Accelerating Multiphase Simulations With Denoising Diffusion Model Driven Initializations

JGR | Machine Learning and Computation

RESEARCH ARTICLE
10.1029/2024JH000293

Special Collection:
Advancing Interpretable AI/ML Methods for Deeper Insights and Mechanistic Understanding in Earth Sciences: Beyond Predictive Capabilities

Accelerating Multiphase Simulations With Denoising Diffusion Model Driven Initializations

Jaehong Chung[1,2], Agnese Marcato[2], Eric J. Guiltinan[2], Tapan Mukerji[1,3,4], Hari Viswanathan[2], Yen Ting Lin[5], and Javier E. Santos[2]

[1]Department of Geophysics, Stanford University, Stanford, CA, USA, [2]Earth and Environmental Sciences Division, Los Alamos National Laboratory, Computational Earth Science Group (EES-16), Los Alamos, NM, USA, [3]Department of Energy Science and Engineering, Stanford University, Stanford, CA, USA, [4]Department of Earth and Planetary Science, Stanford University, Stanford, CA, USA, [5]Computer Computational and Statistical Science Division, Los Alamos National Laboratory, Information Science Group (CCS-3), Los Alamos, NM, USA

Key Points:

- We introduce a method that integrates generative AI with multiphase simulations improving efficiency while maintaining physical accuracy
- We propose simulation metrics to evaluate model performance, ensuring alignment with the targeted physical phenomena
- We demonstrate a significant speed-up over traditional initialization methods in synthetic and real (x-ray) fracture data sets

Correspondence to:
J. Chung,
jhchung1@stanford.edu

Citation:
Chung, J., Marcato, A., Guiltinan, E. J., Mukerji, T., Viswanathan, H., Lin, Y. T., & Santos, J. E. (2024). Accelerating multiphase simulations with denoising diffusion model driven initializations. *Journal of Geophysical Research: Machine Learning and Computation*, *1*, e2024JH000293. https://doi.org/10.1029/2024JH000293

Received 27 JUN 2024
Accepted 4 NOV 2024

Author Contributions:
Conceptualization: Yen Ting Lin, Javier E. Santos
Data curation: Jaehong Chung, Agnese Marcato, Eric J. Guiltinan
Funding acquisition: Javier E. Santos
Investigation: Jaehong Chung, Yen Ting Lin, Javier E. Santos

Abstract This study introduces a hybrid fluid simulation approach that integrates generative diffusion models with physics-based simulations, aiming at reducing the computational costs of flow simulations while still honoring all the physical properties of interest. Pore-scale simulations enhance our understanding of applications such as assessing hydrogen and CO_2 storage efficiency in underground reservoirs. Nevertheless, they are computationally expensive and the presence of non-unique solutions can require multiple simulations within a single geometry. To overcome the computational cost hurdle, we propose a method that couples generative diffusion models and physics-based simulations. While training the data-driven model, we simultaneously generate initial conditions and perform physics-based simulations using these. This integrated approach enables us to receive real-time feedback on a single compute node equipped with both CPUs and GPUs. By efficiently managing these processes within a single compute node, we can continuously monitor performance and halt training once the model meets the specified criteria. To test our model, we generate realizations in a real Berea sandstone fracture which shows that our technique is up to 4.4 times faster than commonly used flow simulation initializations.

Plain Language Summary Pore-scale simulations enhance our understanding of how fluid moves in the subsurface, they have recently been used for evaluating hydrogen storage and carbon dioxide sequestration projects in underground reservoirs. However, these simulations are often limited by their high demand for computational resources, posing a challenge to efficient execution. To address this, we propose a method that integrates artificial intelligence (AI) with physics-based simulators. After training our AI model with synthetic data, we tested our approach using a real reservoir sample obtained through X-ray imaging. This allowed us to showcase the computational efficiency of our method.

1. Introduction

CO_2 and H_2 subsurface storage are seen as important methods to address climate change (Dai et al., 2016; Krevor et al., 2023). The injection of CO_2 and H_2 is a multiphase process whereby these fluids displace existing groundwater. Predicting the behavior of these two-phase fluid systems is challenging, as their dynamics are governed by diverse factors including flow path geometries (Blunt, 2001), fluid saturations (Parker, 1989), and affinity between the fluids and the host rocks (Fatt & Klikoff Jr, 1959). In addition, the presence of fractures, which act as preferential pathways for flow, further complicates the picture (Sweeney et al., 2023; Viswanathan et al., 2022). These fractures are potential risks for fluid leakage (Tongwa et al., 2013), underlining the need for a comprehensive understanding to ensure the integrity and efficiency of underground storage systems (Fitts & Peters, 2013; E. J. Guiltinan et al., 2021; Ting et al., 2022).

Pore-scale simulations, ranging from 10 nm to 10 cm, provide accurate models of how fluids travel through porous media in underground reservoirs (Middleton et al., 2012). In particular, these micro-scale simulations can capture the preferred traveling paths for fluids by identifying local minimum energy states (Blunt et al., 2013). Such configurations arise from the energy balance between fluid displacements and interfacial tensions between fluids and solids (Blunt, 2017). While the pore-scale simulations can characterize accurately and precisely fluid configurations in the pore space, scaling these simulations to the reservoir scale, ranging from 10 cm to 100 m, is impractical due to the expensive computational costs associated with large modeling domains with intricate flow

Methodology: Yen Ting Lin, Javier E. Santos
Validation: Jaehong Chung
Visualization: Jaehong Chung, Agnese Marcato
Writing – original draft: Jaehong Chung
Writing – review & editing: Agnese Marcato, Eric J. Guiltinan, Tapan Mukerji, Hari Viswanathan, Yen Ting Lin, Javier E. Santos

paths (complicated geometries). In addition, the need to run multiple simulations for different saturation levels in the same geometries (e.g., to compute relative permeability curves) further escalates the computational challenges.

Many studies employ machine learning techniques for predicting fluid flow at the macroscale (Yan, Harp, Chen, Hoteit, & Pawar, 2022; Yan, Harp, Chen, & Pawar, 2022). However, applications at the pore scale remain limited. Recent advancements in machine learning have demonstrated significant potential in reducing the computational burden of multiphase flow simulations. Machine learning can help these simulations by offering efficient algorithms that can learn complex patterns and relationships within the data. This capability allows for faster and more accurate predictions without the need for solving extensive and computationally expensive equations traditionally required in multiphase flow modeling. Additionally, machine learning models can be trained on large data sets to generalize well to new, unseen data, thereby enhancing the overall efficiency and scalability of the simulations (J. E. Santos et al., 2024).

Building on these advancements, several recent studies have explored the application of machine learning techniques to multiphase flow problems with encouraging results (Zhao et al., 2023). employed a U-Net architecture to predict fluid displacement in micromodels showing high accuracy in the predicted displacement configurations (Ting et al., 2022). trained a residual U-Net to predict two-phase configurations in fractures, mapping the 3D geometry into a 2D feature space with a lossless algorithm. Their trained model was able to predict the invasion dynamic of an unsteady state flow of a non-wetting front (E. Guiltinan et al., 2020). trained a network to predict the fluid distribution within fractures at steady state using data from lattice Boltzmann simulations. They demonstrated that a trained network can accurately predict fluid residual saturation and distribution based solely on the dry fracture characteristics. Z. Wang et al. (2022) trained a conditional Generative Adversarial Network to predict invasion percolation configurations in 2D spherepacks. While the model demonstrated a reduction in computational time compared to traditional algorithms, its use cases are still limited as these simulations do not capture the full complexity of two-phase fluid interactions. These studies provide an initial demonstration of successful applications of machine learning for multiphase flow problems.

