



**Forthcoming Articles** have been peer-reviewed and accepted for publication in *InterPore Journal*. To make research available as early as possible, accepted manuscripts may be published online prior to completion of the final production process. These versions have not yet undergone final formatting, proofreading, or author corrections and therefore do not represent the **Version of Record**. Minor changes may occur during the final production process. The article will be updated with the final Version of Record when it is published in its assigned journal issue. **Forthcoming Articles should be cited using their DOI, which will remain the same after publication of the final Version of Record.**

**Please cite the following article as:**

Rustamov, N., & Aryana, S. (2026). Fourier-Hermite Kinetic Network for Gas Transport Under High Knudsen Numbers. *InterPore Journal*. Forthcoming Article.  
<https://doi.org/10.69631/s2cmh450>

**Received on:** 26 May. 2026

**Accepted on:** 29 Aug. 2026

**Published online (Accepted version):** 16 Sep. 2026

# FOURIER-HERMITE KINETIC NETWORK FOR GAS TRANSPORT UNDER HIGH KNUDSEN NUMBERS

<sup>1</sup> Nijat Rustamov<sup>1</sup>  and Saman A. Aryana<sup>1</sup> 

<sup>2</sup> <sup>1</sup>*Department of Chemical and Biomedical Engineering, University of Wyoming*

## Correspondence to:

Saman A. Aryana,  
[saryana@uwyo.edu](mailto:saryana@uwyo.edu)

## How to Cite:

This information will  
 be updated by the  
 Editorial Office once  
 the article is ready to  
 be published.  
<https://doi.org/10.69631/xxxxxxx>

**RECEIVED:** 00

Month 0000

**ACCEPTED:** 01

Month 0000

**AVAILABLE**

**ONLINE:** 02 Month  
 0000

**VERSION OF**

**RECORD:** j<sub>final</sub>  
 publication date<sub>i</sub>

## ABSTRACT

This work presents the Fourier-Hermite Kinetic Network (FHKN), a novel sequence-to-state temporal-spectral architecture designed to accelerate the prediction of rarefied gas transport in porous media. Traditional lattice Boltzmann method (LBM) simulations in complex geometries require massive computational overhead due to slow transient relaxation. To bypass auto-regressive time-stepping, the FHKN extracts dominant relaxation frequencies from the initial transient iterations of discrete velocity distribution functions ( $f_i$ ) via a 1D temporal Fourier transform. The architecture employs a truncated Hermite expansion block that isolates the 0th, 1st, and 2nd order moments. This mathematically aligns with the LBM Gauss-Hermite quadrature, acting as a low-pass kinetic filter that discards high-frequency non-equilibrium noise and maps directly to the macroscopic Navier-Stokes limit. Trained on optimized C++ LBM data from simple random granular domains, FHKN significantly outperforms isolated Fourier and Hermite networks, achieving a minimum point-wise  $R^2$  of 0.90 for velocity fields while demonstrating zero-shot scalability to grids with four times the training resolution. To validate engineering utility, the model was subjected to out-of-distribution blind tests on highly tortuous geometries generated via random Gaussian noise. FHKN successfully mapped early transient data to final steady-state fields, predicting macroscopic permeability within a 9% error margin and effectively bypassing the up to 300,000 iterations required for LBM convergence. This framework provides a physically-aligned, highly efficient feature-extraction method for modeling rarefied transport without relying on iterative spatial solvers.

## KEYWORDS

Porous media, Lattice Boltzmann method, Rarefied gas transport, Kinetic networks, Fourier-Hermite spectral decomposition, Shale gas



© 2026 The Authors

This is an open access article published by InterPore under the terms of the Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License (CC BY-NC-ND 4.0)

(<https://creativecommons.org/licenses/by-nc-nd/4.0/>).

## 1. INTRODUCTION

Shales are a major source of natural gas, a comparatively low-carbon fossil fuel, and are increasingly recognized as effective sealing formations for the storage of carbon dioxide and hydrogen, underscor-

ing the importance of improving our understanding of gas transport within these confined and complex media [4, 2, 1]. Nevertheless, several inherent characteristics of shales including strong heterogeneity, non-classical transport phenomena, and complex phase behavior pose substantial challenges to accurately simulating flow and transport in such formations. Shales are predominantly composed of ultra-fine-grained rocks exhibiting a wide range of chemical compositions and structural attributes, with characteristic length scales spanning from nanometers to millimeters [21]. In extreme cases, pore sizes may approach molecular dimensions, giving rise to pronounced confinement effects [36]. Under such conditions, fluid–solid interactions are significantly enhanced, leading to marked departures from bulk fluid behavior. In addition, prior studies have demonstrated that pore surface roughness plays a critical role in governing methane transport within the highly irregular pore networks of shales [12, 20].

