Advanced Energy Systems and Intelligent Geoscience (AESIG)
https://www.mospbs.com/journal/aesig

• *Treatise* •

Boundary-integral Type Neural Network (BINN) for Flow Problems in Reservoirs

Yina Liu[1], Xiang Rao[2,3], Xupeng He[4], Qirun Fu[1], Hussein Hoteit[1]

[1]Physical Science and Engineering (PSE) Division, King Abdullah University of Science and Technology (KAUST), Thuwal, 23955-6900, Saudi Arabia

[2]School of Petroleum Engineering, Yangtze University, Wuhan, Hubei, China

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

[4]Saudi Aramco, Dhahran, Saudi Arabia.

[*]**Corresponding Author:** Xiang Rao

Received: 12 October 2025 **Accepted:** 7 December 2025

Abstract: This study pioneers the application of Boundary-Integral Neural Networks (BINNs) to subsurface flow simulation, addressing steady-state single-phase flow governed by Laplace-type equations in hydrocarbon reservoirs. BINNs synergistically integrate boundary integral equations (BIEs) with deep learning to overcome limitations of traditional mesh-based methods and Physics-Informed Neural Networks (PINNs). By leveraging Green's functions, BINNs transform partial differential equations into boundary-only formulations, where neural networks exclusively approximate boundary unknowns (pressure/flux), and interior solutions are reconstructed analytically. This approach achieves intrinsic dimensionality reduction, eliminating spatial domain discretization while ensuring mathematical consistency through exact boundary condition enforcement. Numerical validation demonstrates BINNs' computational advantages: In a rectangular reservoir, BINNs achieve sub-0.01% relative error (4.80×10−4 MPa) at interior points. For a complex trapezoidal reservoir with geometric singularities, BINNs attain Boundary Element Method (BEM)-comparable accuracy (max error ~0.115MPa) without specialized singularity treatments. Furthermore, in a challenging non-convex domain featuring recessed boundaries and internal production wells, BINNs effectively resolve the pressure singularities near wellbores, achieving high-fidelity reconstruction with a maximum relative error of 0.32%. The method's efficiency is evidenced by rapid convergence with minimal boundary sampling and moderate network sizes. BINNs provide a robust, meshless paradigm for reservoir-scale simulation, effectively handling irregular geometries while maintaining high fidelity and scalability, which uncovers its remarkable computational capabilities in the realm of single-phase flow, and offers an inaugural investigation and benchmark for its prospective extensive utilization in numerical reservoir simulations.

Keywords: Deep learning; Boundary integral equations (BIEs); Physics-informed neural networks

(PINNs); Boundary-Integral Neural Networks(BINNs); Meshless Reservoir Simulation; Dimensionality Reduction

1 Introduction

As the demand for increased accuracy and efficiency in reservoir flow simulation during hydrocarbon exploration and development continues to grow, traditional numerical methods such as the Finite Element Method (FEM) and Finite Volume Method (FVM) face significant challenges in solving the partial differential equations (PDEs) that describe subsurface flow processes. These approaches rely on complex spatial discretization meshes, which pose difficulties when handling intricate geometries or highly nonlinear problems, often resulting in substantial computational costs and mesh generation complexities, while also being prone to numerical dissipation issues (Hughes, 2003; LeVeque, 2002; Rao et al., 2024a).

Recently, deep learning frameworks exemplified by Physics-Informed Neural Networks (PINNs) have emerged, embedding physical laws as residuals within loss functions to enable meshless PDE solutions (Raissi et al., 2019; Rao et al., 2025a; Rao et al., 2025b;). Leveraging the function approximation capabilities of neural networks, PINNs directly learn the solution mappings from data and governing equations, theoretically circumventing the dependency on spatial discretization meshes characteristic of traditional methods (Liu et al., 2025).

However, PINNs encounter limitations when applied to reservoir-scale flow problems. Their global sampling strategies lead to exponential increases in training costs as the computational domain expands, and they often suffer from reduced accuracy near boundary singularities, such as wellbore regions (Krishnapriyan et al., 2021; Wang et al., 2021). To address these issues, Boundary Integral Neural Networks (BINNs) have been developed (Lin et al., 2023; Sun et al., 2023). BINNs integrate the mathematical core of Boundary Integral Equations (BIE) with deep learning architectures, transforming PDEs into boundary-only integral equations via Green's functions (Brebbia et al., 2012). Within this framework, neural networks are tasked solely with learning boundary unknowns—such as potential functions or their normal derivatives—while the solution within the domain is reconstructed through boundary integrals.

Compared to the global sampling-based solution paradigm of PINNs, BINNs demonstrate distinct advantages in efficiency and robustness: 1) BINNs achieve dimensionality reduction by solving BIEs, requiring only boundary sampling and thus significantly enhancing scalability for large-scale reservoir simulations; 2) they impose boundary conditions analytically, overcoming the approximation errors associated with PINN penalty methods and ensuring mathematical consistency of the solution; 3) the inherent Green's function kernel in boundary integral equations naturally adapts to fundamental solution characteristics, enabling accurate capture of solution behavior in regions with singularities, such as well points or crack tips, without specialized treatment.

In recent years, substantial progress has been made in the theoretical development and practical applications of BINNs. Lin et al. (2023) introduced the initial BINet framework, combining boundary integral equations with deep learning, utilizing potential theory to convert PDEs into boundary integral forms, and proving network convergence via neural tangent kernel theory. Zhang et al.

(2022) pioneered the integration of NURBS boundary parameterization and Telles' singularity treatment strategies into BINNs, validating their effectiveness on Laplace equation benchmarks. Sun et al. (2023) systematically demonstrated the dimensionality reduction benefits of boundary integral-based BINNs in transferring unknowns from the domain to the boundary, simplifying loss function design and improving computational efficiency. Concurrently, operator learning approaches driven by boundary integral equations have advanced, exemplified by Meng et al. (2024), who developed BI-DeepONet and BI-TDONet models capable of cross-domain generalization, enabling rapid inference of Laplace and acoustic scattering problems with a single training session. Dai et al. (2024) proposed the BI-GreenNet model, which extends the dimensionality reduction concept to Navier-Stokes equations through a boundary integral weight-sharing reconstruction mechanism. The application scope of BINNs has since expanded to elasticity and acoustics. Damiano and van Waterschoot (2025) employed the Physically Informed Boundary Integral Network (PIBI-Net), a variant of BINN, for acoustic field reconstruction, achieving high robustness with boundary-only data. Zhang et al. (2024) first applied BINNs to two-dimensional elasticity and piezoelectric problems, demonstrating that boundary discretization alone suffices to ensure the automatic satisfaction of governing equations.

