sum_{i=1}^{N_e} R_i^2 \quad (18)$$

where N_e is the number of elements. Note that each element can be seen as a collocation point with control volume in E-PINN, which is different from the collocation point in traditional PINNs. In other words, the number of elements is the number of collocation points. Therefore, E-PINN is feasible for the large problems of thousands or millions of elements.

The exact value of p^{t+1} is obtained by training NN (minimising MSE). After the pressure is obtained, wetting phase saturation can be updated explicitly according to the following equation, which is derived from the discretized formulation of Eq. (5)

$$\begin{aligned} S_{w,i}^{t+1} &= S_{w,i}^t + \frac{\sum_{j \in G_i} \left(\lambda_w T_{ij} (\Phi_{w,j}^{t+1} - \Phi_{w,i}^{t+1}) \right) \Delta t}{\phi V_i} + \frac{q_w \Delta t}{\phi} \\ &- S_{w,i}^t (C_\beta + C_r) (p_i^{t+1} - p_i^t) \end{aligned} \quad (19)$$

Because the pressure fields between two consecutive time steps change gradually, the idea of transfer learning can be utilized to improve training efficiency in this study. Specifically, the trained NN of previous time step is used as the initial NN of current time step. In addition, training efficiency can be further improved by normalizing the inputs and outputs of NN. Note that we have to re-train the NN whenever we need the solution at the next time step. However, very few training is required by utilizing the idea of transfer learning, and this is also the main reason that the computational efficiency of E-PINN is expected to exceed that of the traditional numerical method, in which the large-scale matrix inversion operation is required at each time step. The complete training algorithm is as follows (Algorithm 1).

3. Results and discussions

To show the accuracy and applicability of E-PINN, a series of immiscible displacement examples, in which wetting phase is injected into reservoirs to displace non-wetting phase, are presented in this section. We compare E-PINN and PICNN based on 2D and 3D heterogeneous reservoirs, and demonstrate the superiority of E-PINN. On the other hand, we apply E-PINN to simulate the displacement process in 2D and 3D fractured reservoirs and show its adaptability to complex practical problems, such as variable injection-production schemes and gravity segregation. All the reference solutions mentioned in this section

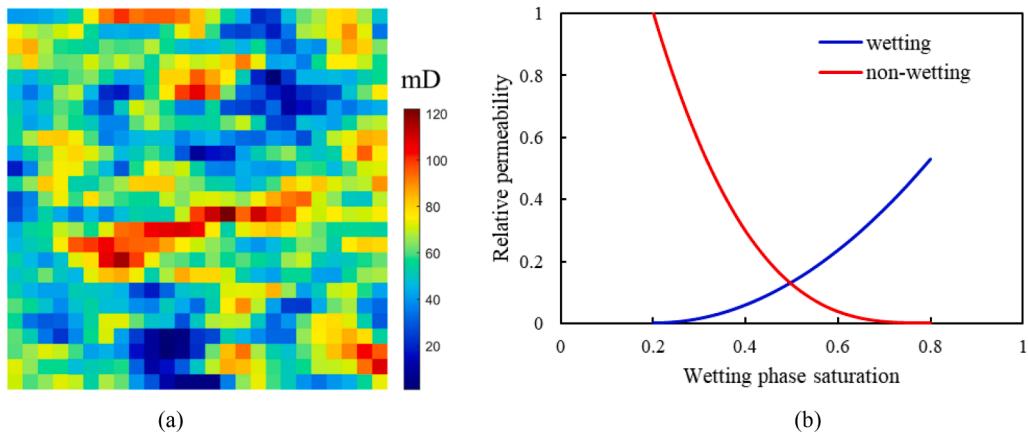


Fig. 3. The permeability distribution (a) and relative permeability curve (b) of 2D heterogeneous reservoir.

Table 1
The summary of 2D reservoir parameters.

Parameter	Value	Parameter	Value
matrix porosity	0.25	wetting phase compressibility,	1E-
matrix compressibility, 1/	1E-8	1/Pa	10
Pa	1000	non-wetting phase	1E-
wetting phase density, kg/	800	compressibility, 1/Pa	10
m ³	1.0	irreducible wetting phase	0.2
non-wetting phase density,	1.8	saturation	0.2
kg/m ³		residual non-wetting phase	0.1
wetting phase viscosity,		saturation	0.0
mPa•s		wellbore radius, m	
non-wetting phase viscosity,		well skin factor	
mPa•s			

are calculated via the in-house reservoir numerical simulator (Yan et al., 2016), and its accuracy has been verified. The NN architecture, optimization parameter and training time for each example are summarized in Appendix A.

3.1. 2D immiscible displacement examples

3.1.1. Heterogeneous reservoir

We use a 2D heterogeneous reservoir model (100m×100 m) with the cartesian grid made up of 25 × 25 elements, and the thickness of this reservoir is 1 m. A wetting phase injection well and a production well are

located at lower left (2 m, 2 m) and upper right (98 m, 98 m), respectively. The flow rates of injection well and production well are both 2 m³/day. Note that the production well will produce both non-wetting and wetting phases after wetting phase breakthrough. We have no flow at reservoir boundaries. The initial pressure and wetting phase saturation of this reservoir are 25 MPa and 0.2. The permeability distribution and relative permeability curve are shown in Fig. 3. Other reservoir parameters are summarized in Table 1.

We apply E-PINN and PICNN to simulate the displacement process for 1000 days with $\Delta t = 0.01$ day. To make a fair comparison, the only difference between E-PINN and PICNN is the network structure. For each time step, the training stops when MSE is less than 1.0E-18. Besides, the maximum epoch numbers of training for the first time step and all other time steps are 2000 and 500, respectively. The network output with the least loss is used as the pressure solution for each time step. Training on the NVIDIA GeForce RTX 4070 Ti GPU with learning rate 0.01, the training costs of PICNN and E-PINN for the entire simulation period (i.e., 1000 days) are about 10170s and 878 s, respectively. This means that the computational efficiency of E-PINN for simulating 2D heterogeneous two-phase flow is more than 10 times faster than that of PICNN. To quantitatively evaluate the accuracy of PICNN and E-PINN, we adopt the following relative L_2 error

$$L_2(\mathbf{p}^*, \mathbf{p}_{ref}) = \frac{\|\mathbf{p}^* - \mathbf{p}_{ref}\|_2}{\|\mathbf{p}_{ref}\|_2} \quad (20)$$

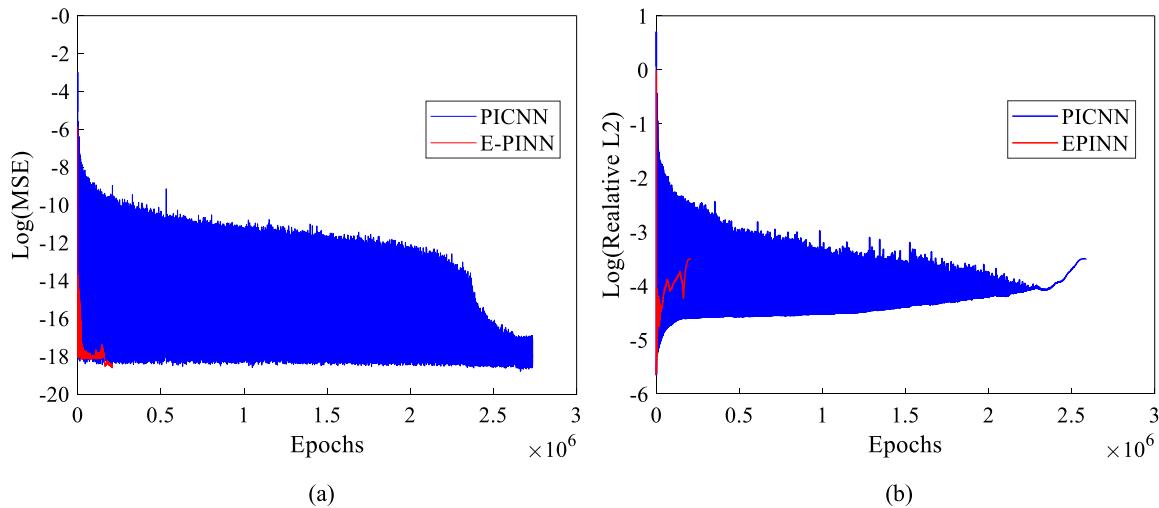


Fig. 4. The changes in MSE (a) and relative L_2 error (b) over all epochs of PICNN and E-PINN for 2D heterogeneous reservoir.

