

# STONet: A neural operator for modeling solute transport in micro-cracked reservoirs

Ehsan Haghighat<sup>1,\*</sup>, Mohammad Hesan Adeli<sup>2</sup>, S. Mohammad Mousavi<sup>3</sup>,  
and Ruben Juanes<sup>1</sup>

<sup>1</sup>Massachusetts Institute of Technology, Cambridge, MA

<sup>2</sup>Sharif University of Technology, Tehran, Iran

<sup>3</sup>Cornell University, Ithaca, NY

\*Corresponding author: [ehsanh@mit.edu](mailto:ehsanh@mit.edu)

## Abstract

In this work, we introduce a novel neural operator, the Solute Transport Operator Network (STONet), to efficiently model contaminant transport in micro-cracked porous media. STONet's model architecture is specifically designed for this problem and uniquely integrates an enriched DeepONet structure with a transformer-based multi-head attention mechanism, enhancing performance without incurring additional computational overhead compared to existing neural operators. The model combines different networks to encode heterogeneous properties effectively and predict the rate of change of the concentration field to accurately model the transport process. The training data is obtained using finite element (FEM) simulations by random sampling of micro-fracture distributions and applied pressure boundary conditions, which capture diverse scenarios of fracture densities, orientations, apertures, lengths, and balance of pressure-driven to density-driven flow. Our numerical experiments demonstrate that, once trained, STONet achieves accurate predictions, with relative errors typically below 1% compared with FEM simulations while reducing runtime by approximately two orders of magnitude. This type of computational efficiency facilitates building digital twins for rapid assessment of subsurface contamination risks and optimization of environmental remediation strategies. The data and code for the paper will be published at <https://github.com/ehsanhaghighat/STONet>.

**Keywords:** Machine Learning; Neural Operators; Fractured Porous Media; Solute Transport.

## 1. Introduction

The depletion of freshwater resources is a pressing global challenge, particularly in regions facing severe droughts leading to the rapid exhaustion of groundwater reserves. A

significant factor contributing to water quality degradation in underground aquifers is the intrusion of seawater: the higher density of saline water facilitates its rapid dispersion and mixing within freshwater aquifers, leading to the groundwater contamination [6, 7]. Assessing the risk of seawater intrusion and developing mitigating strategies requires quantitative modeling of coupled flow and solute transport in porous media [1, 41]. These assessments are further complicated by the common occurrence of fractures in the subsurface, which can significantly alter the flow: typically, fractures exhibit higher permeability than the surrounding domain, thus profoundly modulating groundwater flow and transport [22, 34, 46, 8, 10, 19]. Other factors that can affect the solute transport problem include, but not limited to, transport under partially saturated conditions [21, 57, 42] and as well as other environmental and mechanical conditions such as erosion [39, 35, 43].

Accounting for fractures in the modeling process generally increases the complexity of the computational models of groundwater flow and transport [8, 27]. However, in cases where the size of fractures is much smaller than other dimensions of interest, upscaling approaches like the equivalent continuum model can be employed to implicitly incorporate the impact of these so-called micro-fractures in the modeling framework [38, 56, 25]. Khoei et al. [26] employed this approach extensively in their study by introducing inhomogeneities in the form of micro- and macro-fractures into a homogeneous benchmark problem known as Schincariol [45, 36]. Their investigation focused on assessing the influence of micro-fractures, both in the presence and absence of macro-fractures, on solute transport in the medium. This study leverages their work to create a dataset for training a neural operator.

Numerical modeling techniques, also known as forward models, such as the finite element method (FEM) have traditionally been employed to simulate flow and transport in porous media [44]. Forward models rely on an accurate understanding of model parameters, which are mostly unknown for subsurface applications except at sparse observation or injection/production wells. Therefore, repeated simulations are often performed to find model parameters while matching the data at wells. Although powerful, each forward simulation is computationally intensive, particularly for realistic, three-dimensional simulations, where computational runtimes can extend to several hours or even days for a single scenario. This significant computational demand severely limits their applicability for real-time analyses and identification or optimization tasks, where numerous simulations are necessary to explore parameter spaces or identify optimal reservoir management strategies. In contrast, machine learning (ML) techniques, particularly neural operators, offer highly efficient *inference* capabilities once trained, with prediction times typically reduced to seconds or less [33, 3, 30]. Furthermore, ML models inherently provide analytical differentiation, a feature invaluable for optimization and sensitivity analysis. As a result, ML-based surrogate models have increasingly become an attractive and practical solution for rapid assessment and optimization of complex subsurface and groundwater

**Machine learning (ML)** Over the past few years, there has been an explosive increase in the development and application of deep learning (DL) approaches, partly as a result of data availability and computing power [31]. Recent advances in deep learning approaches have pushed engineers and scientists to leverage ML frameworks for solving classical engineering problems. A recent class of DL methods, namely, Physics-Informed Neural Networks (PINNs), have received increased attention for solving forward and inverse problems and for building surrogate models with lesser data requirements [40, 23, 5]. PINNs leverage physical principles and incorporate them into the optimization process, enabling the network to learn the underlying physics of the problem. The applications of this approach extend to fluid mechanics, solid mechanics, heat transfer, and flow in porous media, among others [20, 24, 11, 14, 4, 37, 15, 12, 2, 52, 55, 53]. A recent architecture, namely Neural Operators, provides an efficient framework for data-driven and physics-informed surrogate modeling [33, 32, 49, 49, 50, 13, 29, 16]. Neural Operators are a class of neural networks that operate on functions rather than vectors, enabling them to capture the relationships between input and output functions. By leveraging the power of Neural Operators, it is possible to construct surrogate models that are both data-efficient and accurate. This makes Neural Operators well-suited for problems where experimental data is limited or computationally expensive to obtain. Once trained, neural operators can be used to perform inference efficiently. Neural operators have recently been used to model transport in porous media [51, 9, 17].

**Our contributions** In this study, we developed a neural operator specifically designed for modeling density-driven flow in fractured porous media. The neural operator leverages the power of deep learning to capture the complex relationships between the equivalent permeability tensor, which is a result of variations in fracture orientation and fracture density, and the pressure gradient, and outputs the spatio-temporally varying concentration field. The new architecture, as detailed in section 3.2, revises the previously introduced En-DeepONet [16] to achieve higher accuracy at the same computational cost.

A key contribution of STONet is its careful selection and encoding of input and output features tailored for the flow and solute transport problem. The input features include the equivalent permeability tensor which encodes both intrinsic permeability as well as fracture statistics (orientation, density, length, aperture), and pressure boundary conditions. The output is the rate of change of concentration, which enables flexible and stable auto-regressive prediction of the concentration field over time. Additionally, STONet introduces a novel attention block with residual connections that operates on the encoded features from the branch and trunk networks. The residual connections ensure stable training and facilitate the propagation of important information from the branch network, which encodes the spatially varying physical properties, throughout the network. This design leads to improved accuracy and generalization compared to other DeepONet

architectures. 112

We generated a dataset obtained using high-fidelity finite element simulations to train 113  
and validate the neural operator. This dataset serves as a benchmark for evaluating the 114  
performance of the proposed neural operator. We applied the developed framework to 115  
predict flow patterns in porous media under heterogeneous conditions, demonstrating its 116  
ability to handle complex geological scenarios. 117

## 2. Governing Equations 118

The governing equations describing the solute transport in fractured porous media include 119  
the mass conservation equation for the fluid phase and solute component. The fluid mass 120  
conservation is expressed as 121

$$\frac{\partial}{\partial t}(\phi\rho) + \nabla \cdot (\rho\mathbf{v}_m) = 0, \quad (1)$$

where  $\phi$  represents the matrix porosity,  $\rho$  denotes the fluid density that varies with the 122  
solute mass fraction (concentration), and  $\mathbf{v}_m$  is the fluid phase velocity vector within the 123  
matrix. This velocity can be expressed in terms of pressure by applying Darcy's law as 124

$$\mathbf{v}_m = -\frac{\mathbf{k}_m}{\mu}(\nabla p - \rho\mathbf{g}). \quad (2)$$

Here,  $\mu$  denotes the fluid viscosity,  $\mathbf{g}$  represents the gravitational acceleration, and  $\mathbf{k}_m$  125  
stands for the permeability tensor of the matrix. In eq. (1), it is assumed that the fluid is 126  
incompressible and the matrix porosity remains constant over time. Substituting Darcy's 127  
law into eq. (1) and neglecting density variations except in the terms involving gravity 128  
(Boussinesq approximation [54, 8]), one obtains 129

$$\nabla \cdot \left( -\frac{\mathbf{k}_m}{\mu}(\nabla p - \rho\mathbf{g}) \right) = 0. \quad (3)$$

As stated earlier, the density is a function of the solute mass fraction. Assuming a linear 130  
state equation [7], the density function is expressed as 131

$$\rho(c) = \rho_0 + \frac{\rho_s - \rho_0}{\rho_0}c, \quad (4)$$

where  $c$  is the mass fraction, taking a value between 0 and 1, and  $\rho_0$  and  $\rho_s$  are the 132  
reference values for the fluid density at zero mass fraction (pure water) and at unit mass 133  
fraction (pure solute), respectively. Additionally,  $\mathbf{k}_m$  is influenced by both the intrinsic 134  
permeability of the pore structure and the geometric characteristics of micro-fractures. 135  
To compute  $\mathbf{k}_m$ , the domain must be divided into Representative Elementary Volumes 136  
(REV). For each REV, the specific permeability matrix can be determined using the 137

equivalent continuum model, outlined in [26], as

138

$$\mathbf{k}_m = \mathbf{k}_r + \frac{1}{12\Sigma} \sum_i b_i^3 l_i \mathbf{M}_i. \quad (5)$$

The first part of the equivalent permeability tensor eq. (5), i.e.,  $\mathbf{k}_r$ , is an isotropic tensor related to the intrinsic permeability of the pore structure. Variables  $b_i$  and  $l_i$  correspond to the aperture and length of micro-fractures, respectively.  $\Sigma$  is the volume of the REV, and  $\mathbf{M}_i$  denotes the conversion matrix defined as

142

$$\mathbf{M}_i = \mathbf{I} - \mathbf{n}_i \otimes \mathbf{n}_i, \quad (6)$$

where  $\mathbf{n}_i$  is the unit vector normal to the  $i^{\text{th}}$  micro-fracture. Note that, while  $\mathbf{k}_r$  is an isotropic tensor,  $\mathbf{M}_i$  is anisotropic due to the varied orientations of the micro-fractures.