Hybrid approaches, which combine the computational efficiency of deep learning with the physical robustness of numerical simulations, have been proposed as a solution to simulate complex physical problems efficiently. Y. D. Wang et al. (2021) introduced an integrated framework that employed a Convolutional Neural Network (CNN) to provide a data-driven initialization, subsequently utilized in single-phase Lattice Boltzmann Method (LBM) simulations. This approach achieved a tenfold acceleration of the simulation process while maintaining physical accuracy. In addition, Chang et al. (2022) demonstrated that machine learning-based initializations can greatly accelerate simulations of electrical conductivity in porous media. These studies show the potential of a hybrid approach to combine the strengths of deep learning and numerical solvers. Our study aims to extend this hybrid approach to multiphase flow simulations, which are orders of magnitude more expensive compared to single phase and electrical conductivity simulations. Hybrid approaches tackle the challenge of achieving computational efficiency while ensuring compliance with fundamental governing equations, such as continuity and momentum balance.

In deep learning, generative models are designed to approximate high-dimensional data distributions (Goodfellow et al., 2016). Recently, diffusion models have gained prominence for their unique ability to progressively convert noise into highly detailed and structured outputs that approximate the distribution of the training data. These models employ a reverse diffusion process to gradually build coherent structures from noise, and a forward diffusion process to incrementally corrupt the data. This bidirectional approach makes diffusion models particularly effective for complex computer vision tasks, such as generating high-resolution, realistic images (Ho et al., 2020; Sohl-Dickstein et al., 2015), establishing a new state-of-the-art for generative modeling.

In this study, we demonstrate that diffusion models can learn to accurately predict steady-state multiphase fluid configurations in fracture geometries. The fluid configurations represent the distribution of the fluid phases within the computational domain. We introduce a geometric conditioning approach to obtain fluid configurations for specific fracture geometries. The predicted fluid configurations approximate the ground-truth, but this data-driven model does not guarantee the continuity and momentum balances, as hard constraints have not been enforced. Using the prediction of a data-driven model to initialize a LBM simulation takes advantage of the fast prediction capabilities of the data-driven model and also ensures the continuity and momentum balance with the simulator. We quantify accuracy using both statistical metrics, such as mean squared error, and by measuring the reduction

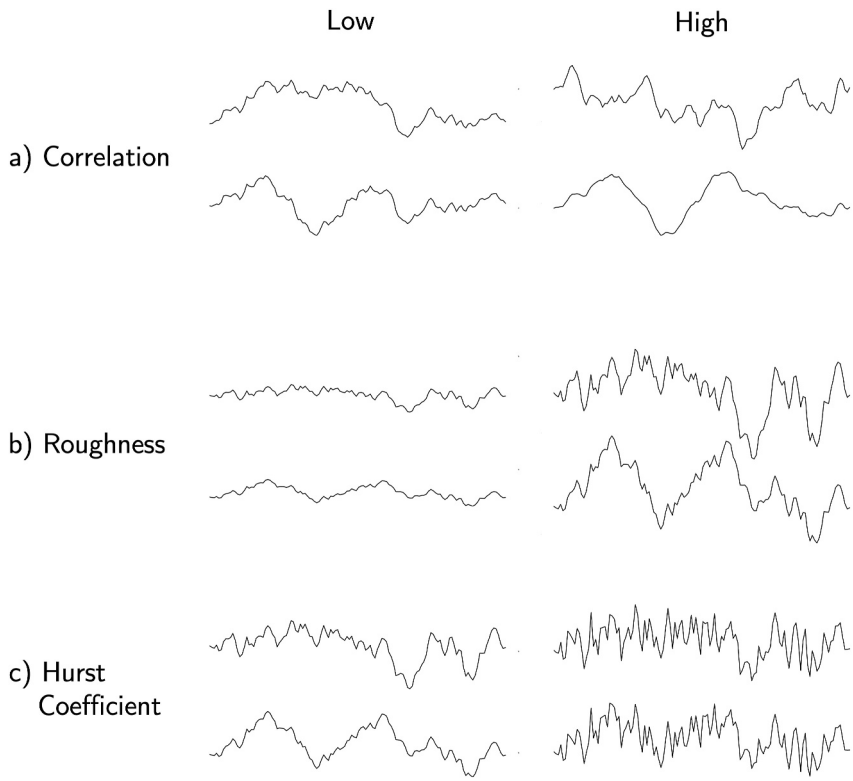

Figure 1. Impact of geometric parameters on fracture characteristics: (a) Increasing spatial correlation in the flow direction, (b) Pronounced peaks and valleys with increased roughness, (c) Higher Hurst coefficient reduces local correlation, making individual asperities more pronounced while maintaining constant overall mean aperture.

in simulation time, indicating how closely the predicted configuration matches the real one. Our results demonstrate that diffusion models can provide accurate configurations for multiphase flow, significantly enhancing the efficiency of physics-based simulations. This approach opens the door to hybrid approaches where machine learning and physics-based methods can coexist, leveraging the strengths of both to solve complex problems more effectively.

The remainder of this paper is organized as follows: Section 2 provides detailed information about the fracture data set and multiphase simulations used in this study. Section 3 describes the diffusion model used and details the geometric conditioning process for specific fracture geometries. Finally, Section 4 presents and discusses the results.

2. Data Set

2.1. Fracture Geometries

To train our model, we generated a data set of 1,000 fractures using `pySimFrac`, an open-source Python-based fracture geometry generator (E. Guiltinan et al., 2024). We used the spectral method to create fractures by varying the following geometric parameters: Hurst exponent, spatial correlation, and surface roughness. The impact of these parameters is illustrated in Figure 1.

Although these fractures are synthetic, the `pySimFrac` implementation and values used align with those measured in profilometry studies of real fractures (Brown, 1995). For readers interested in the underlying methods and their implementations, further details can be found in the literature (Glover et al., 1998; Ogilvie et al., 2006; E. Guiltinan et al., 2024).

The generated geometries are projected onto a 128-by-128 grid, enabling us to run numerous lattice Boltzmann simulations within a computationally feasible time frame. This approach enables us to create a sufficiently diverse data set for training machine learning models. To visualize the diversity within our data set, we employed t-SNE

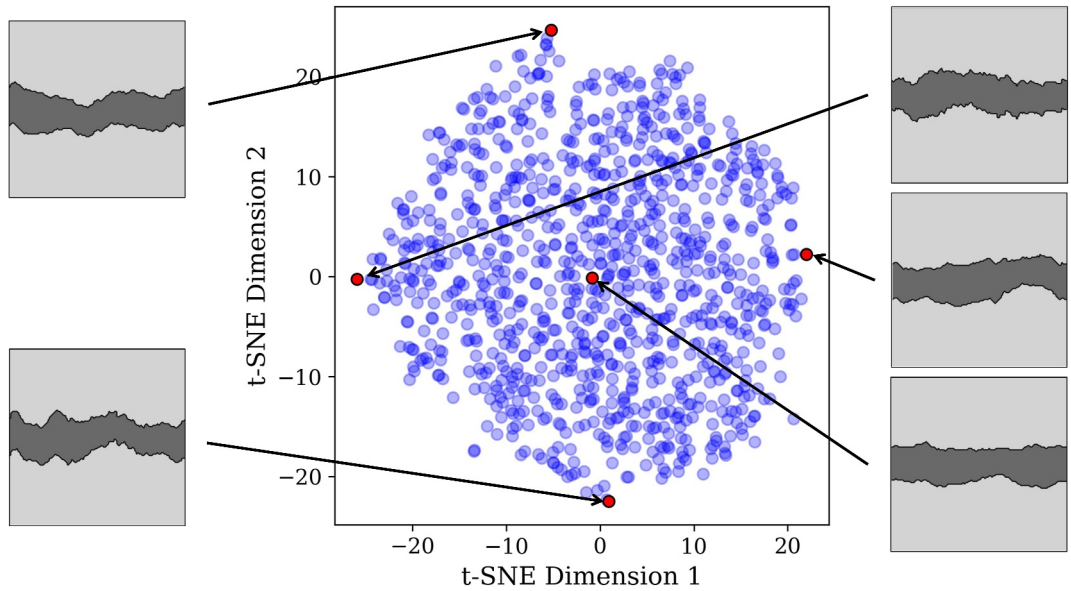

Figure 2. This t-SNE visualization reduces our synthetic rough fracture data set into a two-dimensional space, condensing the original fracture geometry (16,384 pixels) into two-dimensions (Dimension 1 and Dimension 2). The display reveals a continuous spectrum of data rather than isolated groups or clusters, indicating a diverse yet evenly distributed data set. Several sets of hyperparameters were tested, consistently resulting in a similar distribution of points without significant clustering. We highlight a few examples to showcase the range of fractures in the data set.