Despite these advances, the application of BINNs to subsurface flow modeling remains unexplored. This study aims to investigate the foundational application of BINNs in two-dimensional single-phase reservoir flow problems, constructing a boundary integral-based neural network model for flow simulation, thereby providing a novel methodological approach for numerical modeling in hydrocarbon reservoirs.

2 Methodology

2.1 The governing equation

The seepage governing equations in subsurface porous media primarily consists of the fluid mass conservation equation and Darcy's law, which is derived from physical experiments. Assuming steady flow without source or sink terms, the mass conservation equation for the fluid is given by:

$$\nabla \cdot \mathbf{v} = 0 \tag{1}$$

where v is the seepage velocity of the fluid, and in two dimensions, $\mathbf{v} = \left(v_x, v_y\right)^T$.

Darcy's law is described as:

$$\mathbf{v} = -\mathrm{K}\nabla p \tag{2}$$

where $\mathrm{K} = \frac{k}{\mu}$ is the hydraulic conductivity (assumed to be a constant scalar, isotropic), p is the fluid pressure.

Substituting Eq. (2) into Eq. (1) yields an equation solely in terms of pressure p:

$$\nabla \cdot \left(\mathrm{K}\nabla p\left(\boldsymbol{x}\right)\right) = 0, \ = \boldsymbol{x} \in \Omega \subset \mathbb{R}^2 \tag{3}$$

In the absence of internal sources or sinks, as shown in Eq. (4), this seepage problem generally has both Dirichlet boundary conditions representing constant pressure boundaries and Neumann boundary conditions representing sealed or constant flow rates (Rao et al., 2022; Rao et al., 2024b).

$$\begin{gathered} \text{Dirichlet boundary condition:} \\ p\left(\boldsymbol{x}\right) = \overline{p}\left(\boldsymbol{x}\right), \ \boldsymbol{x} \in \Gamma_{DB} \\ \text{Neumann boundary condition:} \\ \mathbf{v}\left(\boldsymbol{x}\right) = -\mathrm{K}\nabla\overline{p}\left(\boldsymbol{x}\right), \ \boldsymbol{x} \in \Gamma_{NB} \end{gathered} \tag{4}$$

where $\overline{p}\left(\boldsymbol{x}\right)$ and $\nabla\overline{p}\left(\boldsymbol{x}\right)$ represents the known pressure and pressure gradient vector at point $\boldsymbol{x}$, and Γ_{DB} is the Dirichlet boundary, Γ_{NB} is the Neumann boundary.

2.2 Boundary integral equations (BIEs)

The fundamental solution of the two-dimensional Laplace equation is:

$$G(\boldsymbol{x},\boldsymbol{a})=\frac{1}{2\pi}\frac{1}{\ln(r)} \tag{5}$$

where $r=\|\boldsymbol{x}-\boldsymbol{a}$ is the Euclidean distance between the source point $\boldsymbol{a}$ and the field point $\boldsymbol{x}$.

After incorporating the fundamental solution, the governing equations can be transformed into boundary integral form based on Green's second equation:

$$\int_{\Omega}\left(p\nabla^2 G-G\nabla^2 p\right)d\Omega=\int_{\Gamma}\left(p\frac{\partial G}{\partial n}-G\frac{\partial p}{\partial n}\right)d\Gamma \tag{6}$$

Substituting $\nabla^2 p=0$ and $\nabla^2 G=-\delta(\boldsymbol{x}-\boldsymbol{a})$, and using the screening property of the Dirac δ-function, we obtain the point expression in the domain:

$$p(\boldsymbol{a})=\int_{\Gamma}G(\boldsymbol{x},\boldsymbol{a})\frac{\partial p}{\partial n_x}(\boldsymbol{x})d\Gamma(\boldsymbol{x})-\int_{\Gamma}p(\boldsymbol{x})\frac{\partial G}{\partial n_x}(\boldsymbol{x},\boldsymbol{a})d\Gamma(\boldsymbol{x}) \tag{7}$$

When the observation point ξ is located on the boundary Γ, integral singularity needs to be considered: for the $\int_{\Gamma}G(\boldsymbol{x},\boldsymbol{a})\frac{\partial p}{\partial n_x}(\boldsymbol{x})d\Gamma(\boldsymbol{x})$, G has weak singularity ($\ln(r)$) the integral exists; for the, $\int_{\Gamma}p(\boldsymbol{x})\frac{\partial G}{\partial n_x}(\boldsymbol{x},\boldsymbol{a})d\Gamma(\boldsymbol{x})$, $\frac{\partial G}{\partial n}=\frac{\partial G}{\partial r}\frac{\partial r}{\partial n}\propto\frac{1}{r}$ which has strong singularity, it is necessary to introduce the Cauchy Principal Value (CPV) integral with geometric coefficients $c(\boldsymbol{a})$, then we obtain (Rao et al., 2018; Rao et al., 2019):

$$c(\boldsymbol{a})p(\boldsymbol{a})+\int_{\Gamma}^{CPV}p(\boldsymbol{x})\frac{\partial G}{\partial n_x}(\boldsymbol{x},\boldsymbol{a})d\Gamma(\boldsymbol{x})=\int_{\Gamma}G(\boldsymbol{x},\boldsymbol{a})\frac{\partial p}{\partial n_x}(\boldsymbol{x})d\Gamma(\boldsymbol{x}) \tag{8}$$

where,

$$c(\boldsymbol{a})=\begin{cases}1 \text{ if } \boldsymbol{a}\in\Omega \text{ (interior point)}\\ \frac{1}{2} \text{ if } \boldsymbol{a}\in\Gamma \text{ and } \Gamma \text{ is smooth at } \boldsymbol{a}\\ \frac{\theta}{2\pi} \text{ if } \boldsymbol{a}\in\Gamma \text{ is a corner point in 2D}\end{cases} \tag{9}$$

2.3 Discretization of BIEs

We discretize the boundary integral equation into a system of linear equations using constant elements. The boundary Γ is divided into n boundary elements $\Gamma_j\,(j=1,2,\cdots,n)$ as shown in Fig. 1. Among these, $n1$ elements belong to the boundary Γ_{DB} and $n2$ boundary elements belong to the boundary Γ_{NB}. For constant elements, points are located at the midpoints of the boundary elements. The function values and normal derivative values on the boundary are assumed constant over each boundary element and equal to the values at the points.