143

144

The next governing equation pertains to the conservation of mass for the solute within the fluid phase and can be written as

145

146

$$\phi \frac{\partial}{\partial t} (\rho c) + \nabla \cdot (\rho c \mathbf{v}_m) - \nabla \cdot (\rho \mathbf{D}_m \nabla c) = 0, \quad (7)$$

in which Fick's law is used for the dispersive and diffusive flux of solute components.  $\mathbf{D}_m$  is the dispersion-diffusion tensor, which is a function of the velocity  $\mathbf{v}_m$ , and expressed as

147

148

$$\mathbf{D}_m = \phi \tau D_m \mathbf{I} + (\alpha_L - \alpha_T) \frac{\mathbf{v}_m \otimes \mathbf{v}_m}{|\mathbf{v}_m|} + \alpha_T |\mathbf{v}_m| \mathbf{I}. \quad (8)$$

Here, the first term in eq. (7) denotes the rate of change in the mass of the solute component, while the second and third terms denote the advective and dispersive transport mechanisms of the solute component, respectively.  $D_m$  refers to the molecular diffusion coefficient associated with the matrix,  $\tau$  represents the tortuosity of the porous medium,  $\mathbf{I}$  is the identity tensor, and  $\alpha_L$  and  $\alpha_T$  denote the longitudinal and transverse dispersivities, respectively. The solution to the aforementioned equations can be achieved through the finite element method, as detailed in [26].

149

150

151

152

153

154

155

### 3. STONet: Neural Operator for solute transport in fractured porous media

156

157

Neural operators are a class of machine learning models designed to learn mappings between infinite-dimensional function spaces. They are particularly well-suited for solving parameterized partial differential equations (PDEs) that arise in various physical phenomena. Unlike traditional neural networks that operate on fixed-dimensional vectors, neural operators can handle functions as inputs and outputs, making them ideal for constructing surrogate models for continuous space-time problems. In this section, we review neural

158

159

160

161

162

163

### 3.1 Enriched DeepONet

The goal of a neural operator is to learn a mapping  $G : U \rightarrow V$  between two function spaces  $U$  and  $V$ . For the problem of solute transport,  $U$  might represent the space of initial and boundary conditions, fracture properties such as orientation, length, and opening, and medium parameters such as permeability and porosity, while  $V$  represents the space of the solute concentration field over time. A neural operator typically consists of two main components: (1) a feature-encoding network, known as the branch network (B); and (2) a query network, known as the trunk network (T). They process the input function  $u \in U$  and encode relevant features. The final output of the neural operator is obtained by combining the outputs of the branch and trunk networks through a suitable operation, such as elementwise multiplication or a learned fusion mechanism (a final network).

DeepONet and its generalization Enriched-DeepONet [16] have proven a good candidate for learning continuous functional spaces. The neural architecture for En-DeepONet is depicted in fig. 1(a), and expressed mathematically as

$$\epsilon_B = B(u; \boldsymbol{\theta}_B), \quad (9)$$

$$\epsilon_T = T(x; \boldsymbol{\theta}_T), \quad (10)$$

$$G(u)(x) = R(\epsilon_{B \odot T}, \epsilon_{B \oplus T}, \epsilon_{B \ominus T}; \boldsymbol{\theta}_R), \quad (11)$$

where  $x \in X$  and  $u \in U$  are a query point in the solution domain  $\Omega$  and a problem parameter set, respectively. Here,  $\boldsymbol{\theta}_\alpha$  represents the set of parameters of each network, and  $\epsilon_B, \epsilon_T$  are encoded outputs of the branch and trunk networks, respectively.  $\odot, \oplus, \ominus$  denote elementwise multiplication, addition, and subtraction of branch and trunk encodings, respectively.  $R$  is a fusion network, known as the root network, which decodes the final outputs.

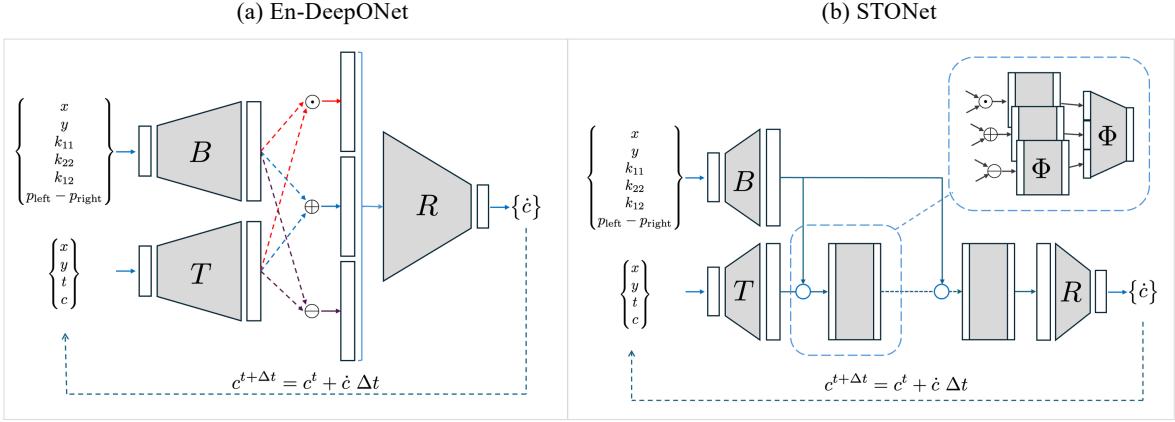


Figure 1: Network architecture. (a) En-DeepONet neural architecture [16]. (b) The revised En-DeepONet architecture, namely STONet. STONet resembles the multi-head attention mechanism of transformer architecture [47] with residual connections [18] and applies the multiplication, addition, and subtraction operations on different layers.

### 3.2 STONet

185

Here, we present an extension of the En-DeepONet architecture that resembles the multi-head attention mechanism of transformer networks [47] with residual connections [18]. The network architecture is depicted in fig. 1(b). The architecture consists of an encoding branch and a trunk network. The output of the branch network is then combined with the output of the trunk network, using elementwise operations such as multiplication or addition, and passed to the new attention block. One may add multiple attention blocks. The final output is then passed on to the output root network. The network architecture is expressed mathematically as

$$\epsilon_B = B(u; \theta_B), \quad (12)$$

$$\epsilon_T = T(x; \theta_T), \quad (13)$$

$$\epsilon_Z^0 = \epsilon_T, \quad (14)$$

$$\epsilon_{B \odot Z}^l = \Phi(\epsilon_{B \odot Z}^{l-1}; \theta_Z^{\odot, l}), \quad l = 1 \dots L, \quad (15)$$

$$\epsilon_{B \oplus Z}^l = \Phi(\epsilon_{B \oplus Z}^{l-1}; \theta_Z^{\oplus, l}), \quad l = 1 \dots L, \quad (16)$$

$$\epsilon_{B \ominus Z}^l = \Phi(\epsilon_{B \ominus Z}^{l-1}; \theta_Z^{\ominus, l}), \quad l = 1 \dots L, \quad (17)$$

$$\epsilon_Z^l = \Phi(\epsilon_{B \odot Z}^l, \epsilon_{B \oplus Z}^l, \epsilon_{B \ominus Z}^l; \theta_Z^l), \quad l = 1 \dots L, \quad (18)$$

$$G(u)(x) = R(\epsilon_Z^L; \theta_R), \quad (19)$$

where  $\Phi$  denotes a single fully-connected layer, and  $L$  is the total number of attention blocks. Lastly, the network is trained on the concentration rate, therefore concentration field is predicted auto-regressively using the forward Euler update as

$$c^{t+\Delta t} = c^t + G(u)(x)\Delta t. \quad (20)$$

Regarding the choice of input and output features, we performed many experiments 197  
to find the best-performing architecture, but we avoided reporting all cases here. Those 198  
experiments include: 199