(t-Distributed Stochastic Neighbor Embedding), a technique for reducing dimensionality and visualizing high-dimensional data structures in a lower-dimensional space (Van der Maaten & Hinton, 2008). Figure 2 shows that while differences exist within our data set, the extremes of the t-SNE plot are relatively limited. The goal of this work is to demonstrate the ability of the machine learning model to generalize to a variety of fractures despite the simplicity of its training set.

2.2. Multiphase Flow Simulations

We conducted two-phase flow simulations on the fractures geometries using the MP-LBM library (J. E. Santos et al., 2022), which employs the lattice Boltzmann method (LBM) to simulate the behavior of two competing immiscible fluids (wetting and non-wetting) in complex geometries. This codebase has undergone extensive benchmarking, including numerical validations with the Washburn equation, the Young-Laplace equation, and comparisons with the analytical solutions of single-phase fluid flow through a slit and a tube.

2.2.1. Numerical Setup and Simulation Details

To ensure clarity for readers who may not be well-versed in LBM, we provide detailed information on the numerical setup used in our simulations.

Simulation Domain: The area of interest in our simulations is the horizontal fracture channel within the 128×128 grid. The entire computational domain serves to represent the fracture geometry; however, the primary focus is on the fluid interactions within the horizontal channel where flow occurs.

Initial Conditions: The initial conditions for the simulations involve placing the two fluids (brine and supercritical CO_2) in specified ratios within the fracture geometry. The initial distribution of these fluids is assigned (via the different methods explained in Section 4.2) across the void space within the fracture domain. The velocity distribution of both fluids is initially set to zero.

Boundary Conditions: The following boundary conditions are applied in the simulations:

- *Periodicity:* This is applied along the horizontal (flow) direction of the fracture, ensuring that the fluid exiting one side of the domain re-enters from the opposite side. This setup allows the simulation to reach a steady-state.

- *No-Slip:* This is applied to the solid boundaries of the fractures, representing the impermeable rock matrix that surrounds the fluid.

Simulation Equations and Quantities: The core simulation equation is the Lattice Boltzmann Equation with the Bhatnagar-Gross-Krook (BGK) collision term (Bhatnagar et al., 1954), which governs the evolution of the density distribution function of each fluid phase:

$$f_a^{\alpha}(\boldsymbol{x}+\boldsymbol{e}_a\Delta t, t+\Delta t) = f_a^{\alpha}(\boldsymbol{x},t) - \frac{1}{\tau}\left[f_a^{\alpha}(\boldsymbol{x},t) - f_a^{\alpha,eq}(\boldsymbol{x},t)\right], \tag{1}$$

where $f_a^{\alpha}(\boldsymbol{x},t)$ is the density function with the ath lattice direction, α represents components for two phase systems $\alpha = 1$ or -1, $\boldsymbol{e}_a$ is the set of discrete velocity vectors of a node, and τ is characteristic relaxation time, related to the fluid viscosity. The equilibrium distribution function, $f_a^{\alpha,eq}(\boldsymbol{x},t)$, describes the distribution function in a state of local equilibrium (Chen et al., 1992). In addition, we consider the acting force on each fluid phase due to the fluid-fluid interaction (interfacial tension) and fluid-solid interaction (wettability) based on the Shan-Chen model (Shan & Chen, 1993). Readers interested in the underlying model for specifying a specific contact angle can refer to the literature (Huang et al., 2007). The key quantities simulated include fluid density, velocity fields, and the evolution of fluid-fluid interfaces, driven by interfacial tension and wettability effects.

Monitored Quantities: The primary quantities monitored during and after the simulations are:

- *Relative Kinetic Energy:* Monitored throughout the simulation to determine convergence.
- *Fluid positions:* These spatial configurations, achieved upon convergence, represent the final distribution of the two fluids within the fracture, indicating the most conductive pathways of the domain.

Fluid and Solid Properties: The simulations were conducted using a wetting angle of 24°, representing a system with brine and supercritical CO_2. It's worth noting that the proposed workflow is capable of learning systems with different fluid properties, it is not limited to our specific system. For each fracture, we ran 13 evenly distributed simulations with fluid ratios varying from 20% to 60% to ensure percolation of both fluids and to exclude isolated blobs that would not converge in the simulation. The fluid ratio is defined as the ratio of the volume of the non-wetting phase (supercritical CO_2) to the volume of the wetting phase (brine). This resulted in a total of 13,000 cases (1,000 synthetic fractures × 13 saturation cases) for the fluid configuration data set in this study.

Steady-steate criterion: During the simulation, both fluids are driven through the fracture by an external force. This body force and the periodic boundary conditions at the inlet and outlet allows us to achieve a steady-state configuration of the two phases. Convergence is monitored by tracking the relative kinetic energy $(rel.\ E_k)$, which is calculated using the following equation (Latt et al., 2021):

$$rel.\ E_k = \frac{1}{2}\left\| \partial \frac{\vec{v}}{\partial t} \right\|^2, \tag{2}$$

where $\partial\vec{v}/\partial t$ denotes the time rate of change of the velocity vector. The simulation is considered converged once the relative energy change across the simulation domain falls below a threshold of 5×10^{-5} between consecutive 1,000 time steps. The simulations take on average around 100 thousand iterations in time (time-steps) to achieve convergence. Once the simulation converges, the fracture's fluid configurations can be considered stable, implying a minimum free energy state. In other words, the converged simulation depicts the most conductive pathways for each sample given the solid geometry and the initial ratios of fluid. It is noteworthy that these stable fluid configurations exhibit intricate variability, even within the same fracture geometry.

In this work, the LBM simulations act as the ground-truth to train our model. Nevertheless, we would like to emphasize that the workflow described above is not limited to LBM simulations, it can also be applied using other methods such as level-set simulations (Jettestuen et al., 2013), molecular dynamics (J. E. Santos et al., 2020), and computational fluid dynamics (Bouras et al., 2022), among others. We chose LBM simulations because it is practical for us, given that we already have a robust open-source solver in place.