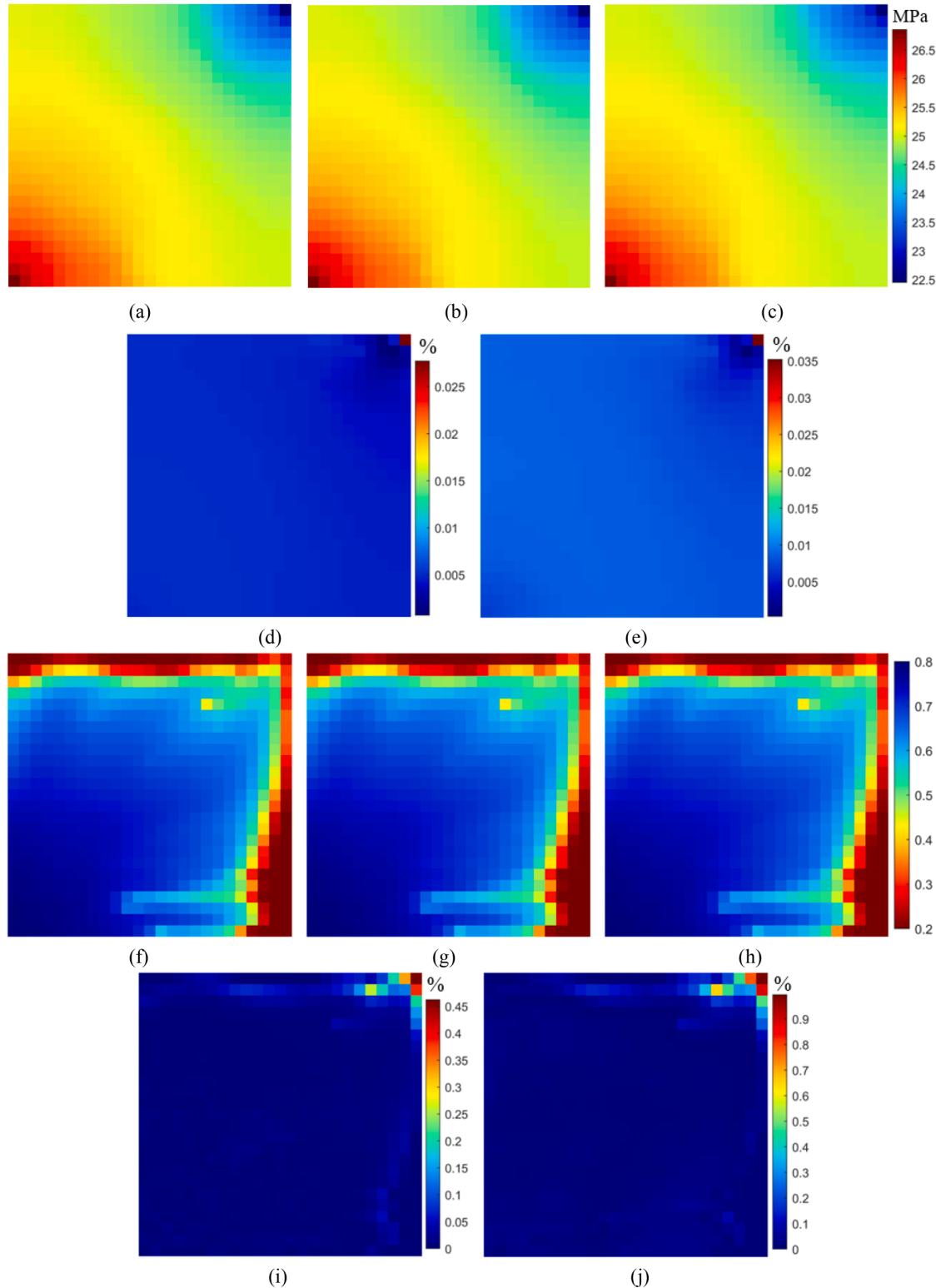


Fig. 5. The fields of reference pressure (a), E-PINN pressure (b), PICNN pressure (c), E-PINN pressure relative error (d), PICNN pressure relative error (e), reference S_w (f), E-PINN S_w (g), PICNN S_w (h), E-PINN saturation relative error (i), PICNN saturation relative error (j) at 500 day.

where $\|\cdot\|_2$ is the standard Euclidean norm; p^* and p_{ref} are the approximate and reference pressure.

Fig. 4 shows the changes in MSE and relative L_2 error over epochs of PICNN and E-PINN for all time steps. We can see that the MSE loss of E-PINN decreases faster and remains at lower level, which greatly reduces the training epochs of E-PINN. This is the principal reason for the

significant improvement in computational efficiency of E-PINN under the same conditions. It also can be seen that both PICNN and E-PINN have high accuracy (relative L_2 error of both are less than 1.0E-3), but there is an overall upward trend with the increase of time. This is because that the MSE of some time steps does not converge to 1.0E-18 before starting the training for the next time step, resulting in an

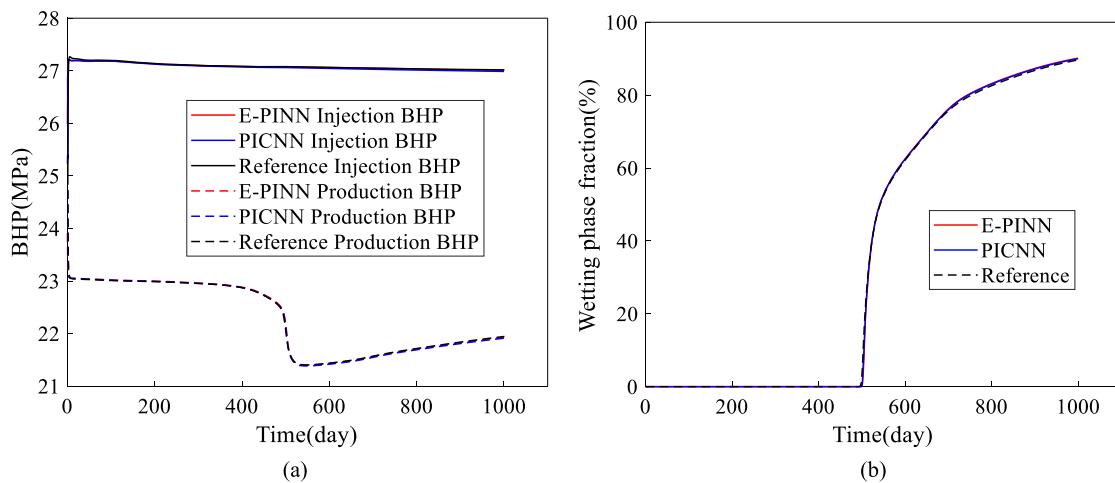


Fig. 6. Comparisons of the BHP of production well and injection well (a) and the wetting phase fraction of production well (b) calculated from different methods.

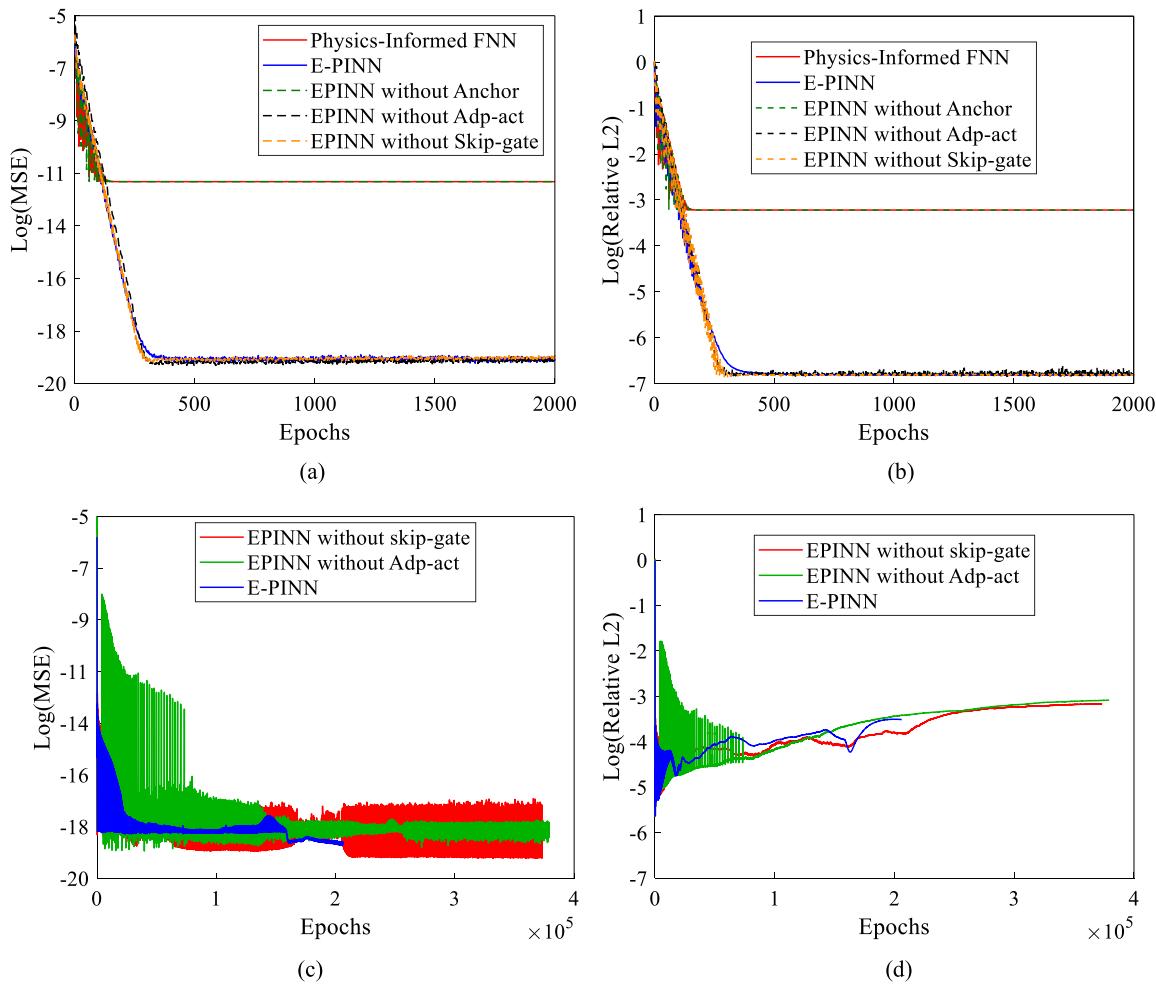


Fig. 7. The change of MSE and relative L_2 error with epochs of different methods for the first time step (a, b) and all time steps (c, d).

increase in relative L_2 error. Therefore, increasing the maximum epoch number can alleviate this problem, but it will reduce computational efficiency. Note that the steep rise of E-PINN in Fig. 4b is caused by the compression in x -direction (the abscissa is too short), and Fig. 7d shows the uncompressed version.

Fig. 5 shows the pressure fields and wetting phase saturation (S_w) fields of reference solution, E-PINN, and PICNN at 500 day and the

relative error profiles for pressure and saturation of E-PINN and PICNN. We can see the results of E-PINN and PICNN are very close to reference solutions, and their relative errors are all less than 1% (Fig. 5d, e, i and j). The bottom hole pressure curves of production well and injection well and the wetting phase fraction curve of production well, which are calculated from different methods, are plotted in Fig. 6, and great agreements between these curves can be seen. These comparisons verify

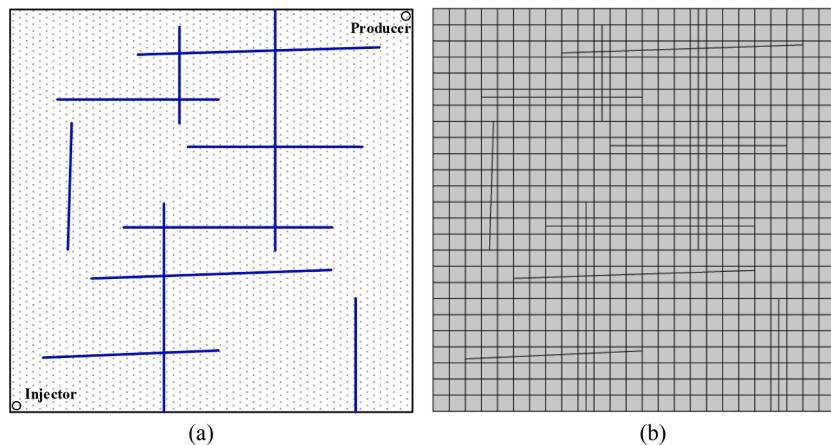


Fig. 8. The 2D fractured reservoir sketch (a) and its gridding (b).

Table 2
The summary of injection-production schemes.

Time (day)	Variable injection-production BHP		Variable injection-production rate	
	Injection BHP (MPa)	Production BHP (MPa)	Injection rate (m³/day)	Production rate (m³/day)
0–300	30.0	20.0	1.5	1.5
300–600	35.0	15.0	1.0	1.0
600–1000	40.0	10.0	2.0	2.0

the accuracy of E-PINN and PICNN in predicting 2D heterogeneous two-phase flow. Note that the wetting phase fraction indicates the ratio of the wetting phase which is produced in the production well compared to the volume of the total fluids produced.