- Predicting  $c$  instead of  $\dot{c}$ . This is not a good choice because it imposes a fixed- 200  
time stepping on the auto-regressive updates, therefore less generalizable. However, 201  
we still tested the architecture but we did not observe any improvements on this 202  
dataset. 203
- Adding velocity field  $\mathbf{v}_m$  as input and output features. Based on the transport 204  
eq. (7), it is clear  $\mathbf{v}_m$  plays an important role in the transport of the contaminant 205  
in the domain. Adding this, however, did not improve the results. Note that 206  
 $\mathbf{v}_m$  is strongly correlated with other features already included, including, equivalent 207  
permeability field and boundary condition ( $p_{\text{left}} - p_{\text{right}}$ ). We suspect that the reason 208  
it did not improve the results here is because of the simplicity of the domain and 209  
the overall distribution of data. However, for more generic cases, the velocity field 210  
(or pressure gradient) should also be added as input and output features, possibly 211  
as a separate network, to achieve more generalization. Worth noting that adding 212  
 $\mathbf{v}_m$  increased the training time and memory requirements. 213

### 3.3 Optimization

The optimization of neural operators involves adjusting the parameters of the branch, 215  
trunk, and root networks to minimize a loss function that measures the discrepancy 216  
between the predicted and true outputs. In this study, the network output is the concen- 217  
tration rate  $\dot{c}$ , as mentioned previously. Hence, the loss function is based on the mean 218  
squared error between the predicted and observed solute concentrations. The optimiz- 219  
ation is performed using the Adam optimizer [28], a variant of the stochastic gradient 220  
descent. The gradients are computed using backpropagation through the computational 221  
graph of the neural operator. Regularization techniques, such as weight decay or dropout, 222  
may be employed to prevent overfitting and improve the generalization of the model. 223

## 4. Results and Discussion

In this section, we present the results of our numerical experiments on the performance of 225  
the proposed neural operator for surrogate modeling of solute transport in micro-cracked 226  
reservoirs. We compare the accuracy and computational efficiency of our approach to the 227  
finite element method. Additionally, we evaluate the effectiveness of the neural operator 228  
in handling different input scenarios and encoding heterogeneous properties of the porous 229  
medium. Finally, we discuss the potential applications of our model in environmental 230  
impact assessment and groundwater management. 231

## 4.1 Problem Description and Parameter Specification

232

A schematic of the problem considered in this study of a solute through a micro-cracked reservoir is shown in fig. 2. The domain under consideration has dimensions of  $70 \times 50$  cm. The deterministic parameters of the problem are given in Table 1. The solution is injected from the left side and transported through the domain, driven by a pressure difference between the left and right boundaries. The porous reservoir is assumed to be confined between two impervious layers and contains randomly distributed micro-cracks of varying density and orientations. The micro-fracture orientations are sampled randomly from a normal distribution with varying mean values but fixed standard deviation, i.e.,  $\theta \sim \mathcal{N}(\mu_\theta, \sigma_\theta = 15^\circ)$ , where  $\mu_\theta$  is sampled uniformly as  $\mu_\theta \sim \mathcal{U}(-60^\circ, 60^\circ)$ . The representative elementary volume (REV) dimensions for assessing the permeability field (using eq. (5)) are assumed  $10 \times 10$  cm. Fracture density, i.e., the number of cracks per REV, is sampled from a Poisson's distribution as  $\Sigma \sim \mathcal{P}(\lambda)$ , where  $\lambda$  is sampled uniformly as  $\lambda \sim \mathcal{U}(30, 70)$ . Fracture length and aperture are sampled from log-normal distributions  $\log(l_c) \sim \mathcal{N}(\mu_{l_c} = 0.05, \sigma_{l_c} = 0.0575)$  and  $\log(a) \sim \mathcal{N}(\mu_a = 1.14 \times 10^{-4}, \sigma_a = 1.15 \times 10^{-4} \times 1.5)$ , respectively. The pressure on the left side is kept fixed, while the pressure on the right side is also randomly perturbed from a normal distribution, i.e.,  $p_{right} = \mathcal{U}(4976, 4996) + 9792.34y$ . Figure 3 depicts a sample medium with micro-fractures. The sampling procedure for finding stochastic parameters is described in Algorithm 1.

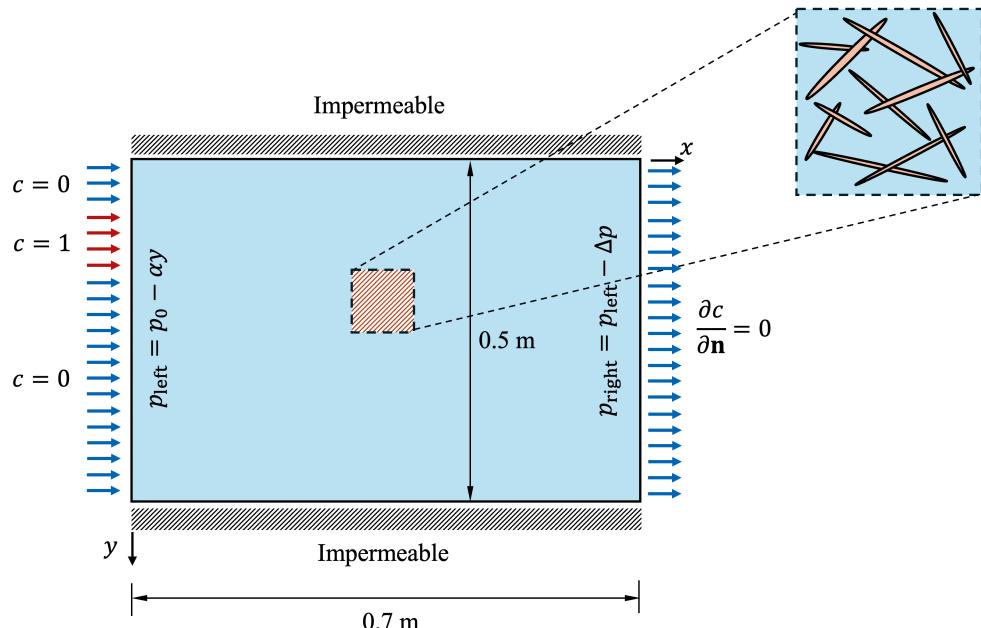


Figure 2: Problem setup. The figure depicts the original problem studied by [45]. The right pressure, as well as microcrack orientation and density (number of fractures per REV), are assigned randomly.

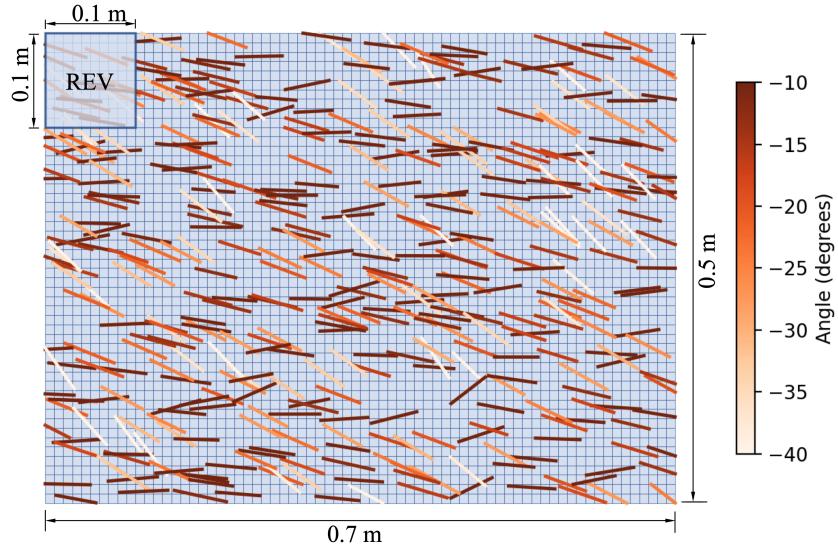


Figure 3: A sample realization of micro-crack distributions.

Table 1: Deterministic parameters.

Parameter	Symbol	Value	Units
Gravitational acceleration	$g$	9.81	( $m/s^2$ )
Water density	$\rho_0$	998.2	( $kg/m^3$ )
Brine density	$\rho_s$	1002	( $kg/m^3$ )
Viscosity	$\mu$	$1.002 \times 10^{-3}$	(Pa.s)
Porosity	$\phi$	0.38	
Diffusion coefficient	$D_m$	$1.61 \times 10^{-9}$	( $m^2/s$ )
Intrinsic permeability	$k_r$	$5.7 \times 10^{-11}$	( $m^2$ )
Longitudinal dispersivity	$\alpha_l$	$1 \times 10^{-3}$	( $m$ )
Transverse dispersivity	$\alpha_t$	$2 \times 10^{-4}$	( $m$ )
Tortuosity	$\tau$	1	

Figure 4 depicts a few snapshots of the solute concentration for different micro-crack 251 distributions. The equivalent permeability field  $k_x$  ( $k_{11}$ ) and  $k_y$  ( $k_{22}$ ) are plotted in the left 252 two columns, while the right two columns highlight concentration and the rate of change 253 of concentration at the last time step (i.e.,  $t = 36$  hr). The changes in the concentration 254 patterns are due to different fracture orientations and pressure gradients (or velocity field 255  $\mathbf{v}_m$ ), as shown in fig. 5. 256