10.1029/2024JH000293

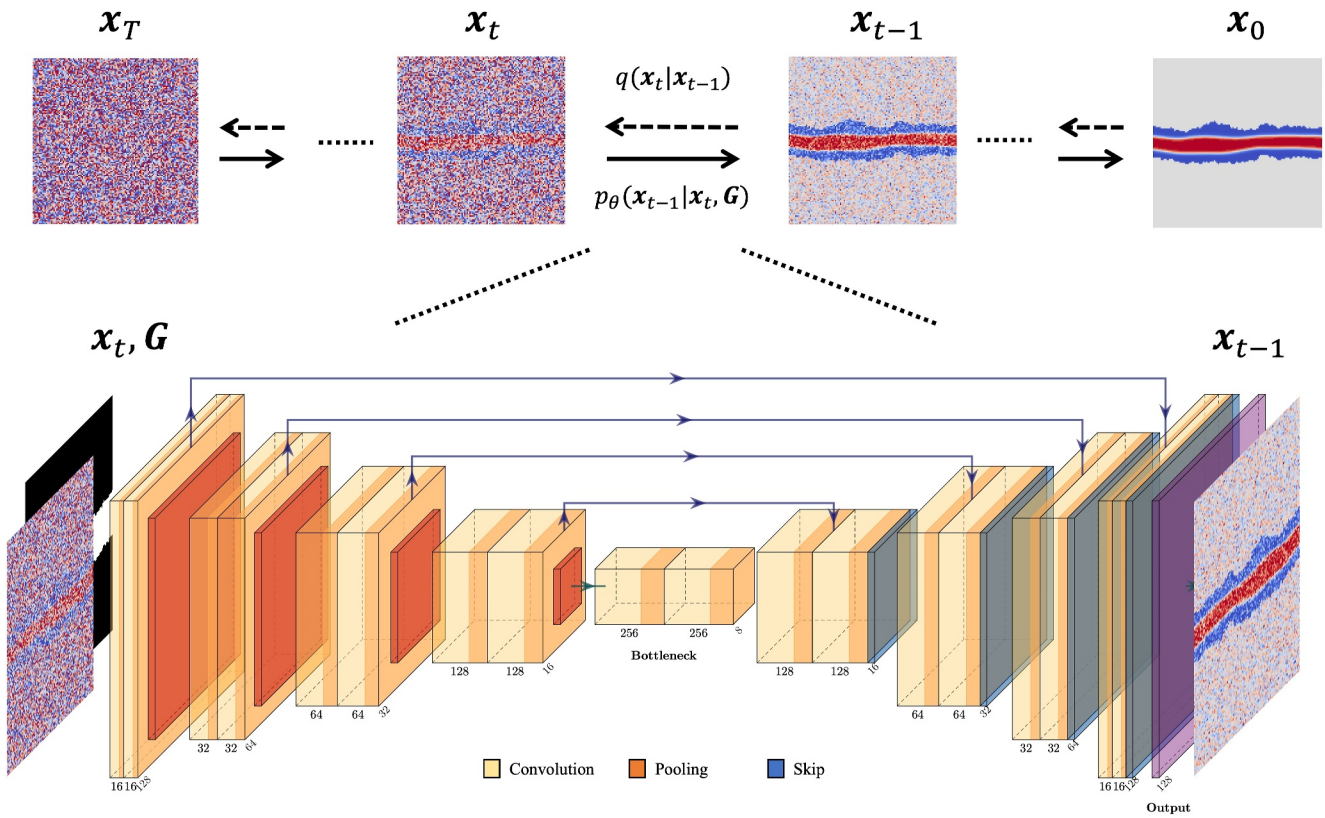

Figure 3. Diagram depicting the diffusion process and the fracture geometry conditioning to guide the process to produce fluid configurations within the predefined pore-space.

3. Diffusion Models Conditioned on Fracture Geometries

We utilize Denoising Diffusion Probabilistic Models (DDPMs) (Ho et al., 2020) to model and sample the data distribution of stable fluid configurations in fractured media. These models work by constructing two Markov Chains: a forward process that gradually adds noise to the data, and a reverse process that reconstructs the original data from this noisy state. Unlike models like Generative Adversarial Networks (one-step neural network mapping trained adversarially with a discriminator), Diffusion Models are easier to train, more stable, and produce higher-quality features.

In the forward process, samples drawn from the data distribution $\mathbf{x_0} \sim q(\mathbf{x_0})$ are progressively transformed into indistinguishable noise, ultimately approaching a simple Gaussian distribution. The reverse process, on the other hand, seeks to undo this transformation. During training, a U-Net (Ronneberger et al., 2015) is trained, and once this process is done, the network can generate new samples starting with noise and gradually converting it back into structured data that resembles the original distribution. This is particularly useful because it allows us to start with a fully noisy field (which is easy to sample) and end up with a realistic representation of a stable fluid configuration. A graphical representation of this workflow is shown in Figure 3.

In summary, the reverse diffusion process reconstructs data from noise, while the forward process systematically degrades it. To train the model, we use Variational Inference by minimizing the Evidence Lower Bound, which effectively increases the likelihood that the model accurately represents the training data (Ho et al., 2020; Sohl-Dickstein et al., 2015). The model is trained to predict the noise, ϵ, added to $\boldsymbol{x}_t$, based on the following loss function:

$$\mathcal{L}(\theta) = \mathbb{E}_{t\sim[1,T],\boldsymbol{x}_0,\boldsymbol{\epsilon}}\left[\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t)\|^2\right], \tag{3}$$

which measures the difference between the actual noise and the model's prediction ϵ_θ at a given step t, helping the model learn to accurately reconstruct the original data distribution.

3.1. Fracture Geometry Conditioning

The geometry of the solid boundaries has a first order influence on the configurations of fluids (Blunt, 2017), therefore, a model should be able to provide fluid configurations for user-specified geometries. This task is similar to inpainting in computer vision (Saharia et al., 2022; Yeh et al., 2017; Yu et al., 2018), which requires the model to synthesize images in unknown areas that are both realistic and consistent with the surrounding background. One unique challenge in our inpainting task lies in the fracture geometries which are characterized by sharp and irregular shapes. To ensure that our model generates fluid configurations that adhere to given geometries (i.e., not solely based on random Gaussian noise), we introduce an auxiliary binary image $\boldsymbol{G} \in \{0,1\}^{H\times W}$ encoding the fracture geometry. Here, H is the height and W is the width of the image:

$$\boldsymbol{G}(i,j) = \begin{cases} 1 & \text{if pixel at } (i,j) \text{ is part of the solid,} \\ 0 & \text{otherwise.} \end{cases} \tag{4}$$

Figure 3 shows a schematic diagram of the diffusion process in our model, which generates denoised images $(\boldsymbol{x}_{t-1})$ given $\boldsymbol{x}_t$ and $\boldsymbol{G}$. Including this additional channel at each denoising step provides the model with the information to condition its realization on the given pore-scale geometry. Once trained, the model can generate fluid realizations for unseen geometries by concatenating the desired fracture geometry, represented as a binary image $(\boldsymbol{G})$, with a random isotropic Gaussian field $(\boldsymbol{x}_T)$, and running the reverse process depicted in Figure 3.

During the model's development, we experimented with using only one channel, where the solid regions were depicted by zeros, and the noise and denoising processes occurred within the fracture space. This approach resulted in very poor outcomes. We hypothesize that in highly rough fractures with very small pore spaces, the limited amount of Gaussian noise added is insufficient to enable the necessary expressiveness for the diffusion process to accurately create fluid configurations.

4. Results and Discussion

4.1. Training and Model Performance

We train the diffusion model with 100 denoising steps over 4,000 training iterations, a process that takes approximately 1 hr on a single A100 GPU. Although research shows that increasing the number of denoising steps can improve quality and recover finer features (Nichol & Dhariwal, 2021), we limit ours to 100 for optimal efficiency. We found that this number strikes a balance between performance and inference speed. We use the cosine noise scheduler, as it has been shown to be the most suitable for processes like these (J. Santos & Lin, 2023).

During training, we monitored two metrics. First, the Mean Squared Error (MSE), which quantifies the discrepancy between predictions and ground truth. While a low MSE value indicates that the model is effectively approximating the training data, determining the optimal model for the physical domain remains challenging. Additionally, we introduce a metric that focuses on the number of iterations required for the LBM simulation to converge, starting from the initial configuration provided by the diffusion model. The rationale behind this metric is based on the functionality of numerical simulators, which approximate solutions to partial differential equations through iterative methods. Therefore, if the fluid configuration generated by our diffusion model closely approximates the numerical solution, the number of iterations needed for convergence should be minimal, as the starting point is already near the converged configuration.