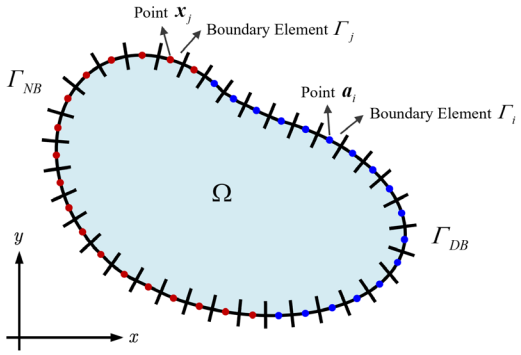


Fig. 1 Discretization of computational domain boundary

For point, Eq. (8) can be discretized as:

$$c(\boldsymbol{a}_i)p(\boldsymbol{a}_i)+\sum_{j=1}^{n}p(\boldsymbol{x}_j)\int_{\Gamma_j}\frac{\partial G}{\partial n_x}(\boldsymbol{x}_j,\boldsymbol{a}_i)d\Gamma(\boldsymbol{x})=\sum_{j=1}^{n}\frac{\partial p}{\partial n_x}(\boldsymbol{x}_j)\int_{\Gamma_j}G(\boldsymbol{x}_j,\boldsymbol{a}_i)d\Gamma(\boldsymbol{x}) \tag{10}$$

where, $\boldsymbol{a}_i$ denotes the point of boundary elements, Γ_i and $\boldsymbol{x}_j$ represents an arbitrary point within boundary element Γ_j. The integrals over the boundary elements in the equation can be computed a high-order Gaussian quadrature formula.

Define:

$$\begin{aligned}\hat{H}_{ij}&=\int_{\Gamma_j}\frac{\partial G}{\partial n_x}(\boldsymbol{x}_j,\boldsymbol{a}_i)d\Gamma(\boldsymbol{x})\\ M_{ij}&=\int_{\Gamma_j}G(\boldsymbol{x}_j,\boldsymbol{a}_i)d\Gamma(\boldsymbol{x})\end{aligned} \tag{11}$$

and let $\frac{\partial p}{\partial n_x}(\boldsymbol{x}_j)=q(\boldsymbol{x}_j)$, then Eq. (10) can be written as:

$$c(\boldsymbol{a}_i)\,p(\boldsymbol{a}_i)+\sum_{j=1}^{n}p(\boldsymbol{x}_j)\hat{H}_{ij}=\sum_{j=1}^{n}q(\boldsymbol{x}_j)M_{ij} \tag{12}$$

where,

$$H_{ij}=\hat{H}_{ij}+\frac{1}{2}\delta_{ij} \tag{13}$$

where, δ_{ij} is Kronecker function, that is:

$$\delta_{ij}=\begin{cases}0, & i\neq j\\ 1, & i=j\end{cases} \tag{14}$$

So, Eq. (12) can be further simplified to:

$$\sum_{j=1}^{n}p(\boldsymbol{x}_j)H_{ij}=\sum_{j=1}^{n}q(\boldsymbol{x}_j)M_{ij} \tag{15}$$

Since n_1 function values and n_2 normal derivative values have been prescribed as boundary conditions, solving Eq. (15) therefore yields all unknown quantities on the boundary: specifically, n_2 function values and n_1 normal derivative values.

2.4 Neural networks and loss function of the BINNs

BINNs adopt a fully connected multi-layer perceptron (MLP) architecture, as illustrated in Fig. 2, to approximate the unknown boundary variables in boundary integral equations (BIEs). The network consists of an input layer, hidden layers, and an output layer, with each containing L hidden layers, and an output layer, with each hidden layer containing $n1$ neurons. The input layer receives the midpoint coordinates of boundary element $\boldsymbol{x}=(x,y)$, and the output layer produces the boundary unknowns $\hat{\boldsymbol{u}}$ containing pressure $\hat{p}$ or normal derivative $\hat{q}$ based on the boundary type.

Mathematically, the network mapping is defined as: for a given input , the output of the -th hidden layer is:

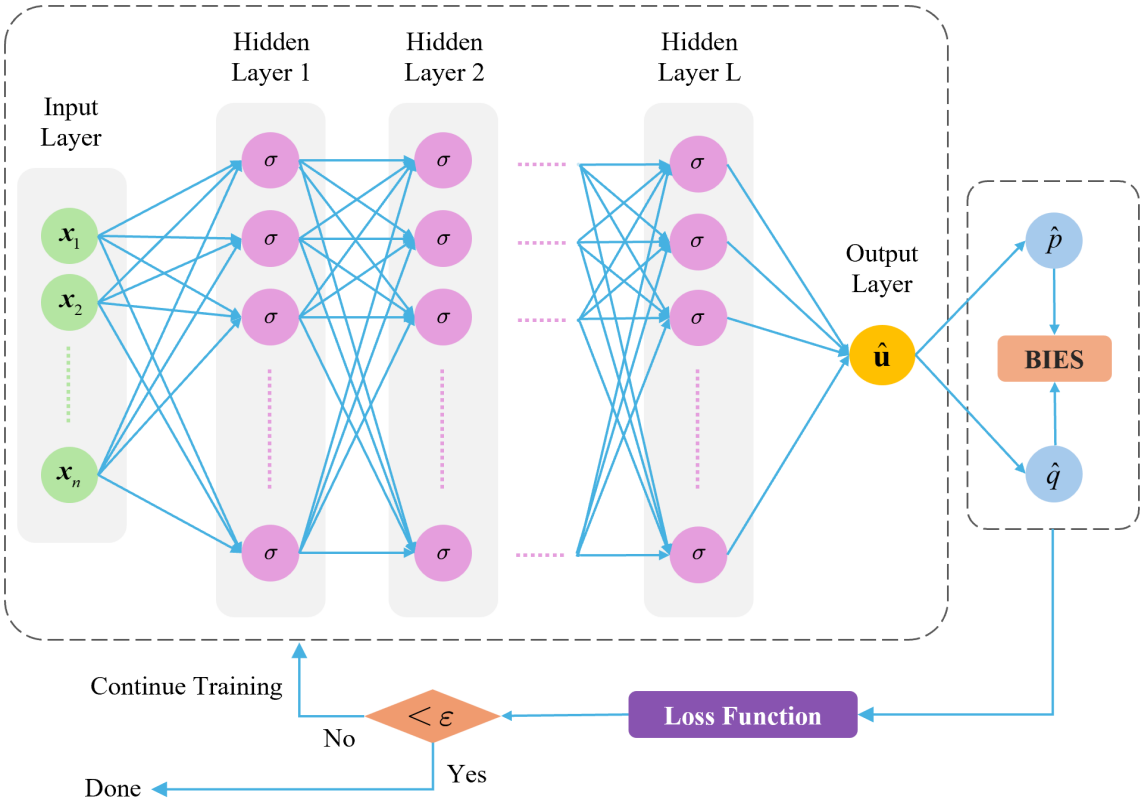


Fig. 2 The framework of the boundary integrated neural networks (BINNs).