In order to clarify whether the high efficiency of E-PINN is derived from FNN and illustrate the effect of three network structures (i.e., adjacency-location anchoring, adaptive activation function, and skip connection & gated updating), we apply physics-informed FNN (only with physics-informed loss function), E-PINN, and E-PINN without one structure to simulate the same displacement process. **Figs. 7a** and **b** show the changes in *MSE* and relative L_2 error over epochs of different methods for the first time step. We can see that the minimum *MSE* loss and minimum relative L_2 error of physics-informed FNN and E-PINN without adjacency-location anchoring are about 1.0E-11 and 1.0E-3, while the minimum *MSE* loss and minimum relative L_2 error of other methods are about 1.0E-19 and 1.0E-7. This means that physics-informed FNN and E-PINN without adjacency-location anchoring both fall short of required training tolerance (i.e., 1.0E-18), and their accuracy is low. **Fig. 7c** and **d** show the changes in *MSE* and relative L_2 error

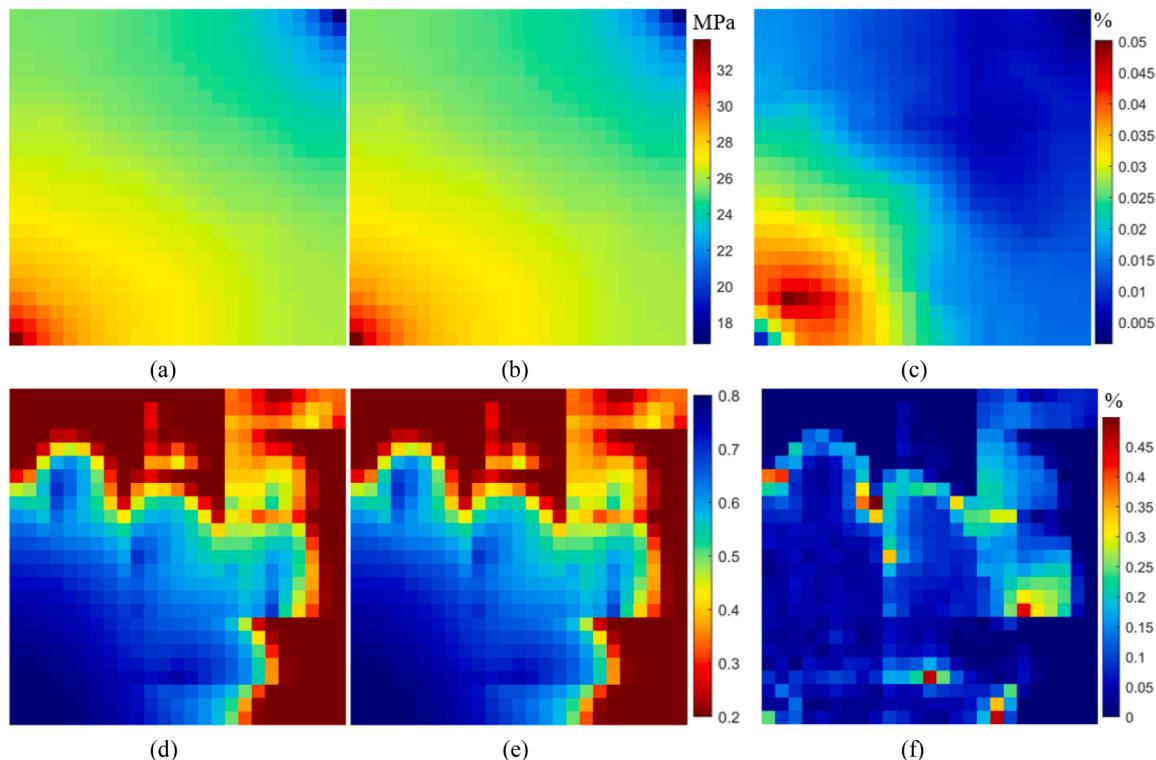


Fig. 9. The fields of reference pressure (a), E-PINN pressure (b), pressure relative error (c), reference S_w (d), E-PINN S_w (e), and saturation relative error (f) for variable injection-production BHP at 500 day.

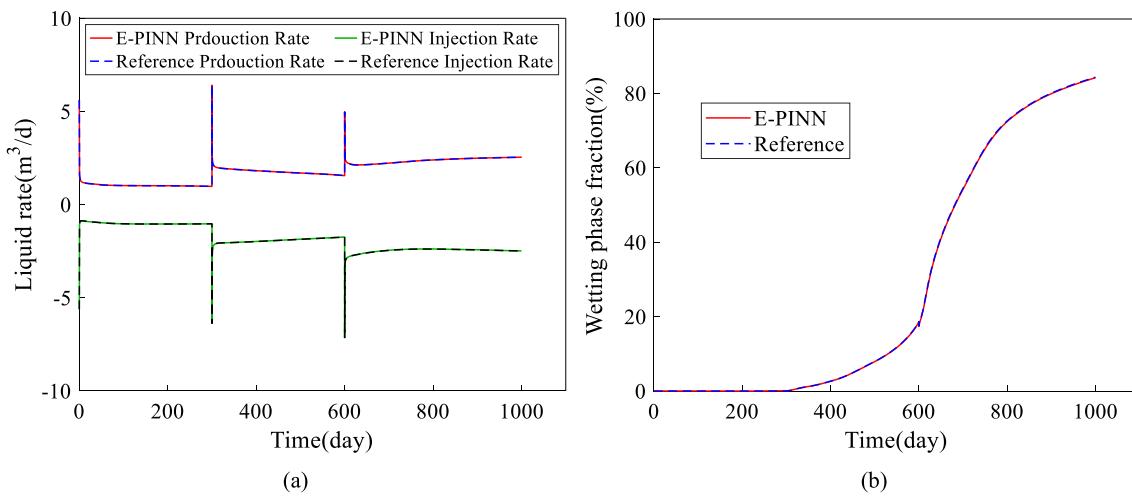


Fig. 10. Comparisons of the injection and production rates (a, negative sign indicates injection), and wetting phase fraction (b) calculated from different methods for the case with variable injection-production BHP.

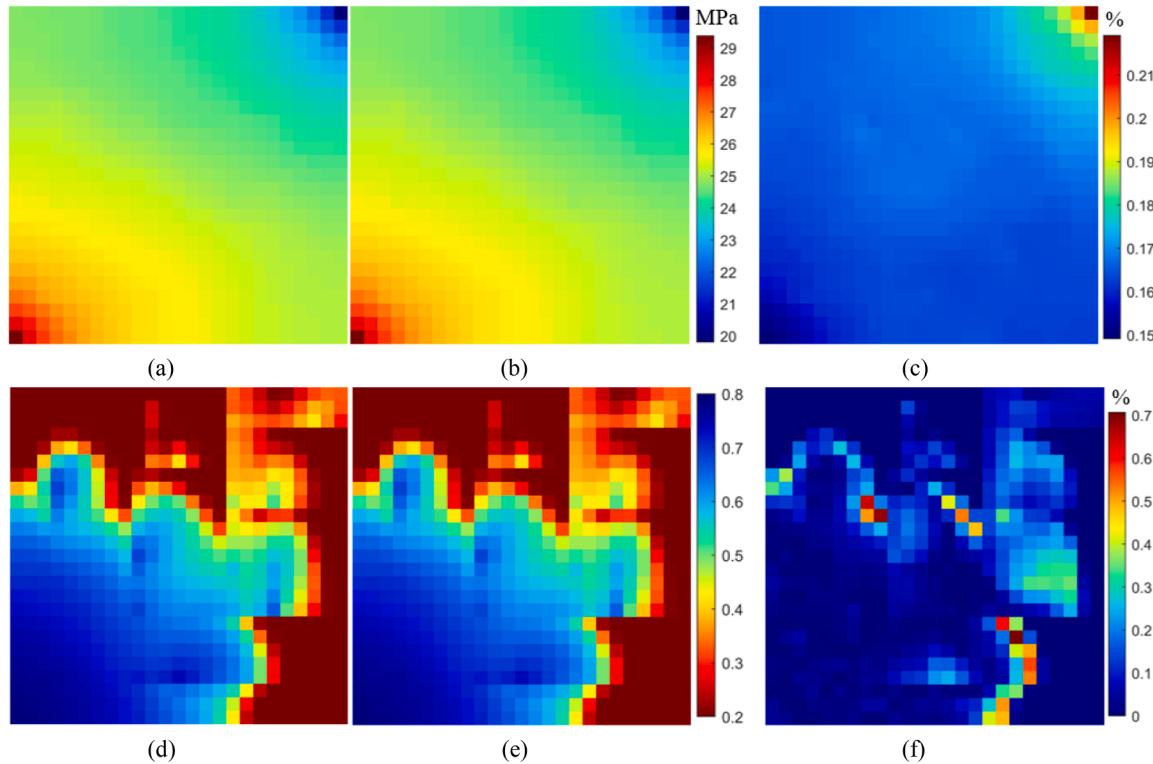


Fig. 11. The fields of reference pressure (a), E-PINN pressure (b), pressure relative error (c), reference S_w (d), E-PINN S_w (e), and saturation relative error (f) for variable injection-production rate at 500 day.

over epochs of E-PINN, E-PINN without adaptive activation function, and E-PINN without skip connection & gated updating for all time steps. Based on above results, it can be seen that the high efficiency of E-PINN is independent of FNN, but is related to the three network structures. Specifically, adjacency-location anchoring can improve the accuracy of training and other two network structures can improve the convergence of training.

3.1.2. Fractured reservoir

Fig. 8 displays a 2D fractured reservoir ($100 \text{ m} \times 100 \text{ m}$) including 11 fractures (blue line) and its computational grids (25×25). The locations of injection well and production well are also depicted in Fig. 8. The

initial wetting phase saturation and pressure of this reservoir are 0.2 and 25 MPa, respectively. The reservoir boundaries are closed. The matrix permeability, fracture permeability, fracture porosity, and fracture aperture are 10 mD, 30,000 mD, 1.0, and 0.01 m, respectively. The relative permeability curve shown in Fig. 3b is adopted to matrix and fracture. Other reservoir parameters are given in Table 1. We apply E-PINN to simulate the displacement process in this fractured reservoir with two different schemes (variable injection-production BHP and variable injection-production rate, Table 2) for 1000 days with $\Delta t = 0.01$ day. The learning rate, training tolerance, and maximum epoch number of each time step are 0.01, 1.0E-18 and 2000, respectively.