Throughout the training, we generate fluid configurations for unseen fracture geometries in intervals of 800 training iterations (during which the model has seen 800 fractures with varying noise levels). These configurations serve as initial conditions for multiphase flow simulations, and we record the number of iterations required by our LBM solver to achieve convergence. Figure 4 illustrates the evolution of the MSE and the convergence iterations over the training period. Notably, the MSE starts high and plateaus around 2,000 training iterations, indicating effective denoising for the training data set. Additionally, we observe a significant reduction in the number of iterations required for convergence. Specifically, the average number of iterations decreased from 2.6

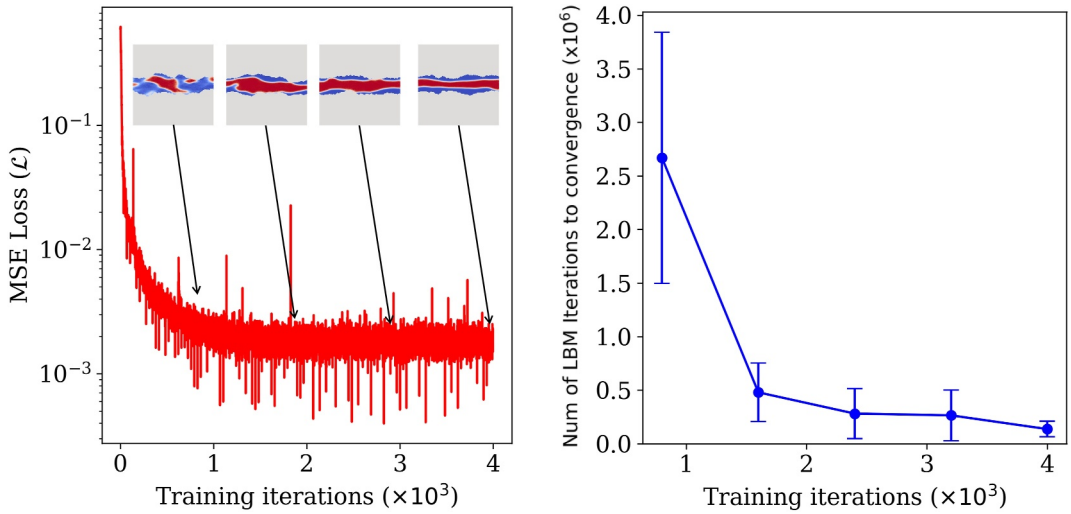

Figure 4. Evaluation metrics during diffusion model training: Left—Evolution of MSE across iterations, with highlighted samples at 1, 2, 3, and 4 K training iterations. Right—Variation in iterations required for convergence, showing the mean and standard deviation.

million to 140 thousand, representing a reduction of approximately 95%. The standard deviation also decreased by 95%, further highlighting the model's increasing consistency. These results suggest that our diffusion model not only captures the visual similarity of the training data set but also learns to generate physical configurations conditioned on the fracture geometry. It is important to note that computing this metric (and running the multiphase lattice Boltzmann simulations) does not incur additional computational overhead, as the simulations are run on the fly using the idle CPUs of our compute node. This approach maximizes the utilization of our computing infrastructure and allows for continuous monitoring of the model's performance without additional resource demands.

4.2. Assessing the Trained Model's Performance Against Different Initialization Methods

We observed that as the training progressed, the model was able to provide solutions very close to the stable configuration of the LBM solver for the training data, as shown in Figure 4. Upon completing the training, we explore the computational advantages of our hybrid approach in data not present in the training set. We generate initial configurations with the trained diffusion model for geometries unseen during training and then assess the number of iterations needed for convergence. The generated configurations for the two phases are used to initialize the fluid phases in the Lattice Boltzmann simulation, while the velocity is initialized to zero in the

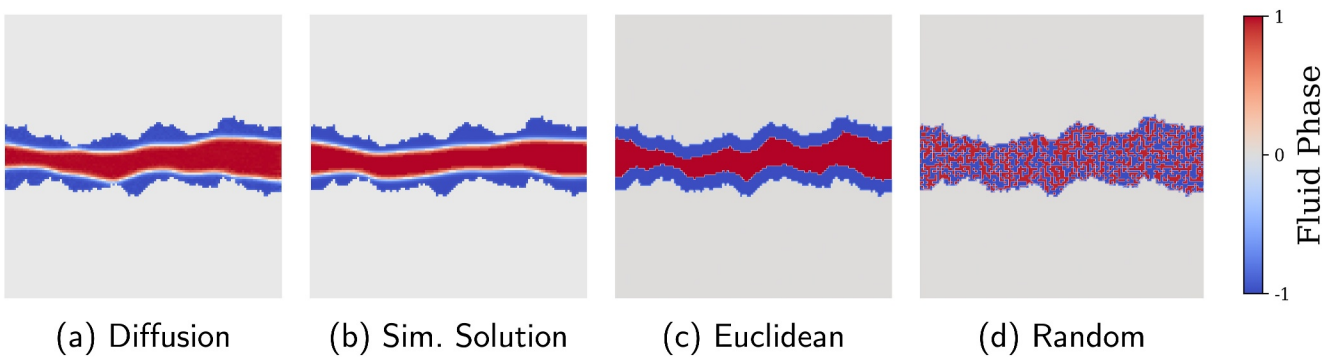

Figure 5. Comparison of fluid densities initialized by different methods within the simulation domain for multiphase simulations utilizing the Shan-Chen model. The Diffusion-Based Initialization synthesizes continuous values similar to the LBM solution. In contrast, the Euclidean Distance-Based and Random Initialization methods produce discrete phase representations. The color bar indicates a continuous range from −1 to 1, which is how the fluid phases are normalized for the machine learning workflow.

Table 1
Comparison of Average Number of Iterations and Simulation Time for Different Initialization Methods

Initialization method	Number of iterations	Simulation time (min)
Simulation solution	43,800	8.5
Generative diffusion	96,100	20.8
Euclidean initialization	123,500	26.7
Random initialization	225,900	46.0

computational domain. To validate our strategy, we benchmark our diffusion-based initialization against two commonly used fluid initialization methods: Euclidean distance-based initialization and random initialization. Additionally, we include a benchmark using the simulation solution as the starting point for a new simulation. Figure 5 presents examples of the different initializations within the simulation domain.

First, the *Diffusion-Based Initialization* method uses a realization from our trained model within a given geometry. Using the same wetting/non-wetting fluid ratio, the *Euclidean Distance-Based Initialization* identifies the non-wetting region in the flow path based on provided saturations and positions the non-wetting fluid away from the solid boundary. This is based on the understanding that the invading fluid tends to occupy pores distant from solid walls (Blunt, 2017). The *Random Initialization* method randomly places fluid particles throughout the connected domain until the desired saturation is reached. Finally, The *Simulation Solution-Based Initialization* uses converged fluid configurations from the LBM simulation as its baseline. It is important to understand that this method sets a benchmark that other initializations are not expected to exceed. We purposely do not use the velocity field saved from the previous simulation since our diffusion model is only trained to provide the fluid configuration, not the velocity field. Therefore, initializing a simulation with only the previously converged fluid flow and not the velocity field may still require several iterations to reconverge. This ensures fairness when comparing it with other methods.

Another advantage of the Diffusion-Based Initialization is the ability to generate continuous density values. Although we are mainly interested in defining the two phases distinctly, slight variations in their density values inform the LBM simulation about local capillary pressure gradients, among other physical features. In contrast, the Euclidean Distance-Based and Random Initialization methods can only produce discrete phases (see Figure 5). Thus, the simulation solution-based initialization method and the random-based initialization method represent the lower and upper bounds of convergence time, respectively, while the Euclidean-based initialization is a reasonable approach, grounded in the domain knowledge that invading fluid tends to occupy pores distant from solid walls (Blunt, 2017).