$$z_l=\sigma_l(W_l z_{l-1}+b_l,),\ l=1,2,\cdots,L \tag{16}$$

where W_l and b_l are the weight matrix and bias vector of the -th layer, respectively; σ_l denotes the activation function (e.g., ReLU, tanh). The output layer maps the hidden layer features to the boundary variables:

$$\begin{cases} p(x;\Theta)=W_{\text{out}}z_L+b_{\text{out}}, \textit{for Neumann boundary} \\ q(x;\Theta)=W_{\text{out}}z_L+b_{\text{out}}, \textit{for Dirichlet boundary} \end{cases} \tag{17}$$

where $\Theta=\{W_l,b_l\}_{l=1}^{L}$ represents the network parameters.

For collocation points x^n on the boundary, the residual term is defined as the mean squared error (MSE) as follows:

$$\mathcal{L}_{BIE}=\frac{1}{n}\sum_{j=1}^{n}\left[LE\left(x_j,p\left(x_j;\Theta\right)\right)-RE\left(x_j,q\left(x_j;\Theta\right)\right)\right]^2 \tag{18}$$

where, $LE\left(x_j,p\left(x_j;\Theta\right)\right)$ and $RE\left(x_j,q\left(x_j;\Theta\right)\right)$ refers to the left term and right term of Eq. (15).

Once the boundary variables $\hat{p}$ and $\hat{q}$ are obtained via the trained network, the pressure at any interior point is reconstructed using the boundary integral formula:

$$p\left(x_{internal}\right)=\sum_{j=1}^{n}\left(q\left(x_j\right)M_{ij}-p\left(x_j\right)H_{ij}\right) \tag{19}$$

This formula ensures that the internal solution automatically satisfies the governing PDE, leveraging the integral representation from potential theory.

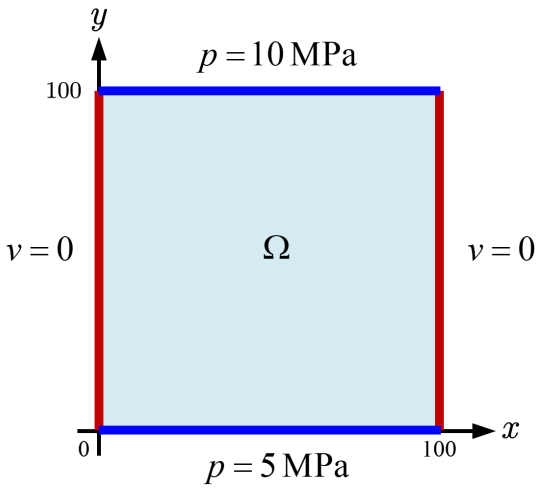
3 Results and Discussions

This section validates the computational efficacy of BINNs through two examples of increasing complexity: a rectangular reservoir and a trapezoidal reservoir featuring geometric singularities. The numerical experiments quantitatively assess three critical aspects: (1) accuracy of boundary-to-interior field reconstruction, (2) robustness in handling irregular boundaries, and (3) convergence efficiency.

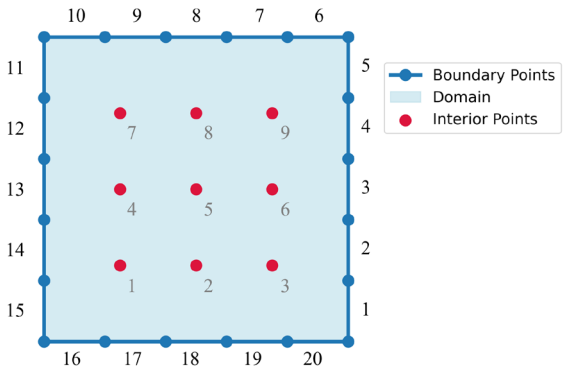
3.1 Example 1

Consider a square oil reservoir region $\Omega=[0,100]\times[0,100]$ with a side length of 100m. This example investigates the steady-state single-phase seepage flow problem without source or sink terms. The governing equation for this system is the Laplace equation, as follows:

$$\nabla\cdot\left(\mathrm{K}\nabla p\left(x\right)\right)=0 \tag{20}$$



(a) Computational domain geometry and boundary conditions



(b) Boundary element and interior verification point distribution

Fig. 3 The reservoir domain, boundary elements, and interior points in Example 1.

where, $K=10^{-6}$ $m^2/(MPa\cdot s)$.

The boundary conditions for this computational domain are defined as follows: the top and bottom boundaries are specified as constant pressure boundaries (Dirichlet conditions), while the left and right boundaries are treated as no-flow boundaries (Neumann conditions), as illustrated in Fig. 3(a).

$$\begin{cases} p(x,0)=5\ \text{MPa}, & 0\le x\le 100 \\ p(x,100)=10\ \text{MPa}, & 0\le x\le 100 \\ v(0,y)=\dfrac{\partial p}{\partial n}(0,y)=0, & 0\le y\le 100 \\ v(100,y)=\dfrac{\partial p}{\partial n}(100,y)=0, & 0\le y\le 100 \end{cases} \tag{21}$$

As shown in Fig. 3(b), each of the four edges of the computational domain is discretized into 5 constant boundary elements, resulting in a total of 20 boundary elements. Additionally, 9 interior verification points were selected to validate the accuracy of the numerical solution.