Fig. 9 presents the pressure fields and wetting phase saturation fields

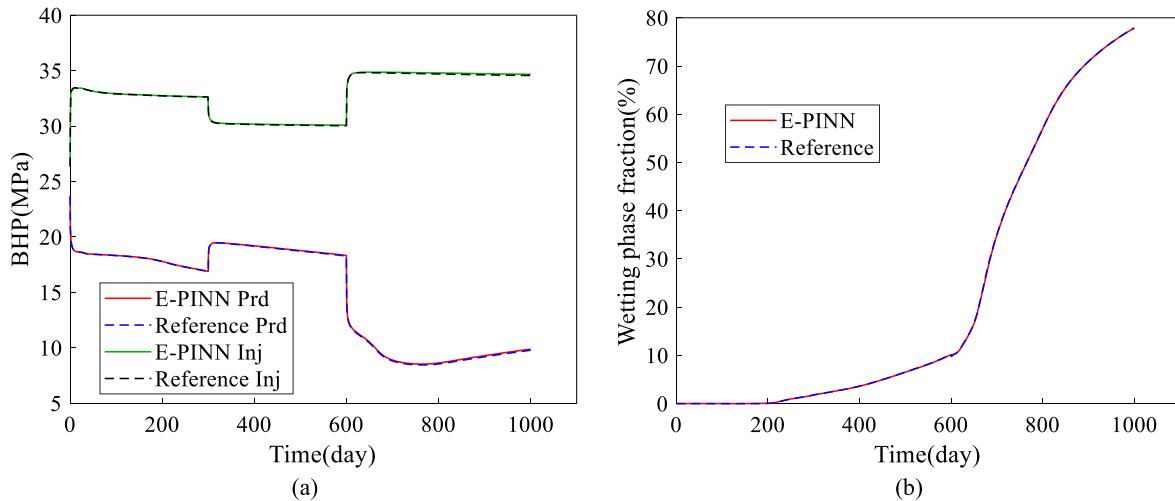


Fig. 12. Comparisons of the injection-production BHP (a), and wetting phase fraction (b) calculated from different methods for the case with variable injection-production rate.

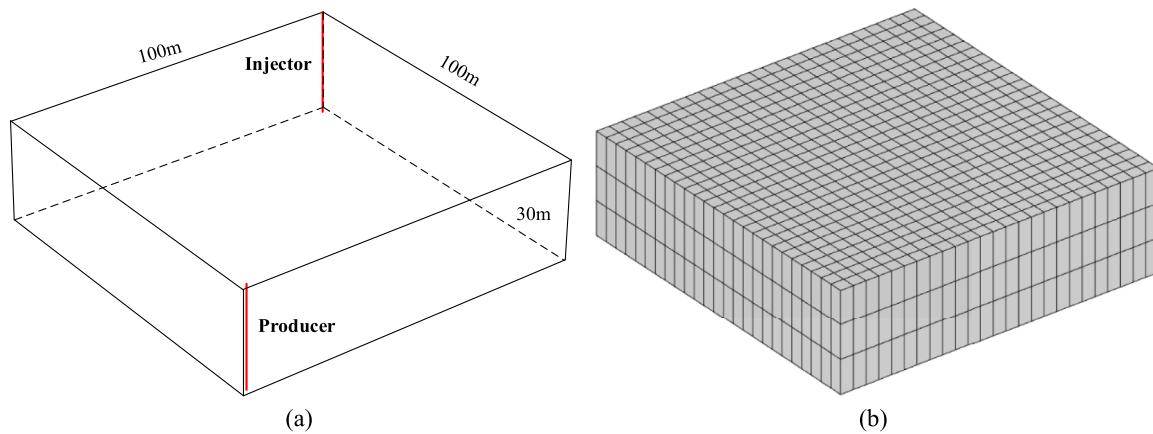


Fig. 13. The 3D heterogeneous reservoir sketch (a) and its gridding (b).

of reference solution and E-PINN for the case with variable injection-production BHP at 500 day and the relative error profiles of pressure and saturation between E-PINN and reference solution. We can see that E-PINN accurately predicts the evolution of pressure and wetting phase saturation in 2D fractured reservoir with extremely low relative errors (both less than 1%). Note that the prediction error of saturation field is higher than that of pressure field, and relatively obvious saturation

difference mainly occurs near the saturation front. It also can be seen that the saturation field is significantly affected by fractures, and E-PINN can effectively reproduce the non-uniform details of saturation field. Comparisons of the injection and production rates and wetting phase fraction calculated from different methods are plotted in Fig. 10, and we can see these results are very consistent with each other. Besides, we can also see that the injection and production rates at different stages change

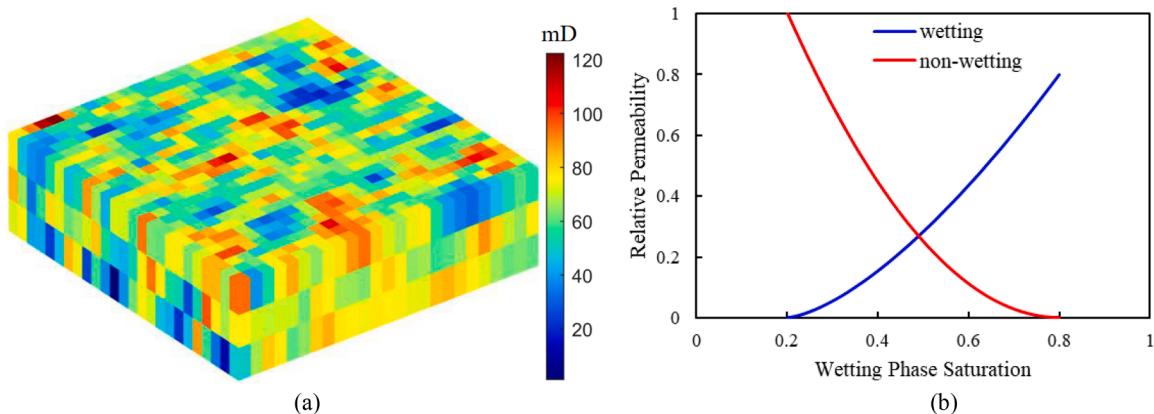


Fig. 14. The permeability distribution (a) and relative permeability curve (b) of 3D heterogeneous reservoir.

Table 3

The summary of 3D reservoir parameters.

Parameter	Value	Parameter	Value
matrix porosity	0.15	wetting phase compressibility, 1E-9	1E-9
matrix compressibility, 1/Pa	1E-10	non-wetting phase compressibility, 1/Pa	0.2
wetting phase density, kg/m ³	1500	irreducible wetting phase saturation	0.1
non-wetting phase density, kg/m ³	500	residual non-wetting phase saturation	0.0
wetting phase viscosity, mPa·s	2.5	wellbore radius, m	
non-wetting phase viscosity, mPa·s	1.0	well skin factor	

with the varying of injection-production BHP, and these features can be accurately predicted by E-PINN.

Fig. 11 shows the pressure and wetting phase saturation fields of reference solution and E-PINN for the case with variable injection-production rate at 500 day and the relative error profiles of pressure and saturation between E-PINN and reference solution. Fig. 12 presents the comparisons of the injection and production BHP and wetting phase fraction calculated from different methods for this case. Again, we can see that the results of E-PINN are highly consistent with reference solutions. These comparisons further demonstrate the adaptability of E-PINN to complex practical problems, such as variable injection-production schemes in fractured reservoirs.

3.2. 3D immiscible displacement examples

3.2.1. Heterogeneous reservoir

Fig. 13 shows a 3D heterogeneous reservoir (100m×100m×30 m) containing one production well and one injection well (red line) and its computational grids (25×25×3). The initial wetting phase saturation and pressure of this reservoir are 0.2 and 25 MPa, respectively. We have no flow at reservoir boundaries. Two vertical wells both penetrate three-layer vertical grids and operate with fixed BHP (i.e., production BHP: 20 MPa; injection BHP: 30 MPa). Fig. 14 presents the permeability distribution and relative permeability curve. Other reservoir parameters are given in Table 3.

The displacement process is simulated by using E-PINN and PICNN for 200 days with $\Delta t = 0.001$ day. The training tolerance is 1.0E-16 and the maximum epoch number for the first 100 time steps and all other time steps are 3000 and 100, respectively. Using a learning rate of 0.0002, the simulation times of PICNN and E-PINN are about 17291s and 2796s, respectively. It means that the computational efficiency of E-

PINN is more than 6 times faster than that of PICNN for predicting 3D heterogeneous two-phase flow. The changes in MSE and relative L_2 error over epochs of PICNN and E-PINN for all time steps are shown in Fig. 15. We can see E-PINN has obvious advantages in computational efficiency and training accuracy.

Fig. 16 shows the pressure fields and wetting phase saturation fields of reference solution, E-PINN, and PICNN at 50 day and the relative error profiles for pressure and saturation of E-PINN and PICNN. It can be seen that the prediction accuracy of EPINN for pressure field and saturation field is very high, and their relative errors are both less than 1% (Fig. 16d and i). However, the prediction accuracy of PICNN for these fields is lower, especially for saturation field, which has a relative error of more than 20% (Fig. 16j). The curves of production rate, injection rate and wetting phase fraction calculated from different methods are plotted in Fig. 17, and we can see the results of E-PINN are highly consistent with reference solutions. However, there are obvious differences between the results of PICNN and reference solutions, and the mean relative errors of production rate, injection rate and wetting phase fraction are 1.87%, 4.42%, 8.32%, respectively. These results show the superiority of E-PINN in predicting 3D heterogeneous two-phase flow.

3.2.2. Fractured reservoir

A 3D fractured reservoir (100 m × 50m × 20 m) with six fractures (blue surface) and two horizontal wells (red line), and its computational grids (20 × 10 × 5) are shown in Fig. 18. The initial wetting phase saturation and pressure of this reservoir are 0.2 and 25 MPa, respectively. We have no flow at reservoir boundaries. Two horizontal wells both penetrate ten-layer horizontal grids and operate with fixed BHP (i.e., production BHP: 23 MPa; injection BHP: 27 MPa). The matrix permeability, fracture permeability, fracture porosity, and fracture aperture are 50 mD, 50,000 mD, 1.0, and 0.01 m, respectively. The relative permeability curve shown in Fig. 14b is adopted for matrix and fracture. Other reservoir parameters are given in Table 3.