Table 1 provides a detailed comparison of 100 simulation results on geometries not seen during training. We compare each of the initialization methods described above.

The simulation solution initialization method, although it may seem somewhat impractical, as it essentially uses the solution of a previous simulation, represents the best possible scenario for initialization. By doing so, it sets a clear benchmark for the fastest possible convergence. This approach allows us to understand the upper limits of efficiency and provides a standard against which the performance of other initialization methods can be measured. On average the simulation took 4.38×10^4 iterations to convergence, an average simulation time of 8.5 min.

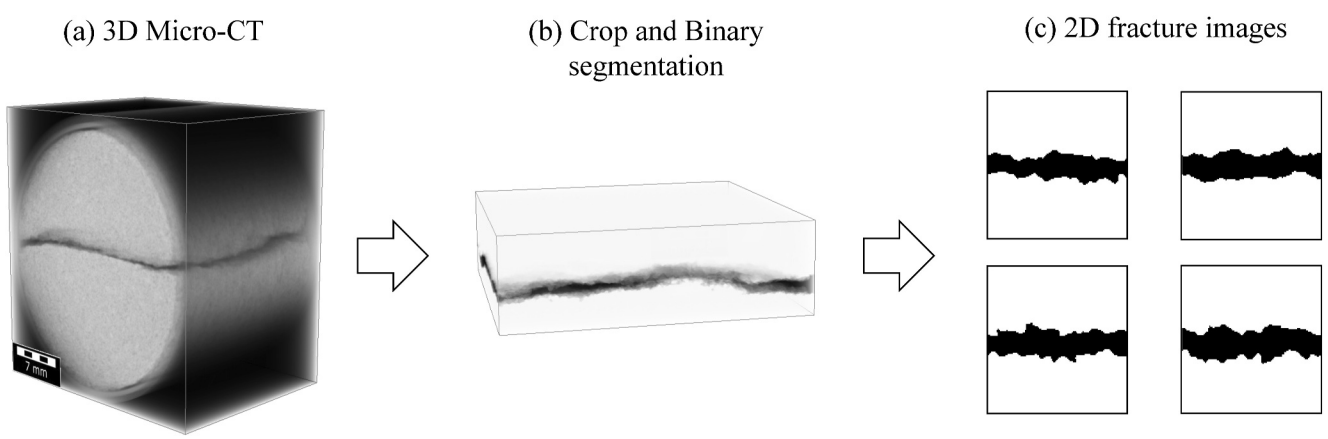

Figure 6. Workflow for extracting 2D fracture images from the micro-CT scans of Berea sandstone (Karpyn et al., 2016). The process includes (a) reading in the large micro-CT file, (b) cropping and performing a binary segmentation on the geometry, and (c) extracting 2D fracture slices.

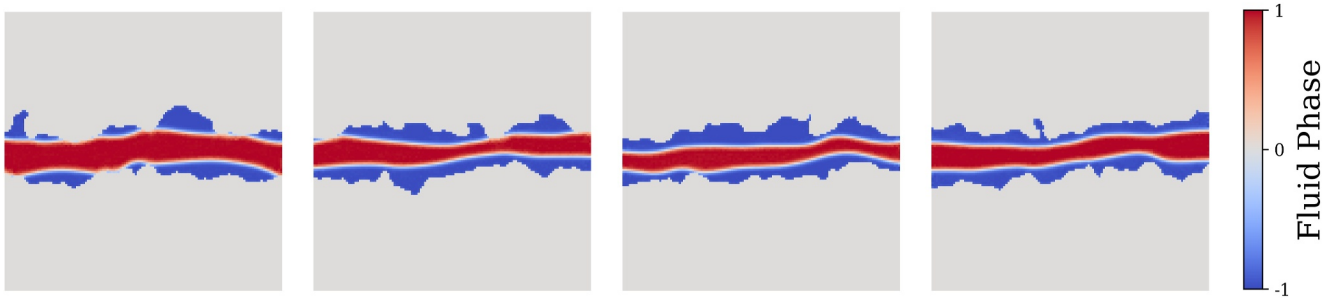

Figure 7. Examples of generated multiphase fluid configurations in 4 different Berea fracture cross-sections.

These values represent the potential lower bound of computational demand for the diffusion model under optimal conditions.

Once this benchmark is established, the diffusion-based initialization method shows an improvement in efficiency compared to other common multiphase simulation initialization methods. On average, the simulations using diffusion-based initialization take 9.61×10^4 iterations, roughly 20.8 min. The diffusion-based initialization reduces the number of iterations required by 1.3 and 2.4 times compared to the Euclidean and Random initializations, respectively, across the entire test set. Even after including the diffusion generation time, our proposed hybrid model remains much faster than the commonly used initialization methods. These results offer compelling evidence of the efficacy of the approach. We hypothesize that the computational cost improvements could be even more pronounced in more complex scenarios, such as in 3D porous media.

4.3. Testing on Real Fracture Data Set

We tested our trained model on a publicly available real fracture in Berea sandstone (Karpyn et al., 2007) with a voxel size of 27.344, 27.344, 32.548 μm. We employed a segmented image as input to our model, which separates the solid matrix and the void space within the fracture. It is worth noting that the inherent pore space within the sandstone matrix is not captured by this binary segmentation. We employed 2D fractures which are slices of the 3D fracture in the direction perpendicular to the longest coordinate of the image. Despite training our model using simple synthetic fractures, we were able to accelerate the LBM simulations on the Berea fracture of 2.7 and 4.4 times compared to the Euclidean and Random initializations, respectively. Figure 6 shows the procedure to extract the fracture cross-sections from the 3D CT-scan (Karpyn et al., 2016). We then generated fluid realization on these cross-sections, some of these are shown in Figure 7. These configurations are then used as the starting points of the LBM simulations. In Figure 8 we show the number of iterations required for convergence and the simulation time for each initialization method, assessed on a data set of 32 fractures, the average values for each initialization is reported in Table 2. Consistent with the synthetic case, the computational costs in terms of iterations to convergence and simulation times still exhibit a clear advantage of our diffusion powered model over commonly used initializations for multiphase flow.

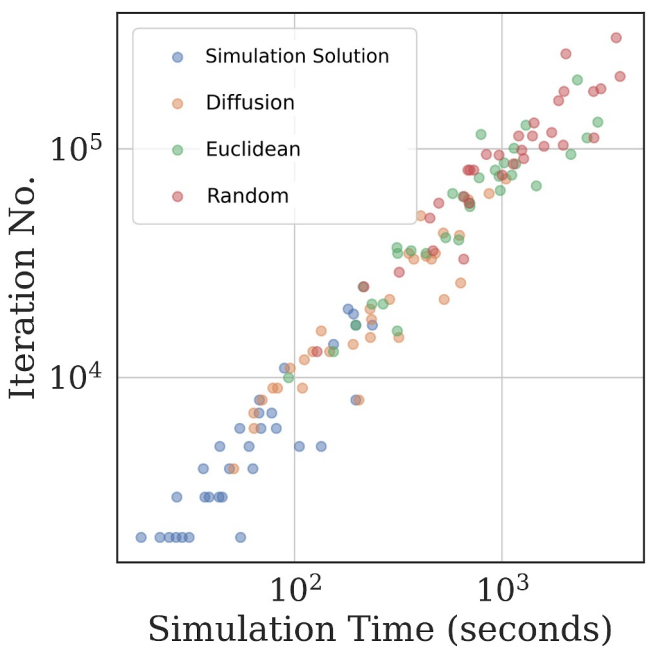

Figure 8. Scatter plot comparing the number of iterations needed for convergence with the corresponding simulation times across different initialization methods for the real fracture data set. Each point represents an individual LBM simulation, the average statistics for each initialization method is listed in Table 2.