---

**Algorithm 1** Sampling steps for stochastic parameters

---

- 1:  $N\text{-samples} \leftarrow$  Total number of samples (500).
- 2:  $N\text{-quadrature} \leftarrow$  Total number of quadrature points inside the domain.
- 3: **for**  $k \leftarrow 1$  to  $N\text{-samples}$  **do**
- 4:     **Sample global parameters (fixed throughout the domain)**
- 5:      $\mu_\theta \sim \mathcal{U}(-60^\circ, 60^\circ) \leftarrow$  Sample  $\mu_\theta$  uniformly between  $-60^\circ$  and  $60^\circ$  for the distribution of micro-fractures within the domain.
- 6:      $\lambda \sim \mathcal{U}(30, 70) \leftarrow$  The global Poisson distribution parameter of fracture density, sampled uniformly between 30 and 70.
- 7:     **Sample local parameters (at each integration point)**
- 8:     **for**  $i \leftarrow 1$  to  $N\text{-quadratures}$  **do**
- 9:          $\theta^i \sim \mathcal{N}(\mu_\theta, \sigma_\theta = 15^\circ) \leftarrow$  Sample  $\theta$  at every integration point inside the discretized Finite Element mesh.
- 10:          $\Sigma^i \sim \mathcal{P}(\lambda) \leftarrow$  Sample fracture density at each integration point using the global value for  $\lambda$ .
- 11:          $\log(l_c^i) \sim \mathcal{N}(\mu_{l_c} = 0.05, \sigma_{l_c} = 0.0575) \leftarrow$  Sample fracture length per integration point.
- 12:          $\log(a^i) \sim \mathcal{N}(\mu_a = 1.14 \times 10^{-4}, \sigma_a = 1.15 \times 10^{-4} \times 1.5) \leftarrow$  Sample fracture aperture per integration point.
- 13:         **Eval equivalent permeability tensor**
- 14:          $\Omega_i \leftarrow$  The Set of all quadrature points within REV defined around integration point  $i$
- 15:          $\mathbf{k}_r \leftarrow$  Intrinsic permeability of pore structure.
- 16:          $\mathbf{M}_i \leftarrow$  Compute conversion matrix over  $\Omega_i$ , defined in eq. (6).
- 17:          $\mathbf{k}_m \leftarrow$  Compute equivalent permeability tensor for integration point  $i$  over  $\Omega_i$ .
- 18:         **end for**
- 19:          $c \leftarrow$  Solve for concentration for each sample and at each time step.
- 20:          $\dot{c} \leftarrow$  Eval  $\dot{c}$  for each sample and at each time step using backward Euler.
- 21:         Store data for sample  $k$  and move to next simulation.
- 22: **end for**

---

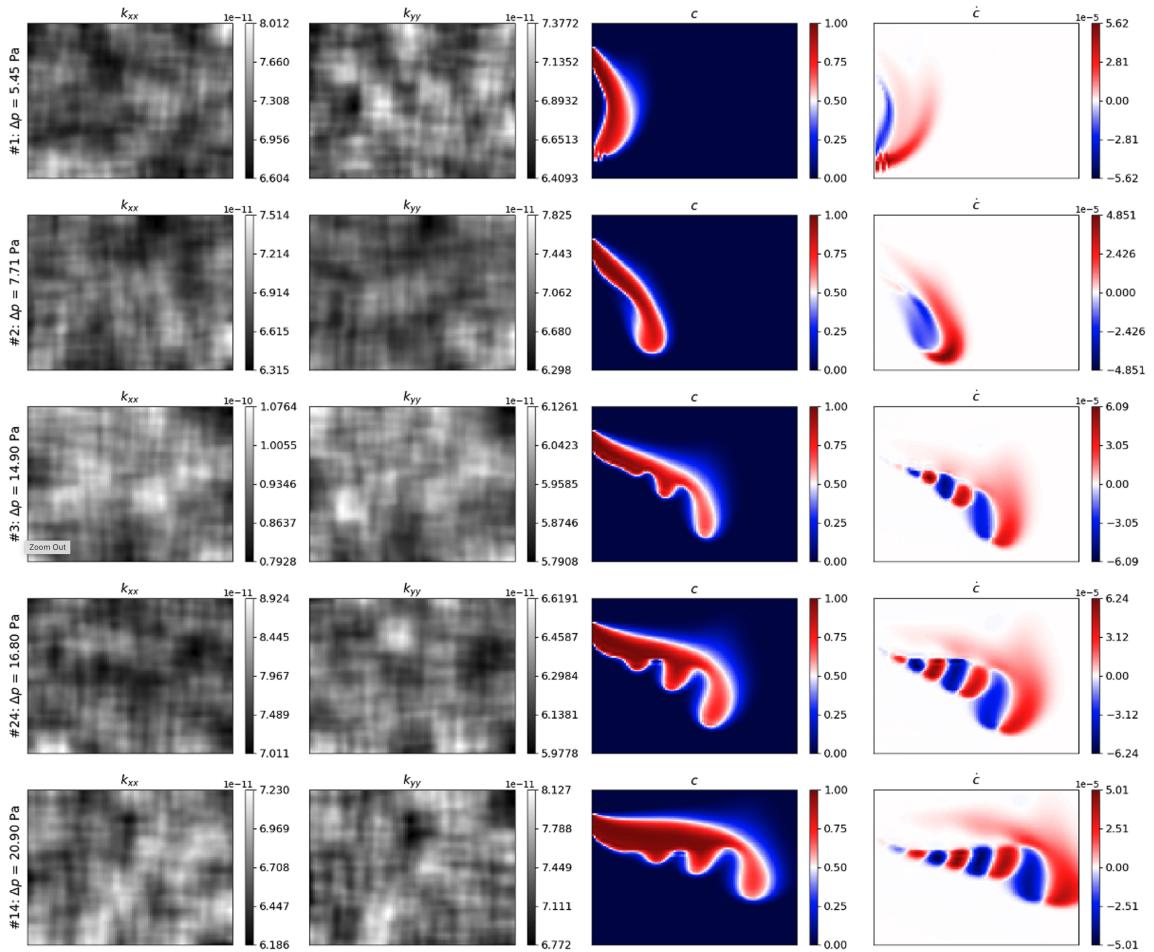


Figure 4: Sample solute concentration for different fracture patterns. Each row represents a different realization of fracture and pressure. The left two columns depict the  $k_{xx}$  and  $k_{yy}$  components of the equivalent permeability tensor. The right two columns represent the concentration and its rate of change, respectively, at the last time step (i.e.,  $t = 36$  h).

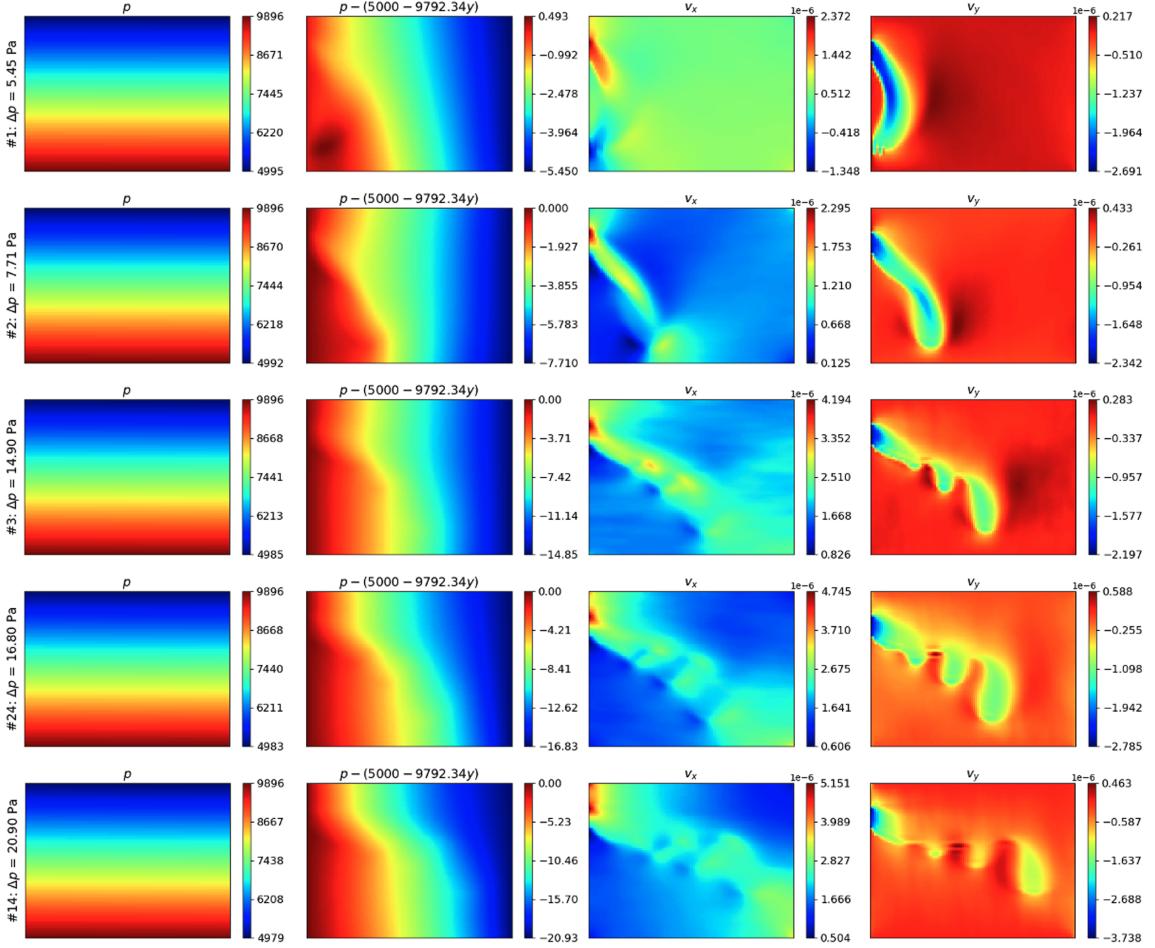


Figure 5: Sample pressure and velocity field for different fracture patterns. Each row represents a different realization, corresponding to each sample in fig. 4. The first column shows the absolute pressure field, the second column shows the relative pressure field (by subtracting the background pressure), and the third and fourth columns show the  $x$  and  $y$  components of the velocity field  $\mathbf{v}_m$ , respectively, at the last time step (i.e.,  $t = 36$  h).