E-PINN is applied to simulate the displacement process in this reservoir with and without gravity for 200 days with $\Delta t = 0.001$ day. The learning rate and training tolerance are 0.0001 and 1.0E-16, the maximum epoch number for the first 200 time steps and all other time steps are 10,000 and 500, respectively. Fig. 19 presents the pressure fields and wetting phase saturation fields of reference solution and E-PINN for the case without gravity at 50 day and the relative error fields for pressure and saturation of E-PINN. The corresponding profiles of the case with gravity at 50 day are shown in Fig. 20. We can see that E-PINN accurately predicts the evolution of pressure and wetting phase saturation in 3D fractured reservoir with and without gravity. It also can be seen that gravity causes wetting phase to flow towards the reservoir

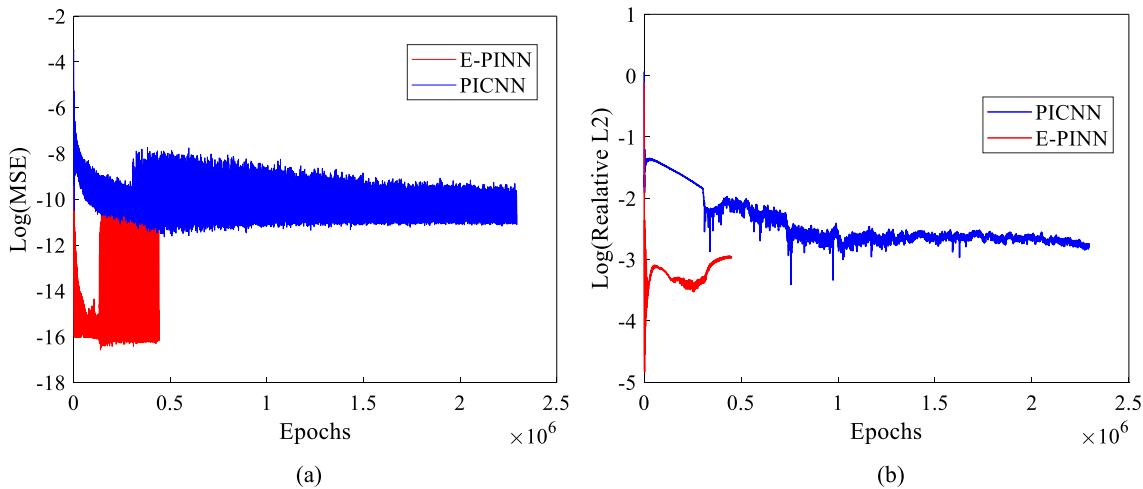


Fig. 15. The changes in MSE (a) and relative L_2 error (b) over all epochs of PICNN and E-PINN for 3D heterogeneous reservoir.

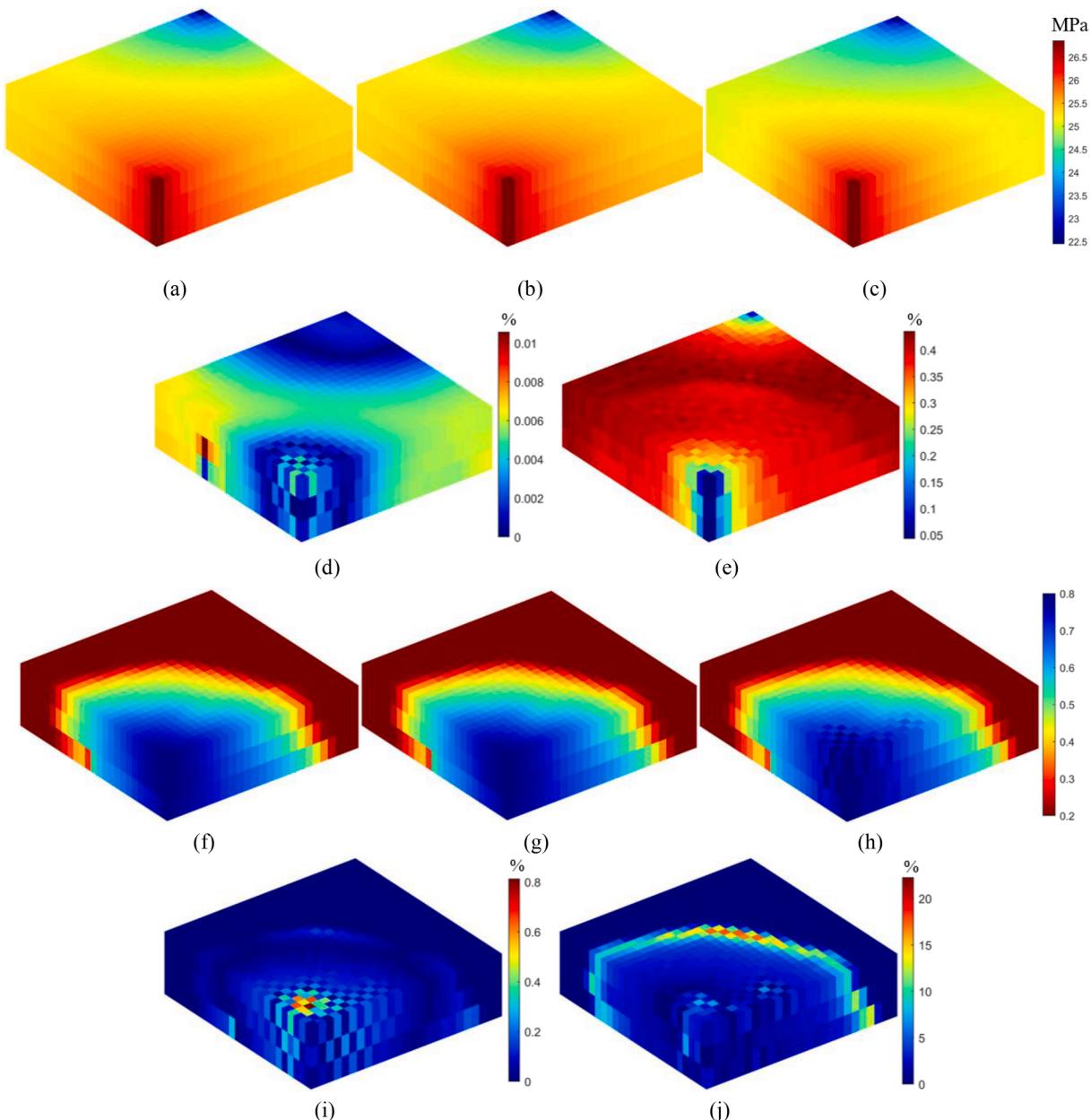


Fig. 16. The fields of reference pressure (a), E-PINN pressure (b), PICNN pressure (c), E-PINN pressure relative error (d), PICNN pressure relative error (e), reference S_w (f), E-PINN S_w (g), PICNN S_w (h), E-PINN saturation relative error (i), PICNN saturation relative error (j) at 50 day.

bottom since wetting phase is denser than non-wetting phase. Comparisons of the wetting phase fraction and cumulative non-wetting phase calculated from different methods for the cases with and without gravity are plotted in Fig. 21, and we can see the results of E-PINN are highly consistent with reference results. Moreover, we can also see that the production well located at the top layer has earlier wetting phase breakthrough time and underestimates cumulative non-wetting phase when gravity is not considered, and these features can be accurately predicted by E-PINN.

4. Conclusions

In this study, an enriched physics-informed neural network named E-PINN is developed for predicting two-phase flow in heterogeneous and fractured porous media without labeled data. Compared to existing PINNs, the improvements of E-PINN are mainly reflected in the following aspects: (1) The EDFM is adopted to explicitly represent

fractures, which are considered in PINNs for the first time, and the physics-informed loss function is constructed by means of the FVM so that the flux continuity between neighboring elements with different properties (e.g. matrix and fracture) can be defined rigorously, while it is non-trivial to realize via AD; (2) A novel physics-informed NN architecture adopting the adjacency-location anchoring, adaptive activation function, skip connection and gated updating is designed to enrich the pressure information and enhance the learning ability of NN; (3) The initial and boundary conditions are constrained through the hard approach, which encodes them into network design and has higher accuracy and efficiency compared with soft enforcement. We compare E-PINN and PICNN via 2D/3D heterogeneous two-phase flow, and demonstrate the superiority of E-PINN. Furthermore, E-PINN is used to simulate the immiscible displacements in 2D/3D fractured reservoirs and show its adaptability to complex practical problems, such as variable injection-production schemes and gravity segregation.

Although the results of E-PINN are in excellent agreement with

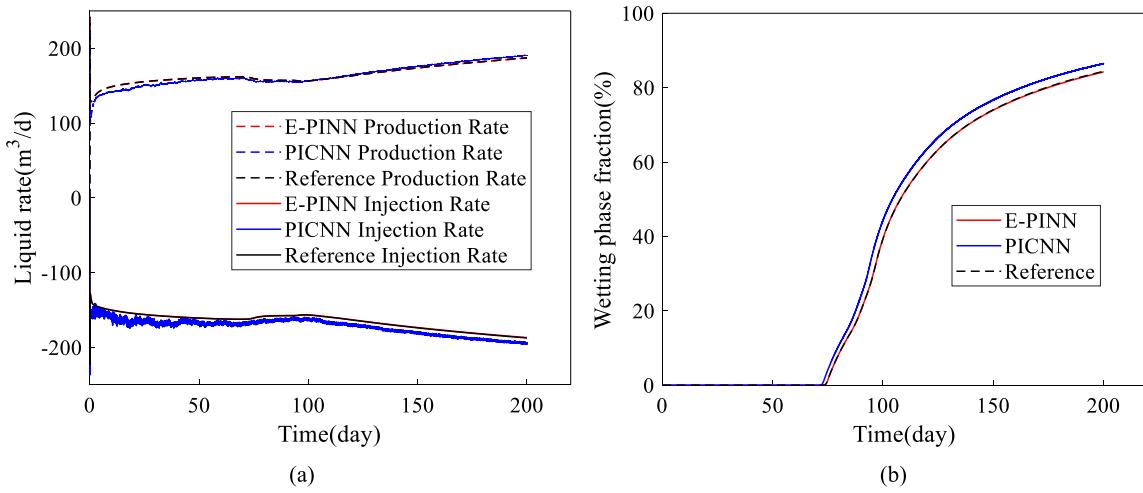


Fig. 17. Comparisons of the injection and production rate (a, negative sign indicates injection), and wetting phase fraction (b) calculated from different methods for 3D heterogeneous two-phase flow.

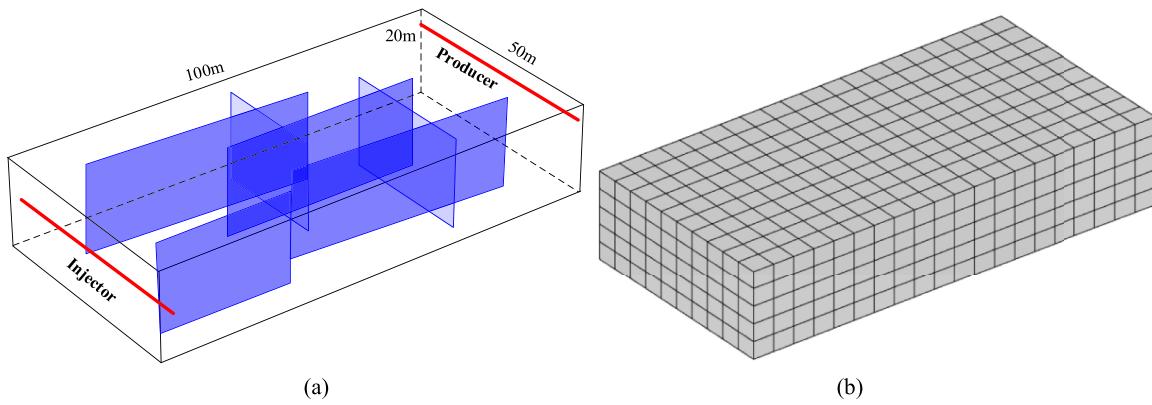


Fig. 18. The 3D fractured reservoir sketch (a) and its gridding (b).