5. Conclusion

In this study, we developed a hybrid approach that combines generative diffusion models with multiphase flow simulations. This method efficiently creates fluid configurations specific to fracture geometries, effectively managing varying saturation levels and significantly reducing the computational cost of pore-scale simulations. Our main contributions are the following. Fracture geometry conditioned training: Our model can generate fluid configurations within specific fracture geometries by incorporating geometric

Table 2
Berea Fracture Test Results

Initialization method	Number of iterations	Simulation time (min)
Simulation solution	6,500	1.3
Generative diffusion	24,400	5.3
Euclidean initialization	65,200	14.7
Random initialization	107,000	22.7

Note. Average number of iterations and average simulation time for 32 fracture geometries using different initialization methods.

conditioning. This allows the model to accurately represent complex fracture surfaces and consider phenomena that occur at or near these interfaces, as well as in more distant regions. Simulation feedback during training: We introduced a new metric to evaluate our model's performance—the number of iterations needed for simulation convergence. This metric improved as the model trained, demonstrating its ability to understand both the visual and physical aspects of fluid configurations. It also provided a more meaningful stopping criterion for training compared to just using the loss function.

Thanks to these advancements and the ability of diffusion models to parameterize complex distributions, our model, although trained with relatively simple synthetic data, demonstrated its effectiveness on a real Berea sandstone fracture. The results showed that our model could generalize well to complex fractures, capturing the main factors affecting multiphase flow. Our hybrid method significantly reduced the iterations needed for simulation convergence, resulting in substantial computational savings. In the case of the Berea sandstone fracture, our model accelerated simulations by up a factor of 4.4 compared to commonly used initialization methods.

Overall, our study provides evidence that a diffusion model-based approach is effective for pore-scale simulations. Looking ahead, exploring three-dimensional (3D) simulations could be a valuable avenue for future research. However, creating a data set for 3D simulations would be very computationally expensive, and adopting different neural networks would be necessary to manage the increased computational demands. Despite these challenges, the promising results from our study motivate further exploration in this direction. Models like this could provide researchers with a powerful tool to explore the sources of complex behaviors observed in nature. Furthermore, advancements in these algorithms could significantly transform reservoir modeling in the earth sciences. The future lies in hybrid approaches that combine machine learning with physics-based models, leveraging the strengths of both to achieve the best results. By integrating these methods, we can harness the computational efficiency of machine learning and the robustness of physics, demonstrating that both can coexist and complement each other effectively.

Data Availability Statement

The code for generating synthetic fractures, performing multiphase flow simulations (Chung & Santos, 2024), and implementing denoising diffusion models and data is available (Chung, 2024).

Acknowledgments
J.C. thanks the Applied Machine Learning (AML) summer program and Center for Nonlinear Studies (CNLS) at Los Alamos National Laboratory (LANL) for the mentoring and the fellowship received. AM gratefully acknowledge the support of the Center for Non-Linear Studies (CNLS) for this work. J.E.S and Y.T. acknowledge the support by the Laboratory Directed Research & Development project "Diffusion Modeling with Physical Constraints for Scientific Data (20240074ER)."

References

Bhatnagar, P. L., Gross, E. P., & Krook, M. (1954). A model for collision processes in gases. I. Small amplitude processes in charged and neutral one-component systems. *Physical Review*, *94*(3), 511–525. https://doi.org/10.1103/physrev.94.511

Blunt, M. J. (2001). Flow in porous media—Pore-network models and multiphase flow. *Current Opinion in Colloid & Interface Science*, *6*(3), 197–207. https://doi.org/10.1016/s1359-0294(01)00084-x

Blunt, M. J. (2017). *Multiphase flow in permeable media: A pore-scale perspective*. Cambridge University Press.

Blunt, M. J., Bijeljic, B., Dong, H., Gharbi, O., Iglauer, S., Mostaghimi, P., et al. (2013). Pore-scale imaging and modelling. *Advances in Water Resources*, *51*, 197–216. https://doi.org/10.1016/j.advwatres.2012.03.003

Bouras, H., Haroun, Y., Bodziony, F. F., Philippe, R., Fongarland, P., & Augier, F. (2022). Use of CFD for pressure drop, liquid saturation and wetting predictions in trickle bed reactors for different catalyst particle shapes. *Chemical Engineering Science*, *249*, 117315. https://doi.org/10.1016/j.ces.2021.117315

Brown, S. R. (1995). Simple mathematical model of a rough fracture. *Journal of Geophysical Research*, *100*(B4), 5941–5952. https://doi.org/10.1029/94jb03262

Chang, B., Santos, J., Victor, R., Viswanathan, H., & Prodanovic, M. (2022). Improving Machine Learning predictions of rock electric properties using 3d geometric features. In *SPE annual technical conference and exhibition*.

Chen, S., Wang, Z., Shan, X., & Doolen, G. D. (1992). Lattice Boltzmann computational fluid dynamics in three dimensions. *Journal of Statistical Physics*, *68*(3–4), 379–400. https://doi.org/10.1007/bf01341754

Chung, J. (2024). "Accelerating multiphase flow simulations with Denoising diffusion model driven initializations [Dataset]. *Zenodo*. Retrieved from https://zenodo.org/records/11452902

Chung, J., & Santos, J. E. (2024). Code for fracture generation and multiphase flow simulation used for training Denoising diffusion model-driven initializations. *Zenodo*. https://doi.org/10.5281/zenodo.14047107

Dai, Z., Viswanathan, H., Middleton, R., Pan, F., Ampomah, W., Yang, C., et al. (2016). CO_2 accounting and risk analysis for CO_2 sequestration at enhanced oil recovery sites. *Environmental science & technology*, *50*(14), 7546–7554. https://doi.org/10.1021/acs.est.6b01744

Fatt, I., & Klikoff, W. A., Jr. (1959). Effect of fractional wettability on multiphase flow through porous media. *Journal of Petroleum Technology*, *11*(10), 71–76. https://doi.org/10.2118/1275-g

Fitts, J. P., & Peters, C. A. (2013). Caprock fracture dissolution and CO_2 leakage. *Reviews in Mineralogy and Geochemistry*, *77*(1), 459–479. https://doi.org/10.2138/rmg.2013.77.13

Glover, P., Matsuki, K., Hikima, R., & Hayashi, K. (1998). Synthetic rough fractures in rocks. *Journal of Geophysical Research*, *103*(B5), 9609–9620. https://doi.org/10.1029/97jb02836

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT Press.

Guiltinan, E., Santos, J. E., & Kang, Q. (2020). Residual saturation during multiphase displacement in heterogeneous fractures with novel deep learning prediction. In *Unconventional resources technology conference, 20–22 July 2020* (pp. 3147–3152).

Guiltinan, E., Santos, J. E., Purswani, P., & Hyman, J. D. (2024). Pysimfrac: A python library for synthetic fracture generation and analysis. *Computers & Geosciences*, *191*, 105665. https://doi.org/10.1016/j.cageo.2024.105665

Guiltinan, E. J., Santos, J. E., Cardenas, M. B., Espinoza, D. N., & Kang, Q. (2021). Two-phase fluid flow properties of rough fractures with heterogeneous wettability: Analysis with lattice Boltzmann simulations. *Water Resources Research*, *57*(1), e2020WR027943. https://doi.org/10.1029/2020wr027943

Ho, J., Jain, A., & Abbeel, P. (2020). Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, *33*, 6840–6851.