The training dataset consists of 500 FEM simulations, and the test dataset consists of 25 random unseen samples. The domain is discretized using an element dimension of  $1 \times 1$  cm, resulting in a total of 3,500 elements and 3,621 nodes. We run the simulation for a total of 36 h using 1200 s implicit time increments. However, the outputs are recorded only at 4 h time increments. The generated dataset covers a wide range of fracture densities, orientations, and lengths, providing a diverse set of training examples for our neural operator.

## 4.2 Sampling Strategy

Since concentration remains near zero for a large portion of the domain (as shown in fig. 4), to reduce the batch size and computational demand, we sub-sample 1,500 random nodes using an importance sampling strategy. Out of these, 1,000 points were selected based on concentration density, while the remaining 500 nodes were uniformly distributed over the domain. Therefore, sampling points are random in space and over time.

### 4.3 Training Performance

270

Let us first compare the performance of the new En-DeepONet architecture (i.e., STONet) 271 with respect to the original architecture. To this end, we vary the network width and 272 embedding dimensions in  $\{50, 100\}$ , the number of layers of the branch and trunk networks 273 in  $\{4, 8, 12\}$ , and the number of the layers of the root network in  $\{4, 8, 12\}$  and number of 274 attention blocks in  $\{4, 8\}$ . The results are shown in fig. 6, where circles indicate the loss 275 with the traditional En-DeepONet architecture, and the star symbols the loss with the 276 new STONet architecture. We observe that for a similar number of parameters, the new 277 STONet architecture outperforms the previous architecture without an increase in the 278 computational cost. We also observe that both architectures improve their performance 279 for larger network widths. 280

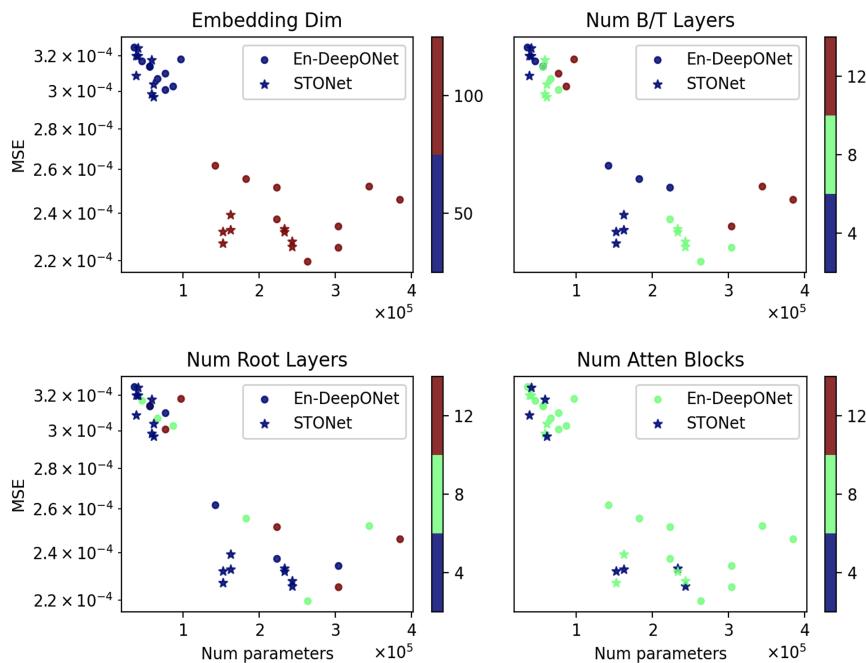


Figure 6: The training performance of the new STONet architecture versus the old En-DeepONet architecture. The x-axis shows the total number of parameters, and the y-axis presents the average loss values of the last 20 epochs for training performed for 500 epochs. Each subplot shows the variations with respect to the width of the network, number of branch and trunk layers, number of root layers, and number of attention blocks, respectively.

Next, to arrive at the optimal neural architecture for modeling this dataset, we explored several network sizes. The first variable is the network width of all networks (i.e., 281 B, T, R,  $\Phi$ ) along with their output dimension (embedding) from  $\{50, 100, 150, 200\}$ . The 282 second variable is the number of layers in the branch and trunk networks from  $\{2, 4, 8, 12\}$ . 283 The third variable is the number of attention blocks from  $\{2, 4, 8, 12\}$ . The training is 284 performed for 2,000 epochs, and the average loss for the last 100 epochs is compared. 285

The results of hyper-parameter exploration are shown in fig. 7. The horizontal axis 287 shows the total number of parameters, while the vertical axis presents the average of the 288

loss value for the last 100 epochs. It is apparent that wider networks lead to a significant 289 improvement in performance. The optimal choice of parameters seems to be 100 for 290 network width, 8 layers for the branch and trunk networks, 8 attention blocks, and 2 root 291 layers. Therefore, this architecture is utilized with further training (up to 50,000 epochs) 292 to arrive at the results presented in the next section. 293

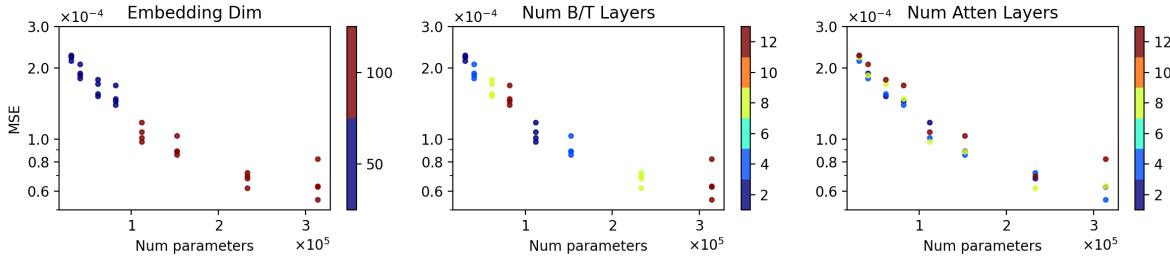


Figure 7: STONet hyper-parameter optimization. The x-axis shows the total number of parameters, and the y-axis presents the average loss values of the last 100 epochs for training performed for 2,000 epochs.

#### 4.4 Model Performance

Figure 8 depicts the predictions for the concentration on five random realizations from the 295 test set. The corresponding predictions for the rate of change of concentration are shown 296 in fig. 9. The results from the full-physics simulations for these five test cases were shown 297 earlier in fig. 4. It is apparent that STONet predicts the full-physics results accurately. 298 The fundamental difference is that the STONet, having been pre-trained, can be used for 299 fast prediction of density-driven flow and transport with any new fracture network, while 300 the FEM simulation would need to be recomputed altogether for any new configuration. 301

The distributions for pointwise absolute and relative error at different time steps for 302 concentration and concentration rate are plotted in Figure 10. Overall, STONet’s predic- 303 tion error is very small, with most predictions having below 1% error. The top figures 304 depict the absolute error distributions and highlight the presence of accumulation error 305 as can be observed from widening distributions at different time steps. However, the 306 distribution of relative error at different time steps, as shown in the bottom figures, re- 307 mains nearly unchanged, indicating a predictable fixed error distribution over time, which 308 is highly desirable. This is also confirmed by inspecting the mean absolute and relative 309 error evolution over time, as plotted in fig. 11. It is worth noting that the accumulation 310 error might be controlled with the addition of observational data and additional training 311 samples using data assimilation techniques such as Active Learning [48]. 312

## 5. Conclusions

In this study, we have presented a new Enriched-DeepONet architecture, STONet, for 314 emulating density-driven flow and solute transport in micro-cracked reservoirs. Our ap- 315