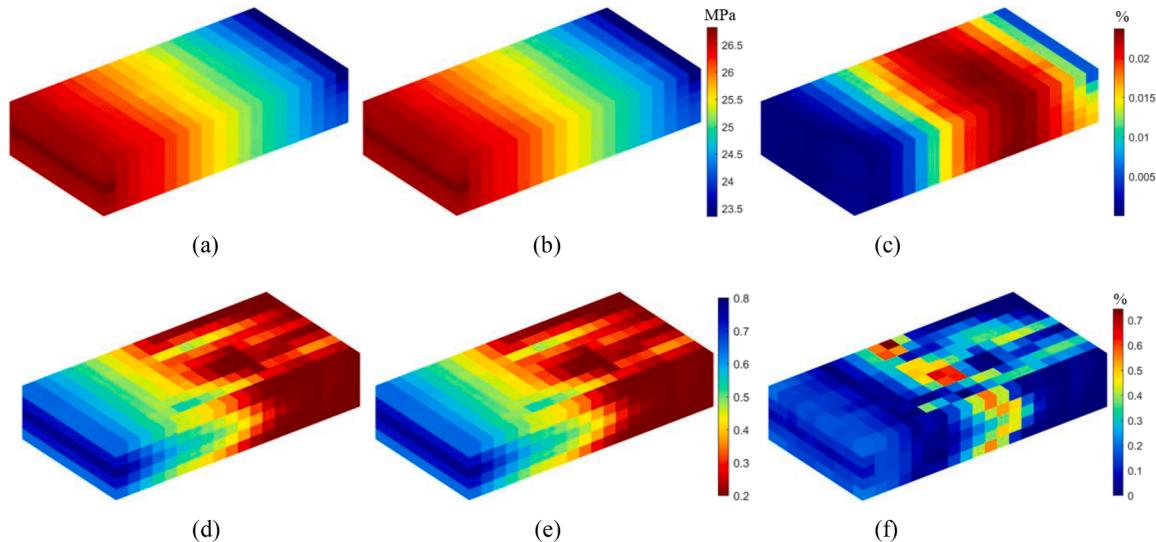


Fig. 19. The fields of reference pressure (a), E-PINN pressure (b), pressure relative error (c), reference S_w (d), E-PINN S_w (e), and saturation relative error (f) for the case without gravity at 50 day.

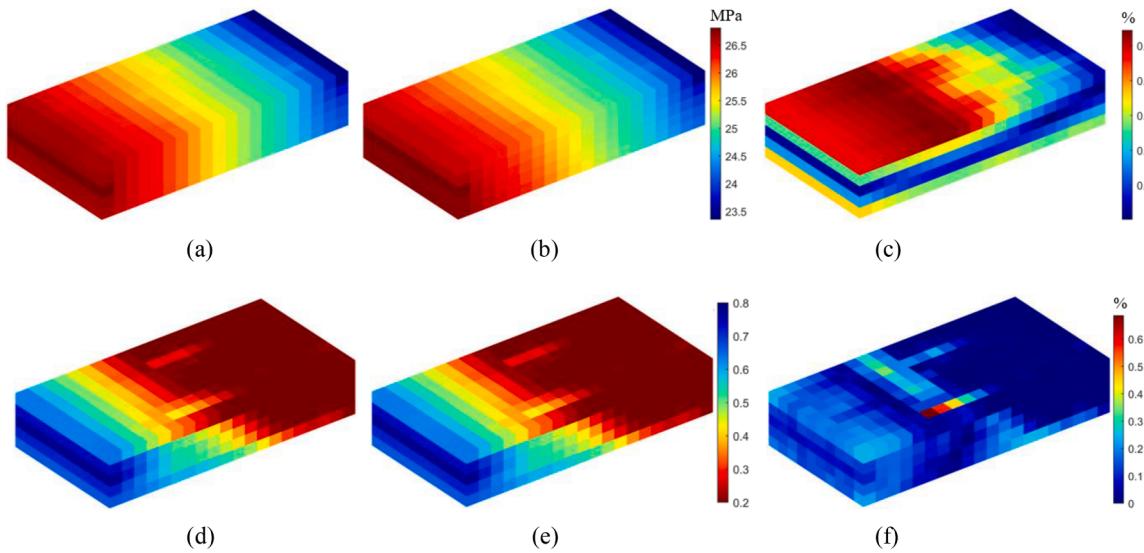


Fig. 20. The fields of reference pressure (a), E-PINN pressure (b), pressure relative error (c), reference S_w (d), E-PINN S_w (e), and saturation relative error (f) for the case with gravity at 50 day.

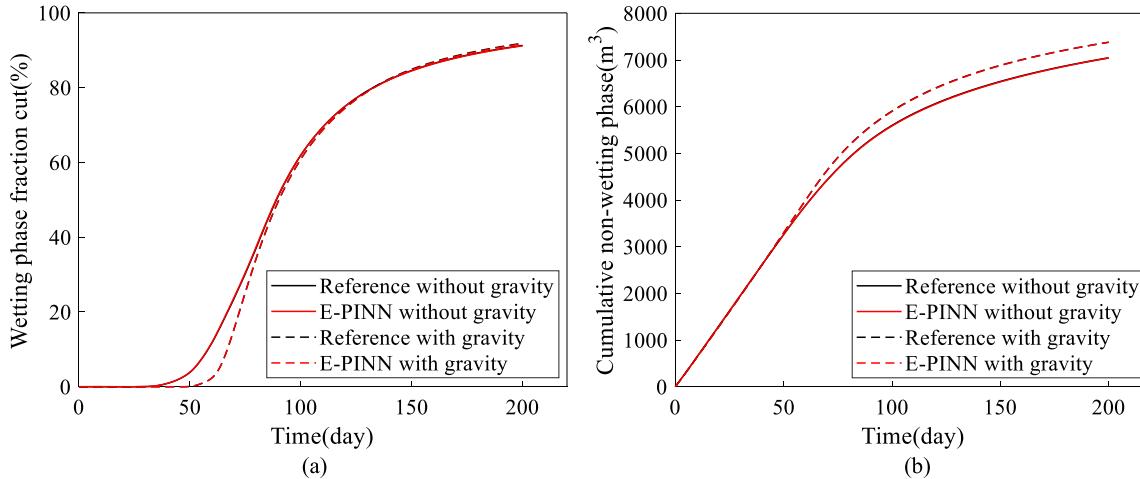


Fig. 21. Comparisons of the wetting phase fraction (a), and cumulative non-wetting phase (b) calculated from different methods for the 3D fractured reservoir with and without gravity.

reference results, it has been found that the training of E-PINN is time-consuming and it is currently uncompetitive in computational efficiency compared with state-of-the-art numerical simulators. Fortunately, E-PINN is currently in its early stages, and further developments could make it more competitive. Besides, the advantage of PINNs, as well as E-PINN, is their inherent ability incorporating measurement data to solve inverse problems. In the future, E-PINN will be applied for inverse problems of immiscible displacements, such as inverting the distribution of permeability. To this end, in addition to the NN for approximating the pressure field, an independent NN needs to be designed in the E-PINN to approximate the permeability field. Then, the outputs of these two NNs will be used to approximate the residual of governing equations. Finally, these two NNs will be trained simultaneously by minimizing the combined loss function, which consists of the residual of governing equations and the error relative to the measurement data, to obtain the permeability field. Furthermore, through the permeability inversion at different time steps, the dynamic permeability field varying with pressure can be obtained. The expected challenges of permeability inversion lie in: (1) the quantity and quality of measurement data; (2) the balance between physics-informed term and data

mismatch term in the loss function; (3) the convergence problem caused by the simultaneous training of two NNs. In addition, E-PINN also can be combined with the operator learning architecture (e.g., DeepONet) to generate surrogate models in a physically driven manner, for instance, a physics-informed DeepONet will be trained to map random initial conditions to the associated pressure and saturation solution within short time step. Then, the long-time prediction of pressure and saturation can be gained via iteratively evaluating the trained model using per prediction as the initial condition for the next step. Moreover, new neural network structures will be tried to further reduce the training time of E-PINN. Please note that the construction idea of E-PINN (i.e., constructing the physics-informed loss function through numerical discretization and extracting connection information between neighboring elements to construct a physics-informed NN architecture) is not only applicable to the two-phase flow problem mentioned in this study, but also to other physical problems that can be numerically discretized.

On the other hand, due to the numerical discretization of spatial and temporal derivatives and the use of IMPEs scheme, E-PINN has the following limitations: (1) it requires the grid generation, which results in additional computational costs; (2) the numerical discretization process

is troublesome, and an unreasonable scheme (e.g., too large time step) may introduce significant calculation error; (3) its *MSE* loss is not general, and the specific *MSE* loss needs to be established for various problems. Therefore, E-PINN can be improved in the future in the following aspects: (1) the meshless algorithm will be used for the numerical discretization to avoid grid generation; (2) the fully implicit scheme provided in [Appendix B](#) will be adopted to reduce the limitations of time step.

CRediT authorship contribution statement

Xia Yan: Writing – original draft, Validation, Supervision, Resources, Methodology, Investigation, Funding acquisition, Conceptualization. **Jingqi Lin:** Visualization, Validation, Software, Methodology, Data curation. **Sheng Wang:** Visualization, Validation, Software, Investigation, Data curation. **Zhao Zhang:** Writing – review & editing, Resources, Project administration, Formal analysis. **Piyang Liu:** Writing – review & editing, Resources, Formal analysis. **Shuyu Sun:** Writing – review & editing, Supervision, Formal analysis. **Jun Yao:** Writing – review & editing, Project administration, Funding acquisition. **Kai Zhang:** Writing – review & editing, Supervision, Resources, Funding acquisition.

Appendix A

This appendix provides the illustrate of NN architecture, optimization parameter and training time for each immiscible displacement example.

In this study, the E-PINN used for all numerical examples adopts the NN architecture described in [Section 2.2](#), and the only difference is the number of neurons, which is dependent with the number of grid elements. On the other hand, the PICNN used for 2D/3D heterogeneous examples adopts the NN architecture described in the reference ([Zhang et al., 2023](#)). The summary of NN architecture, optimization parameter and training time for each example is given in [Table A1](#). The *in_ch*, *out_ch*, *in_dim*, and *out_dim* indicate the number of input channels, output channels, input dimensions, and output dimensions, respectively. Note that the hyperparameters used for E-PINN and PICNN are manually optimized through the comparison of accuracy and efficiency for different settings. To further improve their performance, the automatic hyperparameter optimization ([Akiba et al., 2019](#)) can be used to determine the optimal setting for specific problems.