Huang, H., Thorne, D. T. Jr, Schaap, M. G., & Sukop, M. C. (2007). Proposed approximation for contact angles in Shan-and-Chen-type multicomponent multiphase lattice Boltzmann models. *Physical Review E*, *76*(6), 066701. https://doi.org/10.1103/physreve.76.066701

Jettestuen, E., Helland, J. O., & Prodanović, M. (2013). A level set method for simulating capillary-controlled displacements at the pore scale with nonzero contact angles. *Water Resources Research*, *49*(8), 4645–4661. https://doi.org/10.1002/wrcr.20334

Karpyn, Z., Grader, A., & Halleck, P. (2007). Visualization of fluid occupancy in a rough fracture using micro-tomography. *Journal of Colloid and Interface Science*, *307*(1), 181–187. https://doi.org/10.1016/j.jcis.2006.10.082

Karpyn, Z., Landry, C., & Prodanovic, M. (2016). Induced rough fracture in Berea sandstone core. *Digital Rocks Portal*. https://doi.org/10.17612/P7J012

Krevor, S., De Coninck, H., Gasda, S. E., Ghaleigh, N. S., de Gooyert, V., Hajibeygi, H., et al. (2023). Subsurface carbon dioxide and hydrogen storage for a sustainable energy future. *Nature Reviews Earth & Environment*, *4*(2), 102–118. https://doi.org/10.1038/s43017-022-00376-8

Latt, J., Malaspinas, O., Kontaxakis, D., Parmigiani, A., Lagrava, D., Brogi, F., et al. (2021). Palabos: Parallel Lattice Boltzmann Solver. *Computers & Mathematics with Applications*, *81*, 334–350. https://doi.org/10.1016/j.camwa.2020.03.022

Middleton, R. S., Keating, G. N., Stauffer, P. H., Jordan, A. B., Viswanathan, H. S., Kang, Q. J., et al. (2012). The cross-scale science of CO_2 capture and storage: From pore scale to regional scale. *Energy & Environmental Science*, *5*(6), 7328–7345. https://doi.org/10.1039/c2ee03227a

Nichol, A. Q., & Dhariwal, P. (2021). Improved denoising diffusion probabilistic models. In *International conference on machine learning* (pp. 8162–8171).

Ogilvie, S. R., Isakov, E., & Glover, P. W. (2006). Fluid flow through rough fractures in rocks. II: A new matching model for rough rock fractures. *Earth and Planetary Science Letters*, *241*(3–4), 454–465. https://doi.org/10.1016/j.epsl.2005.11.041

Parker, J. (1989). Multiphase flow and transport in porous media. *Reviews of Geophysics*, *27*(3), 311–328. https://doi.org/10.1029/rg027i003p00311

Ronneberger, O., Fischer, P., & Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–Miccai 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, Part III* (Vol. 18, pp. 234–241). https://doi.org/10.1007/978-3-319-24574-4_28

Saharia, C., Ho, J., Chan, W., Salimans, T., Fleet, D. J., & Norouzi, M. (2022). Image super-resolution via iterative refinement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *45*(4), 4713–4726. https://doi.org/10.1109/tpami.2022.3204461

Santos, J., & Lin, Y. (2023). Understanding Denoising diffusion probabilistic models and their noise schedules via the Ornstein–Uhlenbeck process. In *Neurips 2023 workshop on diffusion Models*.

Santos, J. E., Gigliotti, A., Bihani, A., Landry, C., Hesse, M. A., Pyrcz, M. J., & Prodanović, M. (2022). MPLBM-UT: Multiphase LBM library for permeable media analysis. *SoftwareX*, *18*, 101097. https://doi.org/10.1016/j.softx.2022.101097

Santos, J. E., Marcato, A., Kang, Q., Mehana, M., O'Malley, D., Viswanathan, H., & Lubbers, N. (2024). Learning a general model of Single phase flow in complex 3d porous media. *Machine Learning: Science and Technology*, *5*(2), 025039. https://doi.org/10.1088/2632-2153/ad45af

Santos, J. E., Mehana, M., Wu, H., Prodanovic, M., Kang, Q., Lubbers, N., et al. (2020). Modeling nanoconfinement effects using active learning. *Journal of Physical Chemistry C*, *124*(40), 22200–22211. https://doi.org/10.1021/acs.jpcc.0c07427

Shan, X., & Chen, H. (1993). Lattice Boltzmann model for simulating flows with multiple phases and components. *Physical Review*, *47*(3), 1815–1819. https://doi.org/10.1103/physreve.47.1815

Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., & Ganguli, S. (2015). Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning* (pp. 2256–2265).

Sweeney, M. R., Hyman, J. D., O'Malley, D., Santos, J. E., Carey, J. W., Stauffer, P. H., & Viswanathan, H. S. (2023). Characterizing the impacts of multi-scale heterogeneity on solute transport in fracture networks. *Geophysical Research Letters*, *50*(21), e2023GL104958. https://doi.org/10.1029/2023gl104958

Ting, A. K., Santos, J. E., & Guiltinan, E. (2022). Using machine learning to predict multiphase flow through complex fractures. *Energies*, *15*(23), 8871. https://doi.org/10.3390/en15238871

Tongwa, P., Nygaard, R., Blue, A., & Bai, B. (2013). Evaluation of potential fracture-sealing materials for remediating CO_2 leakage pathways during CO_2 sequestration. *International Journal of Greenhouse Gas Control*, *18*, 128–138. https://doi.org/10.1016/j.ijggc.2013.06.017

Van der Maaten, L., & Hinton, G. (2008). Visualizing data using t-SNE. *Journal of Machine Learning Research*, *9*(11).

Viswanathan, H. S., Ajo-Franklin, J., Birkholzer, J. T., Carey, J. W., Guglielmi, Y., Hyman, J. D., et al. (2022). From fluid flow to coupled processes in fractured rock: Recent advances and new frontiers. *Reviews of Geophysics*, *60*(1), e2021RG000744. https://doi.org/10.1029/2021rg000744

Wang, Y. D., Chung, T., Armstrong, R. T., & Mostaghimi, P. (2021). Ml-LBM: Predicting and accelerating steady state flow simulation in porous media with convolutional neural networks. *Transport in Porous Media*, *138*(1), 49–75. https://doi.org/10.1007/s11242-021-01590-6

Wang, Z., Jeong, H., Gan, Y., Pereira, J.-M., Gu, Y., & Sauret, E. (2022). Pore-scale Modeling of multiphase flow in porous media using a Conditional Generative Adversarial Network (CGAN). *Physics of Fluids*, *34*(12). https://doi.org/10.1063/5.0133054

Yan, B., Harp, D. R., Chen, B., Hoteit, H., & Pawar, R. J. (2022a). A gradient-based deep neural network model for simulating multiphase flow in porous media. *Journal of Computational Physics*, *463*, 111277. https://doi.org/10.1016/j.jcp.2022.111277

Yan, B., Harp, D. R., Chen, B., & Pawar, R. (2022b). A physics-constrained deep learning model for simulating multiphase flow in 3d heterogeneous porous media. *Fuel*, *313*, 122693. https://doi.org/10.1016/j.fuel.2021.122693

Yeh, R. A., Chen, C., Yian Lim, T., Schwing, A. G., Hasegawa-Johnson, M., & Do, M. N. (2017). Semantic image Inpainting with deep generative models. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 5485–5493).

Yu, J., Lin, Z., Yang, J., Shen, X., Lu, X., & Huang, T. S. (2018). Generative image Inpainting with contextual attention. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 5505–5514).

Zhao, D., Hou, J., Wei, B., Liu, H., Du, Q., Zhang, Y., & Sun, Z. (2023). Fast prediction method of displacement front in heterogeneous porous media using deep learning and orthogonal design. *Physics of Fluids*, *35*(8). https://doi.org/10.1063/5.0160984