The analytical solution is obtained using the separation of variables method, providing a benchmark for accuracy assessment:

$$p(x,y)=5+\frac{y}{20}\ \text{MPa}, \quad x,y\in[0,100] \tag{22}$$

To ensure numerical stability and that the gradients of all parameters during BINN training are of similar magnitude, the coordinates are normalized by scaling them by a factor of 100. This transformation defines new dimensionless coordinates: $x'=x/100$, $y'=y/100$. This coordinate transformation is purely mathematical and does not alter the underlying physical relationships represented by the governing equation. The analytical solution for pressure under the normalized coordinate system (x',y') becomes the scaled version of the original solution:

$$p(x',y')=5+5y'\ \text{MPa}, \quad x,y\in[0,1] \tag{23}$$

For the BINN implementation, the network structure uses a matrix $\boldsymbol{x}$ composed of the midpoint coordinates of boundary elements as input. The architecture consists of 2 hidden layers, each configured with 30 neurons, employing the hyperbolic tangent (Tanh) function as the activation function. The network outputs a matrix containing the unknown function values $\hat{p}$ (pressure) and $\hat{q}$ (pressure gradient) at the midpoints of boundary elements. The network training process utilizes the Adam optimizer with a learning rate of 10^{-3} and conducts over 10,000 iterations. Table 1 shows the parameter configuration of Example 1. For pressure calculations at interior points, boundary integral formulas are employed to achieve the complete solution for the entire seepage field.

Fig. 4 demonstrates the convergence behavior of the boundary integral equation (BIE) loss function throughout the training process. The convergence pattern exhibits two distinct phases: during the initial iterations (epochs 0 to 2000), the loss decreases rapidly, indicating that the network parameters quickly adapt to satisfy the boundary integral equation constraints. Subsequently (epochs 2000 to 10000), the loss stabilizes and asymptotically approaches zero, demonstrating that the BINN has successfully fitted both the boundary conditions and governing equations, with the numerical solution accuracy reaching a convergent state.

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

Model	Number of hidden layers	Number of neurons per hidden layer	Total number of parameters	Activation Function	Optimizer (Adams)
BINN	2	30	1051	Tanh	10000 iterations, learning rate 10^{-3}

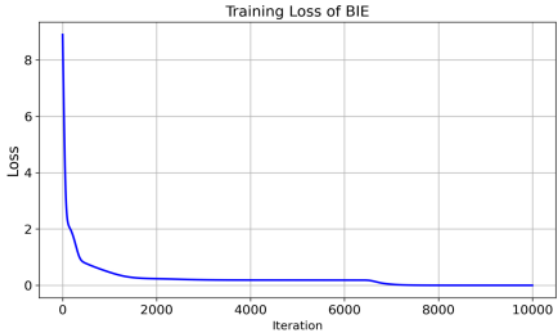


Fig. 4 Loss function of BIE in Example 1.

Table 2 presents a comprehensive error comparison between BINN numerical solutions and analytical solutions at the 9 interior verification points. The error distribution across all verification points shows consistent performance, with no significant bias toward any particular region of the domain. The maximum absolute error of 5.59×10^{-3} MPa and the minimum error of 4.80×10^{-4} MPa which represents less than 0.01% relative error, indicate robust solution quality throughout the computational domain. These results validate the effectiveness of the BINN approach for solving steady-state seepage flow problems in regular geometric domains. The rapid convergence and high accuracy achieved in this example establish a baseline for BINN performance on problems with simple geometries and well-defined boundary conditions. The computational efficiency demonstrated here, combined with the meshless nature of the boundary integral approach, suggests significant potential for extending this methodology to more complex reservoir geometries and boundary condition configurations.

Table 2 Errors of the BINN solution vs. analytical solutions at interior points

Index	Coordinates	BINN solution	Analytical solution	Errors
1	(25, 25)	6.2482	6.25	1.82×10^{-3}
2	(50, 25)	6.2451	6.25	4.95×10^{-3}
3	(75, 25)	6.2469	6.25	3.15×10^{-3}
4	(25, 50)	7.5023	7.50	2.25×10^{-3}
5	(50, 50)	7.5005	7.50	4.80×10^{-3}
6	(75, 50)	7.5014	7.50	1.40×10^{-3}
7	(25, 75)	8.7556	8.75	5.59×10^{-3}
8	(50, 75)	8.7523	8.75	2.26×10^{-3}
9	(75, 75)	8.7551	8.75	5.09×10^{-3}

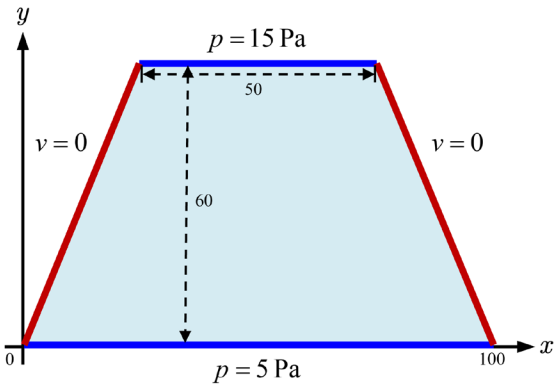
3.2 Example 2

This example addresses the oil reservoir seepage flow problem with complex geometric boundaries, investigating the steady-state single-phase seepage characteristics within a trapezoidal oil reservoir region. The computational domain represents a more realistic reservoir geometry with an upper base width of 50m, a lower base width of 100m, and a height of 60m (Fig. 5(a)). The system satisfies the Laplace equation governing conditions without internal source or sink terms.

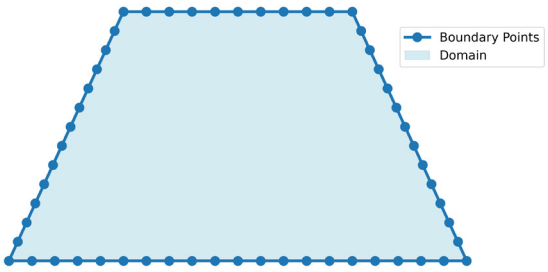
The boundary conditions are configured to simulate realistic reservoir flow patterns: the upper and lower boundaries are specified as constant pressure boundaries (Dirichlet conditions), while the left and right boundaries are treated as no-flow boundaries (Neumann conditions). This configuration creates a pressure-driven flow from the top boundary (15 MPa) to the bottom boundary (5 MPa), with the trapezoidal geometry introducing complex flow patterns due to the converging lateral boundaries (Eq. (24)).

$$\begin{cases} p(x,0)=5\ \text{MPa}, & 0\le x\le 100 \\ p(x,60)=15\ \text{MPa}, & 25\le x\le 75 \\ v(x,y)=\dfrac{\partial p}{\partial n}(0,y)=0, & 0\le x\le 25,\ 0\le y\le 60 \\ v(x,y)=\dfrac{\partial p}{\partial n}(0,y)=0, & 75\le x\le 100,\ 0\le y\le 60 \end{cases} \tag{24}$$

The computational domain boundary discretization employs 56 constant boundary elements (each of 5m length) distributed across all four edges of the trapezoid (Fig. 5(b)).