Table A1

The summary of NN architecture, optimization parameter and training time for each example.

Numerical example	Method	NN architecture	Optimizer and training parameters	Training time
2D heterogeneous	PICNN	CNN block1: <i>in_ch</i> =1, <i>out_ch</i> =25 CNN block2: <i>in_ch</i> =25, <i>out_ch</i> =50 Output: <i>in_dim</i> =1800, <i>out_dim</i> =625	Optimizer: Adam Learning rate: 0.01 Maximum epoch number for the first time step: 2000	10170s
	E-PINN	Input: <i>in_dim</i> =1, <i>out_dim</i> =625 Hidden: <i>in_dim</i> =625, <i>out_dim</i> =625 Output: <i>in_dim</i> =625, <i>out_dim</i> =1	Maximum epoch number for all others: 500 Training tolerance: 1.0E-18	878s
2D fractured variable BHP	E-PINN	Input: <i>in_dim</i> =1, <i>out_dim</i> =763 Hidden: <i>in_dim</i> =763, <i>out_dim</i> =763	Optimizer: Adam Learning rate: 0.01	31028s
2D fractured variable rate	E-PINN	Output: <i>in_dim</i> =763, <i>out_dim</i> =1	Maximum epoch number for all time steps: 2000 Training tolerance: 1.0E-18	26390s
3D heterogeneous	PICNN	CNN block1: <i>in_ch</i> =3, <i>out_ch</i> =100 CNN block2: <i>in_ch</i> =100, <i>out_ch</i> =200 Output: <i>in_dim</i> =7200, <i>out_dim</i> =1875	Optimizer: Adam Learning rate: 0.0002 Maximum epoch number for first 100 time steps: 3000	17291s
	E-PINN	Input: <i>in_dim</i> =1, <i>out_dim</i> =1875 Hidden: <i>in_dim</i> =1875, <i>out_dim</i> =1875 Output: <i>in_dim</i> =1875, <i>out_dim</i> =1	Maximum epoch number for all others: 100 Training tolerance: 1.0E-16	2796s
3D fractured with gravity	E-PINN	Input: <i>in_dim</i> =1, <i>out_dim</i> =1270 Hidden: <i>in_dim</i> =1270, <i>out_dim</i> =1270	Optimizer: Adam Learning rate: 0.0001	10612s
3D fractured without gravity	E-PINN	Output: <i>in_dim</i> =1270, <i>out_dim</i> =1	Maximum epoch number for first 200 time steps: 10,000 Maximum epoch number for all others: 500 Training tolerance: 1.0E-16	8905s

Appendix B

This appendix provides the brief implementation process of the fully implicit scheme.

The procedure of E-PINN-based pressure and saturation prediction and the structure of the E-PINN for the fully implicit scheme are illustrated in [Fig. B1](#). We use two independent physics-informed NNs to approximate pressure and saturation, and the input and output of the NNs are the values at time *t* and *t* + 1, respectively. The pressure and saturation of these inputs and outputs are used for constructing the physics-informed loss function to train NNs and update the pressure and saturation at time *t* + 1.

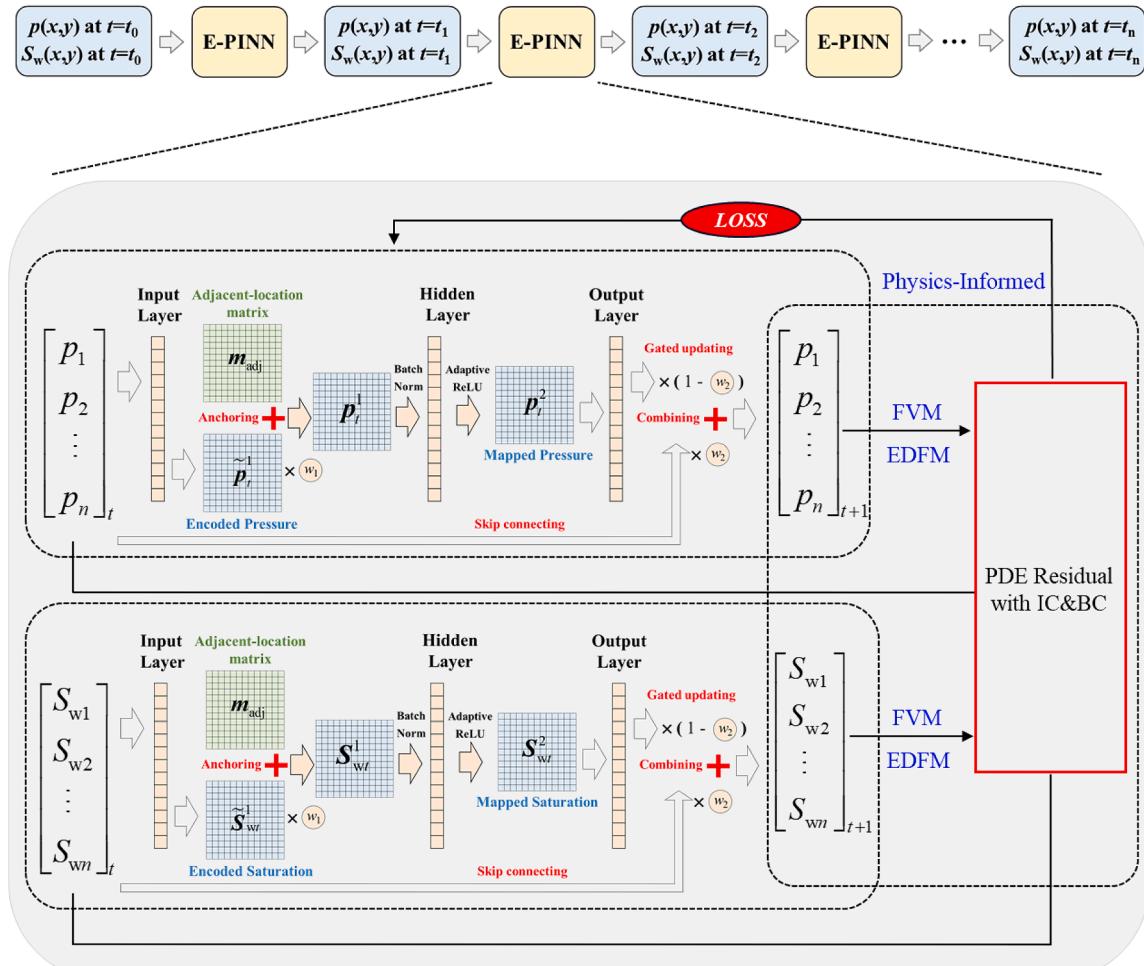


Fig. B1. The procedure of E-PINN-based pressure and saturation prediction.

In order to calculate the residual of governing equations for the fully implicit scheme, we first use the FVM to discretize Eq. (5), which is for non-wetting phase and wetting phase, as

$$\frac{\phi V_i}{\Delta t} (S_{ni}^{t+1} - S_{ni}^t) + \frac{S_n \phi (C_n + C_r) V_i}{\Delta t} (p_i^{t+1} - p_i^t) = \sum_{j \in G_i} (\lambda_{n,ij+1/2} T_{ij} (\Phi_{nj}^{t+1} - \Phi_{ni}^{t+1})) + q_n V_i \quad (\text{B.1})$$

$$\frac{\phi V_i}{\Delta t} (S_{wi}^{t+1} - S_{wi}^t) + \frac{S_w \phi (C_w + C_r) V_i}{\Delta t} (p_i^{t+1} - p_i^t) = \sum_{j \in G_i} (\lambda_{w,ij+1/2} T_{ij} (\Phi_{wj}^{t+1} - \Phi_{wi}^{t+1})) + q_w V_i \quad (\text{B.2})$$

Then, we substitute the outputs of two NNs into Eqs. (B.1) and B.2, considering $S_n=1-S_w$ and $\Phi_\beta=p - \rho_\beta g D$, to approximate the residual of governing equations, and the residual of i th element is

$$R_{ni} = \frac{\phi V_i}{\Delta t} (S_{ni}^* - S_{ni}^t) + \frac{S_n \phi (C_n + C_r) V_i}{\Delta t} (p_i^* - p_i^t) - \sum_{j \in G_i} (\lambda_{n,ij+1/2} T_{ij} (\Phi_{nj}^* - \Phi_{ni}^*)) - q_n V_i \quad (\text{B.3})$$

$$R_{wi} = \frac{\phi V_i}{\Delta t} (S_{wi}^* - S_{wi}^t) + \frac{S_w \phi (C_w + C_r) V_i}{\Delta t} (p_i^* - p_i^t) = \sum_{j \in G_i} (\lambda_{w,ij+1/2} T_{ij} (\Phi_{wj}^* - \Phi_{wi}^*)) + q_w V_i \quad (\text{B.4})$$

where p^* and S_w^* are the outputs of two NNs that represent p^{t+1} and S_w^{t+1} , respectively. As the same as the IMPES scheme, the IC is used as the inputs of NNs at first time step, and the BC is straightly imposed in Eqs. (B.3) and (B.4) by replacing p^* and S^* at boundary points with observed value, thus MSE is equal to MSE_r

$$MSE = MSE_r = \frac{1}{2N_e} \sum_{i=1}^{N_e} (R_{ni}^2 + R_{wi}^2) \quad (\text{B.5})$$

where N_e is the number of elements. Because the pressure and saturation fields between two consecutive time steps change gradually, the idea of transfer learning also can be utilized to improve training efficiency in the fully implicit scheme. In addition, training efficiency can be further

improved by normalizing the inputs and outputs of NN.