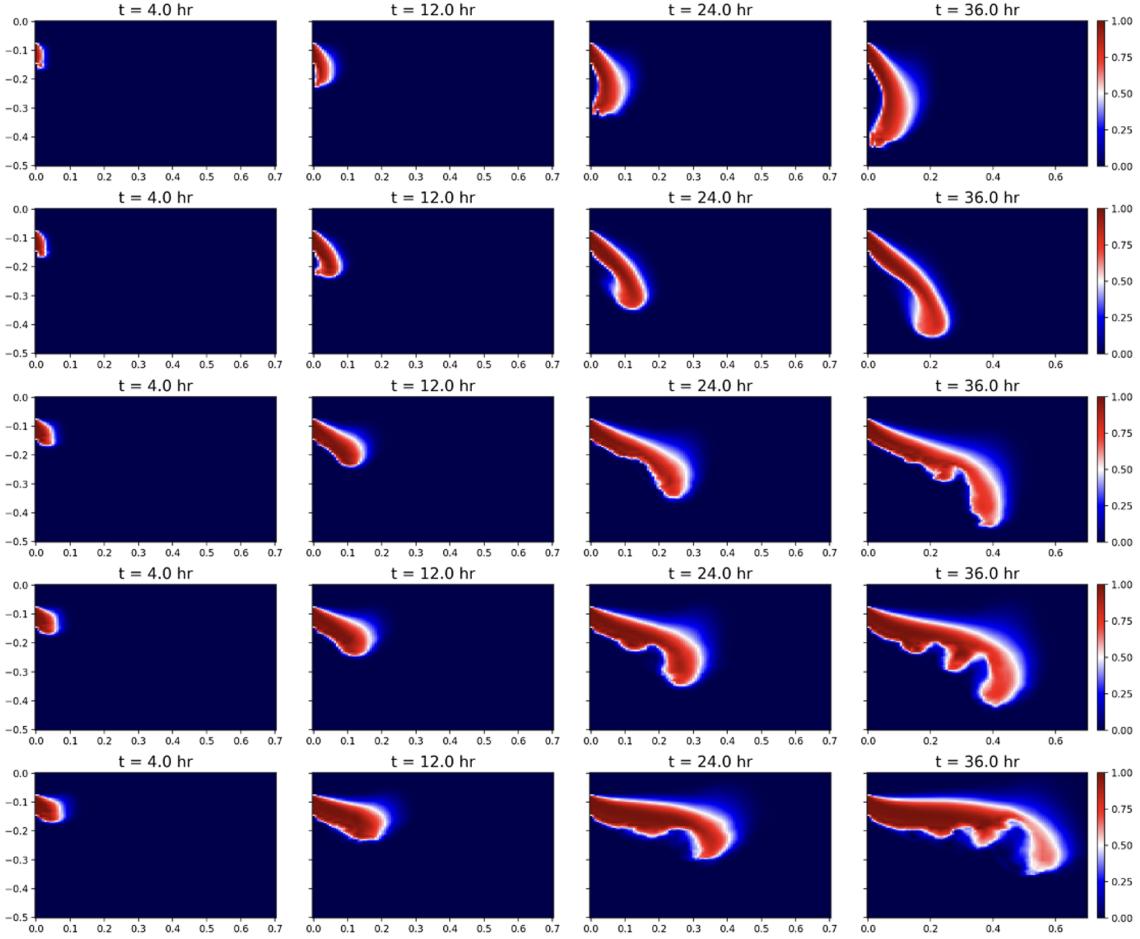


Figure 8: STONet predictions of the concentration field for five samples from the test (unseen) dataset, corresponding to each sample in fig. 4.

proach effectively encodes heterogeneous properties and predicts the concentration rate, 316 achieving accuracy comparable to that of the finite element method. The computational 317 efficiency of STONet enables rapid and accurate predictions of solute transport, facilitat- 318 ing both parameter identification and groundwater management optimization. 319

The STONet model developed in this work has the potential to be applied for fracture 320 network identification and efficient tracing and control of solute transport in micro-cracked 321 reservoirs. By rapidly and accurately predicting the concentration rate, the model can help 322 identify the location and connectivity of fractures in the porous media, which is crucial 323 for optimizing the management of groundwater resources. Additionally, the model can be 324 used to predict the transport of solutes in different scenarios, such as accidental contam- 325 inant spills, allowing for accurate and efficient decision-making. Overall, the ML model 326 has the potential to significantly improve the sustainable management of underground 327 aquifers, contributing to both local and global efforts towards sustainable groundwater 328 resource utilization. 329

While STONet demonstrates promising results for modeling solute transport in micro- 330 cracked reservoirs, several limitations remain. The current study is restricted to a single 331 domain geometry and accounts for limited variations in permeability but doesn't account 332

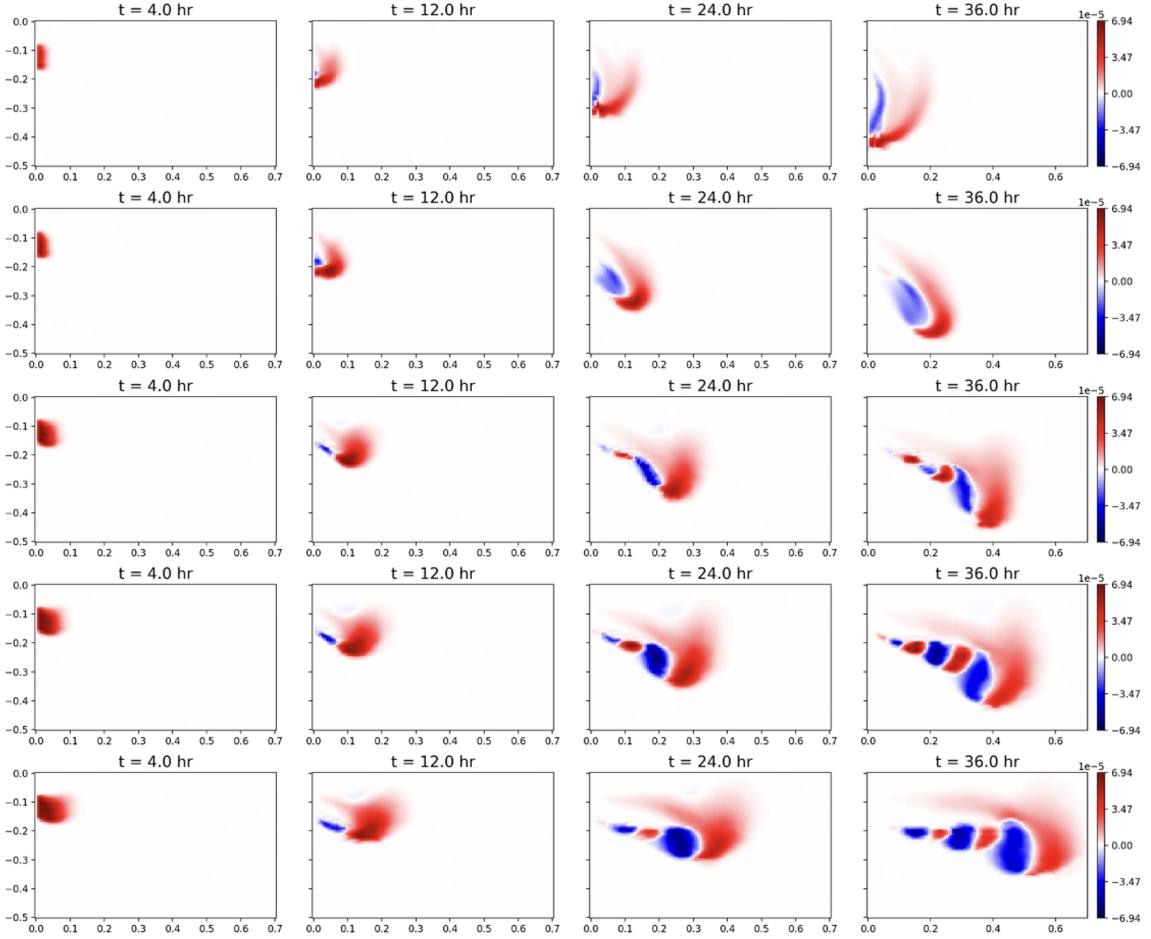


Figure 9: STONet predictions of the rate of change of concentration for five samples from the test (unseen) dataset, corresponding to each sample in fig. 4.

for the presence of distinct geological layers. Additionally, the model assumes a fixed injection location for the contaminant and does not consider scenarios where the contaminant flow originates from multiple or varying locations. The study also does not account for different pressure or flux boundary conditions, which are critical for capturing diverse real-world scenarios. Furthermore, the study is limited to a two-dimensional domain, whereas real-world applications often require three-dimensional modeling to capture the full complexity of subsurface transport. Considering all these choices may require hundreds of thousands of training samples, which is out of the scope of the current study. These limitations highlight the need for further research and development to extend the applicability of STONet to more generic and realistic scenarios.

## Conflicts of Interest

The authors have no conflicts of interest to declare.