A central difficulty in characterizing fluid flow in shale formations arises from the broad distribution of pore sizes, which ranges from nanoscale pores to fractures on the order of millimeters [14, 5]. Following the classification of the International Union of Pure and Applied Chemistry (IUPAC) into micropores ( $< 2$  nm), mesopores (2 – 50 nm), and macropores ( $> 50$  nm), Kuila and Prasad [15] reported that the shale matrix is predominantly composed of micropores and mesopores. As noted by Mehmani et al. [22], the lack of a clear separation between these length scales renders traditional scale-translation approaches, such as homogenization techniques, largely ineffective. Consequently, it becomes necessary to investigate individual scales in isolation and to examine their coupled behavior, which emerges from complex interactions across scales.

Gas transport regimes are commonly classified using the Knudsen number ( $Kn$ ), defined as the ratio of the molecular mean free path to a characteristic length scale of the medium [17]. For  $Kn < 0.001$ , gas flow can be accurately described by continuum mechanics formulations, such as the Navier–Stokes equations with no-slip boundary conditions. When  $0.001 < Kn < 0.1$ , the system enters the slip-flow regime, in which the Navier–Stokes equations augmented with slip boundary conditions remain applicable [7]. As  $Kn$  approaches 0.1, the flow transitions toward a regime intermediate between slip flow and free-molecular transport, where kinetic-theory-based descriptions become more appropriate [26]. For  $Kn > 10$ , transport is dominated by free-molecular behavior. Because the micro- and mesopores that dominate the shale matrix are comparable in size to the methane mean free path, the Knudsen number falls within the slip and transition regimes, particularly at the low pressures [6]. Given the extensive range of length scales inherent to shale systems, a multi-scale modeling framework is therefore required to capture the relevant physics at each scale and to elucidate their combined impact on system-level transport behavior [27].

Over the past several decades, the lattice Boltzmann (LB) method has gained attention as a viable alternative to traditional continuum-based approaches for modeling transport processes [29]. The LB formulation is obtained through a discrete representation of the Boltzmann transport equation, relying on a series of simplifying assumptions [30]. As a mesoscopic technique, the LB method tracks the evolution of particle probability distribution functions and is capable of recovering macroscopic quantities such as pressure and velocity fields [3]. Owing to its algorithmic simplicity, LB often offers computational advantages over conventional numerical methods, including finite volume schemes, and is well suited for efficient parallelization [16]. While the LB framework is particularly effective at bridging the continuum and free-molecular regimes [20], the development of high-fidelity, multi-scale simulations remains computationally demanding, necessitating complementary strategies to mitigate the associated computational costs. To address this, we draw on ideas from operator learning, a machine-learning strategy for constructing fast surrogates of expensive simulations.

Physics-aware machine learning (PaML) encompasses a range of approaches that embed physical knowledge into neural architectures, from soft constraint enforcement to hard architectural design. A widely studied class are physics-informed neural networks (PINNs), which represent PDE solutions by neural networks and enforce governing equations, boundary, and initial conditions through residual losses [25]. PINNs are attractive because they are mesh-free, integrate observational data with PDE constraints, and

can solve forward and inverse problems within a unified framework [34]. However, their optimization is notoriously challenging: training involves balancing PDE residuals with boundary/initial conditions, often leading to ill-conditioned loss landscapes, slow convergence, and sensitivity to hyperparameters. Moreover, standard PINNs often struggle with multi-scale or high-frequency solution features, particularly in still or strongly heterogeneous systems, unless augmented with problem-specific strategies such as adaptive loss reweighting, domain decomposition, or multi-resolution encoding [34]. A different paradigm is offered by neural operators, which learn mappings between function spaces rather than single PDE instances. The Fourier Neural Operator (FNO) [18] parameterizes the integral kernel of the operator in Fourier space, combining global spectral convolution layers with nonlinear activations. This enables discretization-invariance, allowing models trained on coarse grids to generalize to finer ones ("zero-shot" super-resolution), and offers quasi-linear scaling via Fast Fourier Transform (FFT). Once trained, FNOs act as fast surrogates for entire PDE families, achieving orders-of-magnitude speedups compared to traditional CFD solvers while retaining strong accuracy on benchmarks such as Burgers' equation, Darcy flow, and incompressible Navier–Stokes. Importantly, nonlinear activation restores high-frequency accuracy lost by Fourier mode truncation, giving FNOs robustness to multiscale structure. In contrast, traditional CFD methods can provide well-established mechanisms for enforcing numerical stability, discrete conservation, and complex-geometry handling, but often at high computational cost. PINNs provide flexibility in data-scarce or inverse settings where enforcing physics is more important than efficiency, while neural operators like FNO are best suited as surrogates when many forward solves are required across parameter spaces. Hybrid approaches—embedding learned operators within solvers or coupling PINNs with numerical methods—are emerging as a pathway to combine efficiency with physical reliability.