References

- Akiba, T., Sano, S., Yanase, T., et al., 2019. Optuna: a next-generation hyperparameter optimization framework. In: Proceedings of the Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining[C], pp. 2623–2631.
- Almajid, M.M., Abu-Al-Saud, M.O., 2022. Prediction of porous media fluid flow using physics informed neural networks. *J. Petrol. Sci. Eng.* 208, 109205.
- Altaheri, H., Muhammad, G., Alsulaiman, M., et al., 2023. Deep learning techniques for classification of electroencephalogram (EEG) motor imagery (MI) signals: a review. *Int. J. Neural Comput. Appl.*, 35, pp. 14681–14722.
- Bai, J., Rabczuk, T., Gupta, A., et al., 2023. A physics-informed neural network technique based on a modified loss function for computational 2D and 3D solid mechanics. *Comput. Mech.* 71 (3), 543–562.
- Benson, A.R., Gleich, D.F., Leskovec, J., 2016. Higher-order organization of complex networks. *Science* 353 (6295), 163–166.
- Cai, S., Wang, Z., Wang, S., et al., 2021a. Physics-informed neural networks for heat transfer problems. *J. Heat Transfer* 143 (6), 060801.
- Cai, S., Mao, Z., Wang, Z., et al., 2021b. Physics-informed neural networks (PINNs) for fluid mechanics: a review. *Acta Mech. Sin.* 37 (12), 1727–1738.
- Charbuty, B., Abdulazeez, A., 2021. Classification based on decision tree algorithm for machine learning. *J. Appl. Sci. Technol. Trends* 2 (01), 20–28.
- Chen, H., Kou, J., Sun, S., et al., 2019. Fully mass-conservative IMPES schemes for incompressible two-phase flow in porous media. *Comput. Methods Appl. Mech. Eng.* 350, 641–663.
- Coats, K.H., 2000. A note on IMPES and some IMPES-based simulation models. *SPE J.* 5 (03), 245–251.
- Cuomo, S., Di Cola, V.S., Giampaolo, F., et al., 2022. Scientific machine learning through physics-informed neural networks: where we are and what's next. *J. Sci. Comput.* 92 (3), 88.
- Daolun, L., Luhang, S., Wenshu, Z., et al., 2021. Physics-constrained deep learning for solving seepage equation. *J. Petrol. Sci. Eng.* 206, 109046.
- Datta L. A survey on activation functions and their relation with xavier and he normal initialization. arXiv preprint arXiv:200406632, 2020.
- Du, H., Zhao, Z., Cheng, H., et al., 2023. Modeling density-driven flow in porous media by physics-informed neural networks for CO₂ sequestration. *Comput. Geotech.* 159, 105433.
- Eymard, R., Gallouët, T., Herbin, R., 2000. Finite volume methods. In: *Handbook of Numerical Analysis*, 7, pp. 713–1018.
- Fuks, O., Tchelepi, H.A., 2020. Limitations of physics informed machine learning for nonlinear two-phase transport in porous media. *J. Mach. Learn. Model. Comput.* 1 (1).
- Gao, H., Sun, L., Wang, J.-X., 2021. PhyGeoNet: physics-informed geometry-adaptive convolutional neural networks for solving parameterized steady-state PDEs on irregular domain. *J. Computat. Phys.* 428, 110079.
- Goodfellow, I., Bengio, Y., Courville, A., 2016. Deep learning[M]. MIT press.
- Haghigat, E., Juanes, R., 2021. SciANN: a Keras/TensorFlow wrapper for scientific computations and physics-informed deep learning using artificial neural networks. *Comput. Methods Appl. Mech. Eng.* 373, 113552.
- He, Q., Barajas-Solano, D., Tartakovsky, G., et al., 2020. Physics-informed neural networks for multiphysics data assimilation with application to subsurface transport. *Adv. Water Resour.* 141, 103610.
- Hennigh, O., Narasimhan, S., Nabian, M.A., et al., 2021. NVIDIA SimNet™: an AI-accelerated multi-physics simulation framework[A]. In: *Proceedings of the International conference on computational science[C]*. Springer, pp. 447–461.
- Jagtap, A.D., Kawaguchi, K., Karniadakis, G.E., 2020. Adaptive activation functions accelerate convergence in deep and physics-informed neural networks. *J. Comput. Phys.* 404, 109136.
- Karimi-Fard, M., Durlofsky, L.J., Aziz, K., 2004. An efficient discrete-fracture model applicable for general-purpose reservoir simulators. *SPE J.* 9 (02), 227–236.
- Kashefi, A., Mukerji, T., 2023. Prediction of fluid flow in porous media by sparse observations and physics-informed PointNet. *Neural Netw.* 167, 80–91.
- Kingma D.P., Ba J. Adam: A method for stochastic optimization. arXiv preprint arXiv: 1412.6980, 2014.
- LeCun, Y., Bengio, Y., Hinton, G., 2015. Deep learning. *Nature* 521 (7553), 436–444.
- Lehmann, F., Fahs, M., Alhbail, A., et al., 2023. A mixed pressure-velocity formulation to model flow in heterogeneous porous media with physics-informed neural networks. *Adv. Water Resour.* 181, 104564.
- Li, Z., Liu, F., Yang, W., et al., 2021. A survey of convolutional neural networks: analysis, applications, and prospects. *IEEE trans. Neural Netw. Learn. Syst.*
- Li, J., Zhang, D., Wang, N., et al., 2022. Deep learning of two-phase flow in porous media via theory-guided neural networks. *SPE J.* 27 (02), 1176–1194.
- Liu, Y., Zheng, Y., Zhang, D., et al., 2022. Towards unsupervised deep graph structure learning. In: *Proceedings of the Proceedings of the ACM Web Conference 2022[C]*, pp. 1392–1403.
- Losapio, D., Scotti, A., 2023. Local Embedded Discrete Fracture Model (LEDFM). *Adv. Water Resour.* 171, 104361.
- Liu, L., Meng, X., Mao, Z., et al., 2021. DeepXDE: a deep learning library for solving differential equations. *SIAM Rev.* 63 (1), 208–228.
- Lv, S., Li, D., Zha, W., et al., 2024. Solving seepage equation using physics-informed residual network without labeled data. *Comput. Methods Appl. Mech. Eng.* 418, 116563.
- Martin, J.C., 1959. Simplified equations of flow in gas drive reservoirs and the theoretical foundation of multiphase pressure buildup analyses. *Trans. AIME* 216 (01), 321–323.
- Matsuo, Y., LeCun, Y., Sahani, M., et al., 2022. Deep learning, reinforcement learning, and world models. *Neural Netw.* 152, 267–275.
- Moortgat, J., Firoozabadi, A., 2013. Three-phase compositional modeling with capillarity in heterogeneous and fractured media. *SPE J.* 18 (06), 1150–1168.
- Otter, D.W., Medina, J.R., Kalita, J.K., 2020. A survey of the usages of deep learning for natural language processing. *IEEE trans. Neural Netw. Learn. Syst.* 32 (2), 604–624.
- Paszke, A., Gross, S., Massa, F., et al., 2019. Pytorch: an imperative style, high-performance deep learning library. *Adv. Neural Inform. Process. Syst.* 32.
- Peaceman, D.W., 1993. Representation of a horizontal well in numerical reservoir simulation. *SPE Adv. Technol. Series* 1 (01), 7–16.
- Pisner, D.A., Schnyer, D.M., 2020. Support vector machine[M]. *Machine Learning*. Elsevier, pp. 101–121.
- 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. *J. Computat. Phys.* 378, 686–707.
- Redondo, C., Rubio, G., Valero, E., 2018. On the efficiency of the IMPEs method for two phase flow problems in porous media. *J. Petrol. Sci. Eng.* 164, 427–436.
- Reichstein, M., Camps-Valls, G., Stevens, B., et al., 2019. Deep learning and process understanding for data-driven Earth system science. *Nature* 566 (7743), 195–204.
- Reyad, M., Sarhan, A.M., Arafa, M., 2023. A modified Adam algorithm for deep neural network optimization. *Neural Comput. Appl.* 1–18.
- Rodriguez-Torrado R., Ruiz P., Cueto-Felgueroso L., et al. Physics-informed attention-based neural network for solving non-linear partial differential equations. arXiv preprint arXiv:210507898, 2021.
- Saad, B., Saad, M., 2013. Study of full implicit petroleum engineering finite-volume scheme for compressible two-phase flow in porous media. *SIAM J. Numer. Anal.* 51 (1), 716–741.
- Sainath, T.N., Vinyals, O., Senior, A., et al., 2015. Convolutional, long short-term memory, fully connected deep neural networks. In: *Proceedings of the 2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)[C]*. Ieee, pp. 4580–4584.
- Santurkar, S., Tsipras, D., Ilyas, A., et al., 2018. How does batch normalization help optimization? *Adv. Neural Inform. Process. Syst.*, p. 31.
- Sun, A.Y., Yoon, H., Shih, C.-Y., et al., 2022. Applications of Physics-Informed Scientific Machine Learning in Subsurface science: A survey[M]. *Knowledge Guided Machine Learning*; Chapman and Hall/CRC, pp. 111–132.
- Tartakovsky, A.M., Marrero, C.O., Perdikaris, P., et al., 2020. Physics-informed deep neural networks for learning parameters and constitutive relationships in subsurface flow problems. *Water Resour. Res.* 56 (5), e2019WR026731.
- Voulodimos, A., Doulamis, N., Doulamis, A., et al., 2018. Deep learning for computer vision: a brief review. *Computat. Intell. Neurosci.* 2018.
- Wang, N., Zhang, D., Chang, H., et al., 2020. Deep learning of subsurface flow via theory-guided neural network. *J. Hydrol.* 584, 124700.
- Wang, P., Fan, E., Wang, P., 2021. Comparative analysis of image classification algorithms based on traditional machine learning and deep learning. *Patt. Recogn. Lett.* 141, 61–67.
- Wu, K., Xiu, D., 2020. Data-driven deep learning of partial differential equations in modal space. *J. Comput. Phys.* 408, 109307.
- Xu, J., Wei, H., Bao, H., 2023. Physics-informed neural networks for studying heat transfer in porous media. *Int. J. Heat Mass Transfer* 217, 124671.
- Yan, X., Huang, Z., Yao, J., et al., 2016. An efficient embedded discrete fracture model based on mimetic finite difference method. *J. Petrol. Sci. Eng.* 145, 11–21.
- Yan, X., Huang, Z., Yao, J., et al., 2018. An efficient numerical hybrid model for multiphase flow in deformable fractured-shale reservoirs. *SPE J.* 23 (04), 437, 1,412–411.
- Yan, X., Huang, Z., Yao, J., et al., 2019. Numerical simulation of hydro-mechanical coupling in fractured vuggy porous media using the equivalent continuum model and embedded discrete fracture mode. *Adv. Water Resour.* 126, 137–154.
- Yildiz, T., 2006. Assessment of total skin factor in perforated wells. *SPE Reservoir Eval. Eng.* 9 (01), 61–76.
- Yin, H., Zhang, G., Wu, Q., et al., 2023. A deep learning-based data-driven approach for predicting mining water inrush from coal seam floor using micro-seismic monitoring data. *IEEE Trans. Geosci. Remote Sens.*
- Yu, Y., Si, X., Hu, C., et al., 2019. A review of recurrent neural networks: LSTM cells and network architectures. *Neural Comput.* 31 (7), 1235–1270.
- Zhang, Z., Yan, X., Liu, P., et al., 2023. A physics-informed convolutional neural network for the simulation and prediction of two-phase Darcy flows in heterogeneous porous media. *J. Comput. Phys.* 477, 111919.
- Zhang, Z., 2022. A physics-informed deep convolutional neural network for simulating and predicting transient Darcy flows in heterogeneous reservoirs without labeled data. *J. Petrol. Sci. Eng.* 211, 110179.
- Zhou, J., Cui, G., Hu, S., et al., 2020. Graph neural networks: a review of methods and applications. *AI open* 1, 57–81.