(a) Computational domain geometry and boundary conditions



(b) Boundary element and interior verification point distribution

Fig. 5 The reservoir domain, boundary elements, and interior points in Example 2

To address the solution challenges introduced by the complex trapezoidal geometry, the BINN model adopts a significantly enhanced architecture compared to Example 1. The network utilizes a 4-layer hidden layer structure, with each layer configured with 60 neurons, maintaining the hyperbolic tangent (Tanh) function as the activation function. This architectural enhancement, increasing the total parameter count to 11,221, is necessary to capture the more complex flow patterns and pressure gradients introduced by the irregular boundary geometry. The increased network complexity reflects the fundamental challenge of representing non-uniform flow fields where the pressure gradients vary significantly across the domain due to geometric effects. The additional layers and neurons provide the network with sufficient representational capacity to approximate the intricate boundary integral relationships inherent in problems with irregular geometries. Table 3 comprehensively summarizes the model parameters.

The iterative convergence process of the BIE loss function, shown in Fig. 6, demonstrates robust training characteristics despite the increased complexity of the problem. During the initial training phase, the loss value rapidly decreases from approximately 12, and by 900 iterations, the loss approaches zero and stabilizes. This convergence pattern indicates that the enhanced network architecture successfully accommodates the additional complexity while maintaining training stability and achieving satisfactory equation fitting. The convergence rate, while slightly slower than Example 1 due to the increased problem complexity, still demonstrates efficient training characteristics. The stabilization of the loss function near zero confirms that the BINN model has effectively learned to satisfy the boundary integral equation constraints throughout the irregular domain.

Fig. 7 presents a comprehensive comparison of pressure field distributions between the BINN

Table 3 Parameters of the neural networks and optimizers in Example 2

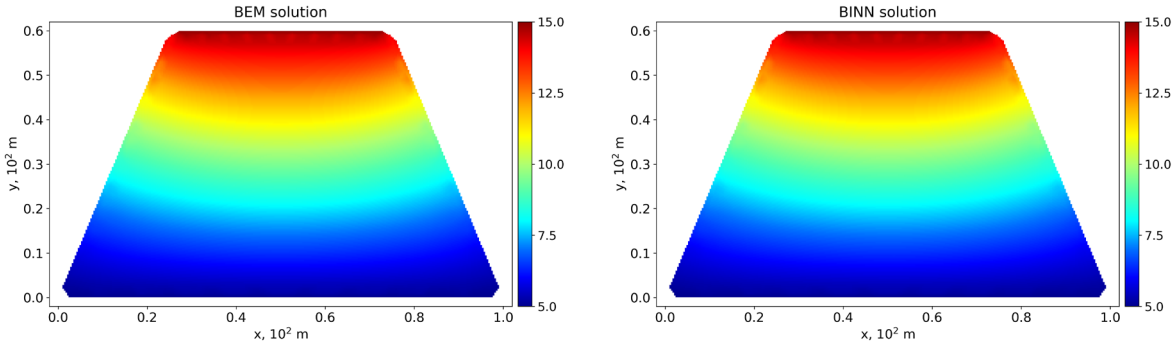
Model	Number of hidden layers	Number of neurons per hidden layer	Total number of parameters	Activation Function	Optimizer (Adams)
BINN	4	60	11221	Tanh	10000 iterations, learning rate 10^{-3}

Fig. 6 Loss function of BIE in Example 2.

solution and the Boundary Element Method (BEM) reference solution. Both computational approaches demonstrate remarkable consistency in capturing global pressure distribution patterns and local gradient variations. The pressure field characteristics presented by the BEM solution (Fig. 7(a)) align closely with those predicted by the BINN solution (Fig. 7(b)), demonstrating that BINN can effectively handle the mathematical complexities introduced by non-rectangular domain geometries.

The absolute error distribution analysis, presented in Fig. 8, provides detailed insight into the spatial accuracy characteristics of the BINN solution. The maximum error in the entire computational domain is ~0.115 MPa, with an average error of ~0.015 MPa and a root mean square error of ~0.022 MPa. These error levels represent excellent accuracy for practical engineering applications, particularly considering the geometric complexity of the problem. A notable observation is the localized error concentration phenomenon at both ends of the trapezoidal upper boundary, corresponding to the geometric discontinuity points where the inclined side walls meet the horizontal upper boundary. This error concentration is physically reasonable and mathematically expected, as these locations represent singular points where boundary conditions transition abruptly, creating challenges for any numerical method. Importantly, the errors in the smooth interior regions of the domain approach zero, demonstrating that BINN maintains high solution accuracy away from geometric singularities. The spatial error distribution shows that 97% of the domain exhibits errors below 0.1 MPa, with significant errors confined to less than 5% of the domain area near the geometric discontinuities. The error statistics indicate that BINN achieves numerical accuracy comparable to the well-established BEM under equivalent boundary conditions.

The comprehensive comparison demonstrates that BINN achieves numerical accuracy comparable to the BEM through neural network-based global continuous mapping of boundary integral kernels. The method successfully handles irregular geometries while maintaining computational efficiency and solution accuracy. The ability to



(a) BEM solution (b) BINN solution

Fig. 7 Comparison of pressure field distributions between BINN solution and BEM solution.

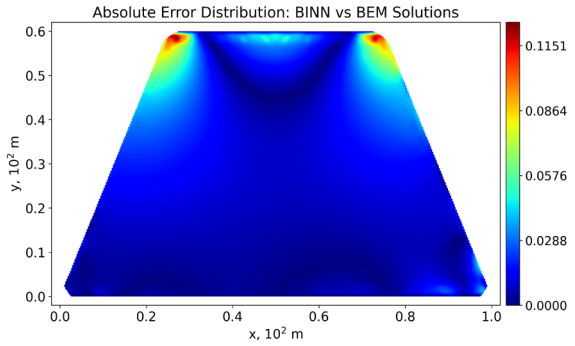


Fig. 8 Absolute error distribution of BINN solution and BEM solution (unit: MPa)

represent complex flow patterns without traditional mesh generation represents a significant advantage for reservoir engineering applications where complex geological boundaries are common.