343

344

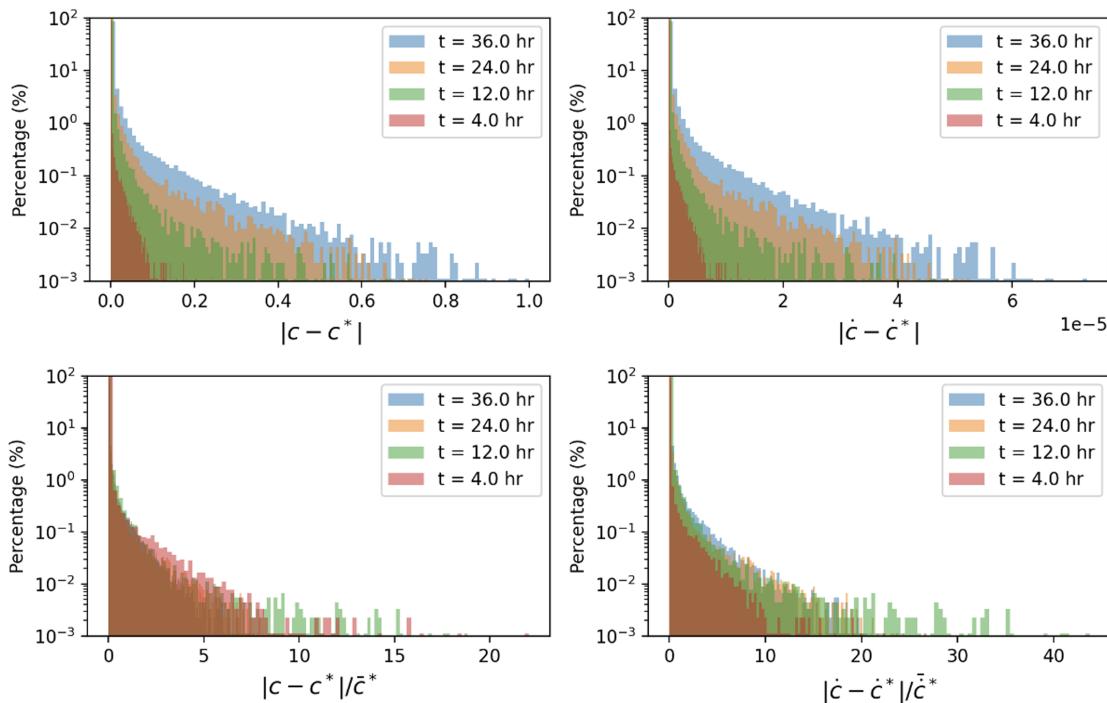


Figure 10: Unrolling error distribution. Pointwise absolute (top row) and relative (bottom row) error distribution for STONet predictions on all 25 test samples at different time steps for (Left) concentration and (Right) concentration rate. The error in the majority of points remains within 1%. While widening absolute error distributions point to error accumulation, the distribution of relative errors remains nearly constant, which is desirable.

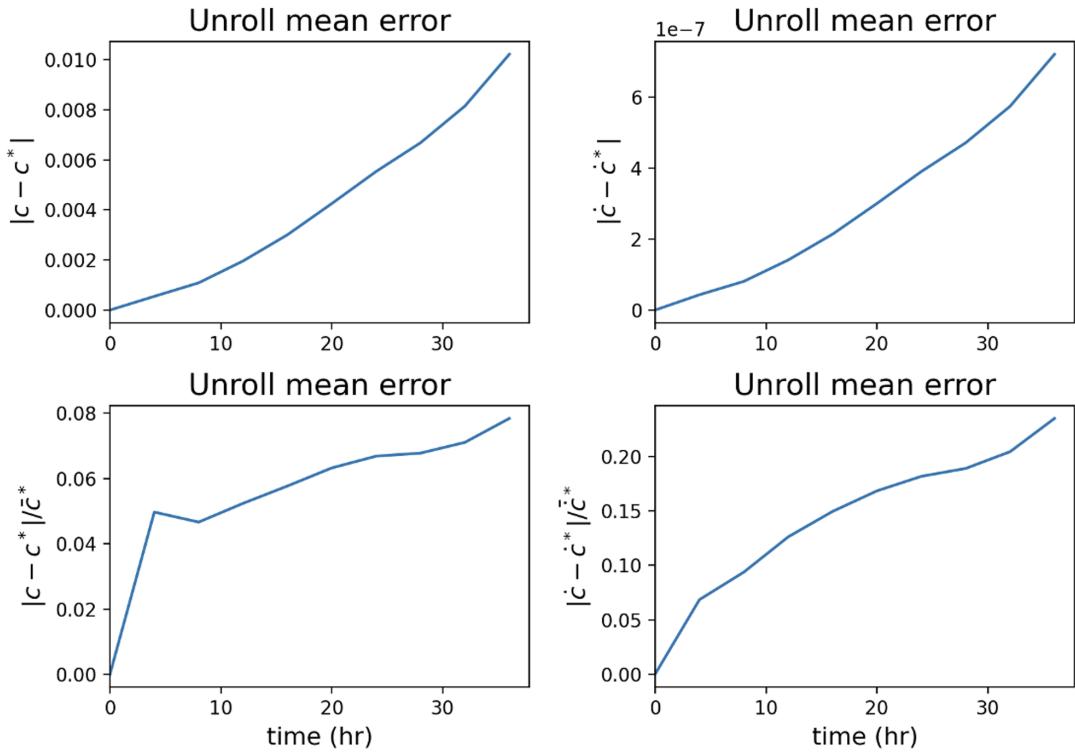


Figure 11: Unrolling error. The plot shows the mean absolute (top row) and relative (bottom row) error of the unrolling process as a result of the auto-regressive nature of the architecture. While the evolution of mean absolute error points to error accumulation, the evolution of mean relative errors remains nearly constant, which is desirable.

## Author Contributions

345

EH conceptualized the problem and contributed to the theoretical development and implementation. MHA contributed to the implementation, running the test cases and generating the figures. MM prepared the training dataset. All authors contributed to drafting and revising the manuscript.

346

347

348

349

## Data availability

350

The data and code for reproducing the results reported in this manuscript will be published at [351](https://github.com/ehsanhaghigat/STONet) [352](https://github.com/ehsanhaghigat/STONet)

## References

353

- [1] E. Abarca, E. Vázquez-Suñé, J. Carrera, B. Capino, D. Gámez, and F. Batlle. Optimal design of measures to correct seawater intrusion. *Water Resources Research*, 42(9):W09415, 2006.
- [2] D. Amini, E. Haghigat, and R. Juanes. Physics-informed neural network solution of

- thermo–hydro–mechanical processes in porous media. *Journal of Engineering Mechanics*, 148(11):04022070, 2022. 358  
359
- [3] K. Azizzadenesheli, N. Kovachki, Z. Li, M. Liu-Schiaffini, J. Kossaifi, and A. Anandkumar. Neural operators for accelerating scientific simulations and design. *Nature Reviews Physics*, 6(5):320–328, 2024. 360  
361  
362
- [4] S. Cai, Z. Mao, Z. Wang, M. Yin, and G. E. Karniadakis. Physics-informed neural networks (pinns) for fluid mechanics: A review. *Acta Mechanica Sinica*, 37(12):1727–1738, 2021. 363  
364  
365
- [5] S. Cuomo, V. S. Di Cola, F. Giampaolo, G. Rozza, M. Raissi, and F. Piccialli. Scientific machine learning through physics-informed neural networks: Where we are and what's next. *Journal of Scientific Computing*, 92(3):88, 2022. 366  
367  
368
- [6] M. Dentz, D. Tartakovsky, E. Abarca, A. Guadagnini, X. Sanchez-Vila, and J. Carrera. Variable-density flow in porous media. *Journal of fluid mechanics*, 561:209–235, 2006. 369  
370  
371
- [7] H.-J. Diersch and O. Kolditz. Variable-density flow and transport in porous media: approaches and challenges. *Advances in water resources*, 25(8-12):899–944, 2002. 372  
373
- [8] H.-J. G. Diersch. *FEFLOW: finite element modeling of flow, mass and heat transport in porous and fractured media*. Springer Science & Business Media, 2013. 374  
375
- [9] Y. Do Kim and L. J. Durlofsky. Neural network surrogate for flow prediction and robust optimization in fractured reservoir systems. *Fuel*, 351:128756, 2023. 376  
377
- [10] D. Faulkner, C. Jackson, R. Lunn, R. Schlische, Z. Shipton, C. Wibberley, and M. Withjack. A review of recent developments concerning the structure, mechanics and fluid flow properties of fault zones. *Journal of Structural Geology*, 32(11):1557–1575, 2010. 378  
379  
380  
381
- [11] O. Fuks and H. A. Tchelepi. Limitations of physics informed machine learning for nonlinear two-phase transport in porous media. *Journal of Machine Learning for Modeling and Computing*, 1(1), 2020. 382  
383  
384
- [12] S. Goswami, M. Yin, Y. Yu, and G. E. Karniadakis. A physics-informed variational deeponet for predicting crack path in quasi-brittle materials. *Computer Methods in Applied Mechanics and Engineering*, 391:114587, 2022. 385  
386  
387
- [13] S. Goswami, A. Bora, Y. Yu, and G. E. Karniadakis. Physics-informed deep neural operator networks. In *Machine learning in modeling and simulation: methods and applications*, pages 219–254. Springer, 2023. 388  
389  
390