3.3 Example 3

This example investigates seepage with recessed boundaries and multi-well coupling. The computational domain is an irregular polygon with an upward-protruding notch on its bottom edge (Fig. 9), distinct from the regular rectangular domains in prior examples. A permeability K=500 mD is adopted, with all four boundaries set to a constant pressure of 10 MPa (Eq. (25)). Crucially, one production well is placed at $^{[120,80]}$m with intensities of -500 m^2/s, simulating asymmetric pressure fields in realistic reservoirs.

$$p(x,y)=15\ \text{MPa},\ (x,y)\in\Gamma \tag{25}$$

The computational domain boundary discretization employs 128 constant boundary elements (each of 5m length) distributed across all four edges of the irregular computational domain (Fig. 5(b)).

To tackle the computational challenges posed by the irregular non-convex domain and the singular pressure field induced by the production well, the BINN model inherits the enhanced architecture established in Example 2. The specific parameter

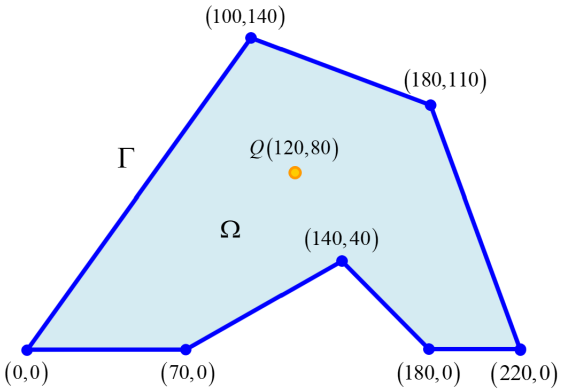


Fig. 9 Computational domain geometry and boundary conditions.

configuration is identical to that listed in Table 3, ensuring a consistent benchmark for evaluating the method's adaptability to different geometric topologies.

Fig. 10 illustrates the convergence behavior of the BIE loss function for Example 3. Similar to previous examples, the training process exhibits a two-stage characteristic: a precipitous drop in the loss value within the first 200 iterations, followed by a stable asymptotic approach to zero. Despite the introduction of the source term (production well) and the "notch" boundary singularity, the loss function shows no significant oscillations, stabilizing at a magnitude of $O(10^{-3})$. This confirms that the BINN effectively learns the mapping relationship between the boundary integrals and the internal source singularity, demonstrating

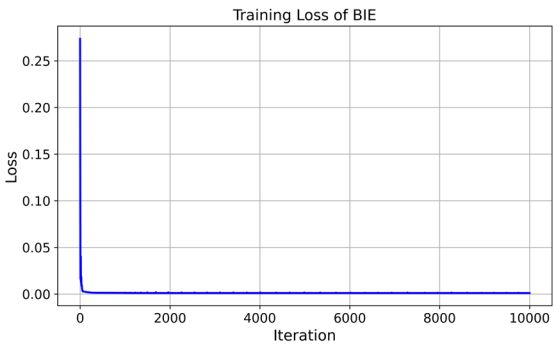
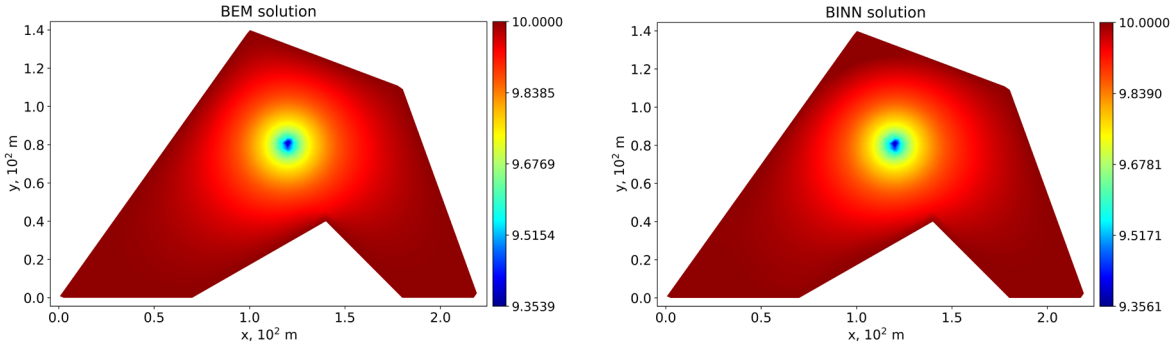


Fig. 10 Loss function of BIE in Example 2.



(a) BEM solution (b) BINN solution

Fig. 11 Comparison of pressure field distributions between BINN solution and BEM solution.

robust numerical stability even for domains with recessed boundaries.

Fig. 11 compares the pressure field distributions obtained from the BINN solution and the BEM reference solution. Visually, the BINN solution (Fig. 11(b)) accurately reconstructs the radial flow pattern and the significant pressure drop cone surrounding the production well, showing high consistency with the BEM result (Fig. 11(a)). The method also correctly captures the pressure gradients near the complex re-entrant corners of the domain.

The absolute error distribution analysis, detailed in Fig. 12, provides a rigorous quantitative assessment of the BINN solution's spatial fidelity. The method achieves a global Average Error of approximately 0.0020 MPa and an RMSE of ~0.0045 MPa. The Maximum Error recorded is ~0.0323 MPa. When normalized against the characteristic reservoir pressure of 10 MPa, this maximum deviation corresponds to a relative error of merely 0.32%, while the domain-averaged relative error is negligible at 0.02%. As observed in the error map, the peak errors are strictly localized near the polygon vertices, while the error in the vast majority of the bulk domain remains negligible.

solution and BEM solution (unit: MPa)

This example validates the potential of BINNs in handling seepage problems with recessed boundaries and multi-well coupling. By integrating the fundamental solution for point sources into the boundary integral formulation, BINN successfully solved the pressure field in a domain with internal singularities. The results, characterized by low error metrics and accurate capture of the pressure funnel, demonstrate that the proposed method possesses significant potential for simulating realistic reservoirs with complex well patterns and irregular geological boundaries.