- [14] E. Haghigiat, M. Raissi, A. Moure, H. Gomez, and R. Juanes. A physics-informed deep learning framework for inversion and surrogate modeling in solid mechanics. *Computer Methods in Applied Mechanics and Engineering*, 379:113741, 2021. 391  
392  
393
- [15] E. Haghigiat, D. Amini, and R. Juanes. Physics-informed neural network simulation of multiphase poroelasticity using stress-split sequential training. *Computer Methods in Applied Mechanics and Engineering*, 397:115141, 2022. 394  
395  
396
- [16] E. Haghigiat, U. bin Waheed, and G. Karniadakis. En-deeponet: An enrichment approach for enhancing the expressivity of neural operators with applications to seismology. *Computer Methods in Applied Mechanics and Engineering*, 420:116681, 2024. 397  
398  
399  
400
- [17] Y. Han, F. P. Hamon, S. Jiang, and L. J. Durlofsky. Surrogate model for geological co<sub>2</sub> storage and its use in hierarchical mcmc history matching. *Advances in Water Resources*, 187:104678, 2024. 401  
402  
403
- [18] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 404  
405  
406
- [19] N. Hosseini, Z. Bajalan, and A. Khoei. Numerical modeling of density-driven solute transport in fractured porous media with the extended finite element method. *Advances in Water Resources*, 136:103453, 2020. 407  
408  
409
- [20] A. D. Jagtap, E. Kharazmi, and G. E. Karniadakis. Conservative physics-informed neural networks on discrete domains for conservation laws: Applications to forward and inverse problems. *Computer Methods in Applied Mechanics and Engineering*, 365:113028, 2020. 410  
411  
412  
413
- [21] J. Jiménez-Martínez, A. Alcolea, J. A. Straubhaar, and P. Renard. Impact of phases distribution on mixing and reactions in unsaturated porous media. *Advances in Water Resources*, 144:103697, 2020. 414  
415  
416
- [22] R. Juanes, J. Samper, and J. Molinero. A general and efficient formulation of fractures and boundary conditions in the finite element method. *International Journal for Numerical Methods in Engineering*, 54(12):1751–1774, 2002. 417  
418  
419
- [23] G. E. Karniadakis, I. G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, and L. Yang. Physics-informed machine learning. *Nature Reviews Physics*, 3(6):422–440, 2021. 420  
421
- [24] E. Kharazmi, Z. Zhang, and G. E. Karniadakis. hp-vpinns: Variational physics-informed neural networks with domain decomposition. *Computer Methods in Applied Mechanics and Engineering*, 374:113547, 2021. 422  
423  
424

- [25] A. Khoei, N. Hosseini, and T. Mohammadnejad. Numerical modeling of two-phase fluid flow in deformable fractured porous media using the extended finite element method and an equivalent continuum model. *Advances in water resources*, 94:510–528, 2016. 425  
426  
427  
428
- [26] A. Khoei, S. Mousavi, and N. Hosseini. Modeling density-driven flow and solute transport in heterogeneous reservoirs with micro/macro fractures. *Advances in Water Resources*, 182:104571, 2023. 429  
430  
431
- [27] A. R. Khoei and M. Taghvaei. A computational dual-porosity approach for the coupled hydro-mechanical analysis of fractured porous media. *International Journal for Numerical and Analytical Methods in Geomechanics*, 2024. 432  
433  
434
- [28] D. P. Kingma. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 435  
436
- [29] N. Kovachki, Z. Li, B. Liu, K. Azizzadenesheli, K. Bhattacharya, A. Stuart, and A. Anandkumar. Neural operator: Learning maps between function spaces with applications to pdes. *Journal of Machine Learning Research*, 24(89):1–97, 2023. 437  
438  
439
- [30] N. B. Kovachki, S. Lanthaler, and A. M. Stuart. Operator learning: Algorithms and analysis. *arXiv preprint arXiv:2402.15715*, 2024. 440  
441
- [31] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *nature*, 521(7553):436–444, 2015. 442  
443
- [32] Z. Li, N. Kovachki, K. Azizzadenesheli, B. Liu, K. Bhattacharya, A. Stuart, and A. Anandkumar. Neural operator: Graph kernel network for partial differential equations. *arXiv preprint arXiv:2003.03485*, 2020. 444  
445  
446
- [33] L. Lu, P. Jin, G. Pang, Z. Zhang, and G. E. Karniadakis. Learning nonlinear operators via deeponet based on the universal approximation theorem of operators. *Nature machine intelligence*, 3(3):218–229, 2021. 447  
448  
449
- [34] J. Molinero, J. Samper, and R. Juanes. Numerical modeling of the transient hydrogeological response produced by tunnel construction in fractured bedrocks. *Engineering Geology*, 64(4):369–386, 2002. 450  
451  
452
- [35] N. J. Moore, J. Cherry, S.-H. Chiu, and B. D. Quaife. How fluid-mechanical erosion creates anisotropic porous media. *Physica D: Nonlinear Phenomena*, 445:133634, 2023. 453  
454  
455
- [36] J. L. Musuuza, S. Attinger, and F. A. Radu. An extended stability criterion for density-driven flows in homogeneous porous media. *Advances in water resources*, 32(6):796–808, 2009. 456  
457  
458

- [37] S. A. Niaki, E. Haghigheh, T. Campbell, A. Poursartip, and R. Vaziri. Physics-informed neural network for modelling the thermochemical curing process of composite-tool systems during manufacture. *Computer Methods in Applied Mechanics and Engineering*, 384:113959, 2021.
- [38] M. Oda. An equivalent continuum model for coupled stress and fluid flow analysis in jointed rock masses. *Water resources research*, 22(13):1845–1856, 1986.
- [39] B. D. Quaife and M. N. J. Moore. A boundary-integral framework to simulate viscous erosion of a porous medium. *Journal of Computational Physics*, 375:1–21, 2018.
- [40] M. Raissi, P. Perdikaris, and G. E. Karniadakis. 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, 2019.
- [41] S. Saeedmonir, M. Adeli, and A. Khoei. A multiscale approach in modeling of chemically reactive porous media. *Computers and Geotechnics*, 165:105818, 2024.
- [42] A. Saeibehrouzi, R. Holtzman, P. Denissenko, and S. Abolfathi. Solute transport in unsaturated porous media with spatially correlated disorder. *Advances in Water Resources*, 191:104773, 2024.
- [43] A. Saeibehrouzi, S. Abolfathi, P. Denissenko, and R. Holtzman. Non-monotonic impact of erosion on solute dispersion in porous media. *Geophysical Research Letters*, 52(4):e2024GL114312, 2025.
- [44] M. Sahimi. *Flow and transport in porous media and fractured rock: from classical methods to modern approaches*. John Wiley & Sons, 2011.
- [45] R. A. Schincariol, F. W. Schwartz, and C. A. Mendoza. Instabilities in variable density flows: Stability and sensitivity analyses for homogeneous and heterogeneous media. *Water resources research*, 33(1):31–41, 1997.
- [46] M. L. Sebben, A. D. Werner, and T. Graf. Seawater intrusion in fractured coastal aquifers: A preliminary numerical investigation using a fractured henry problem. *Advances in water resources*, 85:93–108, 2015.
- [47] A. Vaswani. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017.
- [48] N. Wang, H. Chang, and D. Zhang. Inverse modeling for subsurface flow based on deep learning surrogates and active learning strategies. *Water Resources Research*, 59(7):e2022WR033644, 2023.
- [49] S. Wang, H. Wang, and P. Perdikaris. Learning the solution operator of parametric partial differential equations with physics-informed deeponets. *Science advances*, 7(40):eabi8605, 2021.

- [50] G. Wen, Z. Li, K. Azizzadenesheli, A. Anandkumar, and S. M. Benson. U-fno—an enhanced fourier neural operator-based deep-learning model for multiphase flow. *Advances in Water Resources*, 163:104180, 2022. 494  
495  
496
- [51] G. Wen, Z. Li, Q. Long, K. Azizzadenesheli, A. Anandkumar, and S. M. Benson. Real-time high-resolution co 2 geological storage prediction using nested fourier neural operators. *Energy & Environmental Science*, 16(4):1732–1741, 2023. 497  
498  
499
- [52] J. Xu, H. Wei, and H. Bao. Physics-informed neural networks for studying heat transfer in porous media. *International Journal of Heat and Mass Transfer*, 217: 124671, 2023. 500  
501  
502
- [53] X. Yan, J. Lin, S. Wang, Z. Zhang, P. Liu, S. Sun, J. Yao, and K. Zhang. Physics-informed neural network simulation of two-phase flow in heterogeneous and fractured porous media. *Advances in Water Resources*, 189:104731, 2024. 503  
504  
505
- [54] R. K. Zeytounian. Joseph boussinesq and his approximation: a contemporary view. *Comptes Rendus Mecanique*, 331(8):575–586, 2003. 506  
507
- [55] Z. Zhang, X. Yan, P. Liu, K. Zhang, R. Han, and S. Wang. A physics-informed convolutional neural network for the simulation and prediction of two-phase darcy flows in heterogeneous porous media. *Journal of Computational Physics*, 477:111919, 2023. 508  
509  
510  
511
- [56] C. B. Zhou, R. S. Sharma, Y. F. Chen, and G. Rong. Flow–stress coupled permeability tensor for fractured rock masses. *International Journal for Numerical and Analytical Methods in Geomechanics*, 32(11):1289–1309, 2008. 512  
513  
514
- [57] L. Zhuang, A. Raoof, M. G. Mahmoodlu, S. Biekart, R. de Witte, L. Badi, M. T. van Genuchten, and K. Lin. Unsaturated flow effects on solute transport in porous media. *Journal of Hydrology*, 598:126301, 2021. 515  
516  
517