4 Conclusion

This work establishes Boundary-Integral Neural Networks (BINNs) as a transformative methodology for reservoir flow simulation,

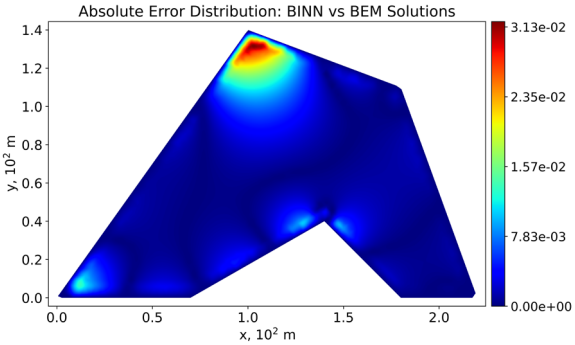


Fig. 12— Absolute error distribution of BINN

demonstrating their efficacy in solving two-dimensional single-phase seepage problems. BINNs fundamentally address key challenges of conventional approaches by unifying BIEs with deep learning: they eliminate spatial discretization meshes required by FEM/FVM and overcome the computational inefficiency of PINNs in large domains through dimensionality reduction. Neural networks learn only boundary variables, while interior solutions are analytically reconstructed via Green's functions, ensuring exact satisfaction of boundary conditions and inherent handling of geometric singularities.

Validation on rectangular, trapezoidal and and irregular non-convex reservoirs confirms BINNs' superior performance. In regular domains, errors as low as O(10−4) MPa are achieved with minimal training. For geometrically complex trapezoidal cases with pressure singularities, BINNs match BEM accuracy (~0.115 MPa max error) using moderate-sized networks. Crucially, the successful application to production well simulation in domains with recessed boundaries highlights BINNs' distinct advantage in analytically handling internal singularities via Green's functions, demonstrating remarkable robustness where traditional methods often require extensive mesh refinement.demonstrating robustness to irregular boundaries. The framework's rapid convergence and boundary-only sampling highlight its computational efficiency and scalability.

BINNs thus offer a paradigm ssshift toward meshless, boundary-focused reservoir simulation. Future extensions to multiphase flow and fractured reservoirs are warranted, leveraging BINNs' inherent capacity for discontinuous physics, which holds significant potential to advance high-fidelity, computationally efficient forecasting in subsurface engineering.

Acknowledgement

Prof. Dr. Rao acknowledges the support from King Abdullah University of Science and Technology and Yangtze University.

Funding Statement

None.

Author Contributions

The authors confirm their responsibility for the following: study conception and design, data collection, analysis and interpretation of results, and manuscript preparation.

Availability of Data and Materials

None.

Conflicts of Interest

The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] Brebbia, C. A., Telles, J. C. F., & Wrobel, L. C. (2012). Boundary element techniques: theory and applications in engineering. Springer Science & Business Media.

[2] Dai, Y., Li, Z., An, Y., & Deng, W. (2024). Numerical solutions of boundary problems in partial differential equations: A deep learning framework with Green's function. Journal of Computational Physics, 511, 113121

[3] Damiano, S., & van Waterschoot, T. (2025). Sound Field Reconstruction Using Physics-Informed Boundary Integral Networks. arXiv preprint arXiv:2506.03917.

[4] Hughes, T. J. (2003). The finite element method:

linear static and dynamic finite element analysis. Courier Corporation.

[5] Krishnapriyan, A., Gholami, A., Zhe, S., Kirby, R., & Mahoney, M. W. (2021). Characterizing possible failure modes in physics-informed neural networks. Advances in neural information processing systems, 34, 26548-26560.

[6] LeVeque, R. J. (2002). Finite volume methods for hyperbolic problems (Vol. 31). Cambridge university press.

[7] Lin, G., Hu, P., Chen, F., Chen, X., Chen, J., Wang, J., & Shi, Z. (2023). BINet: Learn to solve partial differential equations with boundary integral networks. CSIAM Transactions on Applied Mathematics, 4, 275–305.

[8] Liu, B., Wei, J., Kang, L., Liu, Y., & Rao, X. (2025). Physics-informed neural network (PINNs) for convection equations in polymer flooding reservoirs. Physics of Fluids, 37(3).

[9] Meng B.,Lu Y., &Jiang Y. (2024). Solving Partial Differential Equations in Different Domains by Operator Learning method Based on Boundary Integral Equations. arXiv:2406.02298.

[10] 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.

[11] Rao, X. (2023). A generic workflow of projection-based embedded discrete fracture model for flow simulation in porous media. Computational Geosciences, 27(4), 561-590.

[12] Rao, X., Cheng, L., Cao, R., Jiang, J., Fang, S., Jia, P., & Wang, L. (2018). A novel Green element method based on two sets of nodes. Engineering Analysis with Boundary Elements, 91, 124-131.

[13] Rao, X., Cheng, L., Cao, R., Zhang, X., & Dai, D. (2019). A mimetic Green element method. Engineering Analysis with Boundary Elements, 99, 206-221.

[14] 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.

[15] Rao, X., Zhao, H., & Liu, Y. (2024). A novel meshless method based on the virtual construction of node control domains for porous flow problems. Engineering with Computers, 40(1), 171-211.

[16] Rao, X., Liu, Y., He, X., & Hoteit, H. (2025). Physics-informed Kolmogorov–Arnold networks to model flow in heterogeneous porous media with a mixed pressure-velocity formulation. Physics of Fluids, 37(7).

[17] Rao, X., Liu, Y., & Shen, Y. (2025). Quantum-Classical Physics-Informed Neural Networks for Solving Reservoir Seepage Equations. arXiv preprint arXiv:2512.03923.

[18] Sun, J., Liu, Y., Wang, Y., Yao, Z., & Zheng, X. (2023). BINN: A deep learning approach for computational mechanics problems based on boundary integral equations. Computer Methods in Applied Mechanics and Engineering, 410, 116012.

[19] Wang, S., Wang, H., & Perdikaris, P. (2021). On the eigenvector bias of Fourier feature networks: From regression to solving multi-scale PDEs with physics-informed neural networks. Computer Methods in Applied Mechanics and Engineering, 384, 113938.

[20] Zhang, H., Anitescu, C., Bordas, S., Rabczuk, T., & Atroshchenko, E. (2022). Artificial neural network methods for boundary integral equations [Preprint]. TechRxiv.

[21] Zhang, P., Xie, L., Gu, Y., Qu, W., Zhao, S., & Zhang, C. (2024). Boundary integrated neural networks for 2D elastostatic and piezoelectric problems. International Journal of Mechanical Sciences, 280, 109525.