Related to the present work is a body of learning-based methods that shorten the number of iterations to a converged solution. A common strategy is to run the numerical solver for a limited number of iterations and use the resulting data either to predict the converged flow field or to warm-start the solver. Ning et al. trained a diffusion probabilistic model that maps an unconverged airfoil solution toward its steady state, achieving a 2.4-fold speed-up [24]. Hu et al. proposed a data-driven flow initialization to accelerate simulations of flow around underwater vehicles [13]. While their method does not directly predict the steady-state solution, it yields up to roughly 2 to 3.5 times faster wall-clock times depending on the convergence threshold. Similarly, Lim et al. employed a graph convolutional network for flow-field initialization to accelerate numerical convergence by 30%–50% [19]. A common theme in these methods is that they operate on macroscopic flow fields produced by continuum solvers. To our knowledge, none of these strategies has been formulated at the kinetic level, that is, directly on the distribution functions of a lattice Boltzmann simulation, which is the setting we address here.

Motivated by the efficiency of these spectral operator-learning approaches, in this work we combine Fourier and Hermite transforms to construct a Fourier–Hermite Kinetic Network (FHKN) that maps the temporal evolution of lattice Boltzmann (LB) distribution functions to their steady-state solution. While our method cannot be classified as a neural operator by definition, it adopts the core idea of the FNO, a global spectral parameterization of the mapping, realized here through the Fourier and Hermite transforms rather than through learned kernels acting between function spaces. Because reaching the steady state with the lattice Boltzmann method (LBM) otherwise requires many time-marching iterations, we run LBM for only a few hundred iterations and train FHKN to predict the steady-state solution directly, thereby avoiding the costly remainder of the simulation and acting as a fast surrogate once trained. The motivation for combining Fourier and Hermite transforms is that the Hermite expansion naturally discretizes the equilibrium and non-equilibrium Maxwell–Boltzmann distributions used in the LB equation, while the Fourier transform captures the temporal evolution of the distribution functions. Specifically, the Fourier transform models time dynamics, and the Hermite transform maps the moment space of intermediate distributions to the steady state.

The remainder of this paper is organized as follows. Section 2 describes the LBM simulation framework, Section 3 presents the mathematical formulation of the FHKM. Section 4 reports the results, and Section 5 summarizes the findings.

## 2. SIMULATION FRAMEWORK

The framework for simulating gas transport in rarefied environments is based on the lattice Boltzmann equation, which can be expressed as [11]:

$$f_\alpha(\mathbf{x} + \mathbf{e}_\alpha \delta t, t + \delta t) - f_\alpha(\mathbf{x}, t) = \Omega_\alpha(f(\mathbf{x}, t)). \quad (1)$$

Here,  $f_\alpha(\mathbf{x}, t)$  denotes the particle distribution in direction  $\alpha$  at position  $\mathbf{x}$  and time  $t$ ,  $\Omega_\alpha$  is the corresponding collision operator, which depends on the set of populations  $\mathbf{f}$  at the node. We use the  $D_2Q_9$  lattice model and a multiple-relaxation-time (MRT) collision operator for improved numerical stability.

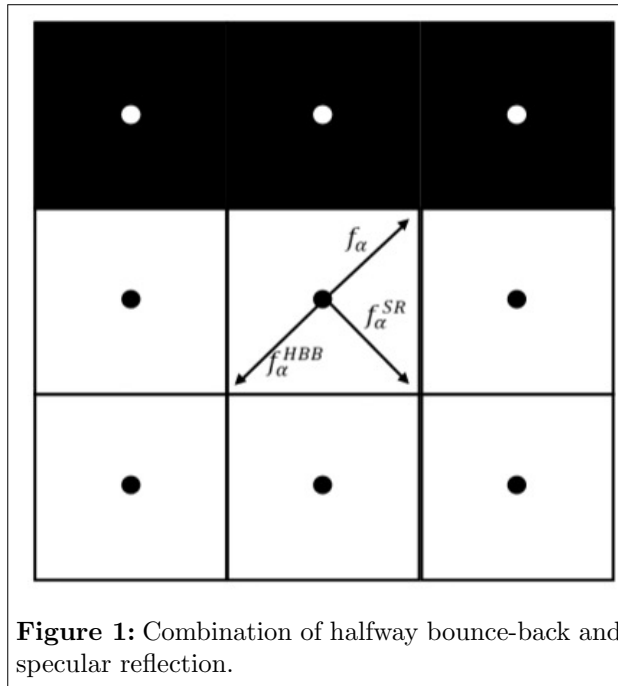
To capture the effects of rarefaction, the relaxation parameters  $\tau_s$  (related to viscosity) and  $\tau_q$  (related to the third-order non-hydrodynamic energy flux) are modified and calculated using the following formulas [23, 10]:

$$\tau_s = \frac{1}{2} + \sqrt{\frac{6}{\pi}} \frac{HKn}{1 + 2Kn}, \quad (2)$$

$$\tau_q = \frac{1}{2} + \frac{3 + A_2 \pi (2\tau_s - 1)^2}{8(2\tau_s - 1)}. \quad (3)$$

Here,  $H$  denotes the local characteristic length (i.e., pore size) and  $Kn$  is the local Knudsen number. In practice,  $H$  must be interpreted consistently with the spatial discretization, such as scaling to lattice units ( $H/\delta x$ ). The parameter  $A_2$  is a coefficient associated with the second-order velocity slip boundary condition [33]. Furthermore, Eq. (3) follows from Eq. (51b) of Guo et al., assuming that  $\tau_s$  remains locally constant near the wall [10].

To reduce spurious non-hydrodynamic contributions, particularly near boundaries under confinement, the non-equilibrium components of the distribution function are regularized by projecting them onto a second-order Hermite subspace, as proposed in [35]. In slip and transitional flow regimes, accurate



boundary conditions are critical. Due to increased gas slip along the wall, a second-order slip boundary

condition is necessary.

$$u_s = A_1 Kn \left. \frac{\partial u}{\partial n} \right|_w - A_2 Kn^2 \left. \frac{\partial^2 u}{\partial n^2} \right|_w \quad (4)$$

where  $A_1$  and  $A_2$  are the first and second order slip coefficients, respectively,  $\mathbf{n}$  is the wall normal vector, and  $w$  denotes the derivative of the quantity at the wall. The values of the slip coefficients  $A_1 = 0.6$  and  $A_2 = 0.9$  are adapted from Wang and Aryana [33]. To capture the slip velocity at the solid walls, we combine halfway bounce-back (HBB) and specular reflection (SR) conditions [31, 9]:

$$f_\alpha(\mathbf{x}_b, t + \delta t) = r f_\alpha^*(\mathbf{x}_b, t) + (1 - r) f_{\alpha^{SR}}^*(\mathbf{x}_b, t), \quad (5)$$

where  $(\cdot)^*$  denotes the post-collision population (evaluated consistently with the chosen halfway implementation),  $\bar{\alpha}$  is the lattice direction opposite to  $\alpha$  (HBB), and  $\alpha^{SR}$  is the discrete index corresponding to the specularly reflected direction (SR). In the limit  $r \rightarrow 1$ , Eq. (5) reduces to pure bounce-back, producing no-slip behavior, whereas  $r \rightarrow 0$  approaches pure specular reflection. The schematic of bounce-back and specular reflection is shown in Figure 1. The bounce-back component is simply the reversed direction, whereas the specular reflection is reflected in the manner of light reflection. The equation below determines the direction of this reflection.

$$\mathbf{e}_\alpha^{SR} = \mathbf{e}_\alpha - 2(\mathbf{e}_\alpha \mathbf{n}) \mathbf{n} \quad (6)$$

where  $\mathbf{e}_\alpha^{SR}$  is the reflected velocity vector,  $\mathbf{e}_\alpha$  is the incoming velocity toward the wall, and  $\mathbf{n}$  is the wall normal direction. The weighting factor  $r$  is determined from the second-order slip formulation derived in [10]:

$$r = \left( 1 + \sqrt{\frac{\pi}{6}} A_1 + \frac{\tau'_s(0)}{8(\tau_s(0) - \frac{1}{2})} \right)^{-1}, \quad (7)$$

where  $\tau_s(0)$  is the shear-related relaxation time evaluated at the wall and  $\tau'_s(0)$  denotes its derivative in the wall-normal direction. When  $\tau_s$  is locally constant near the wall,  $\tau'_s(0) = 0$ , and Eq. (7) reduces to the leading-order expression provided in [10]. To efficiently handle irregular geometries, we categorize distinct boundary types and implement a lookup table to optimize the calculation of specular reflection directions during simulation, as previously described in [28].

## 2.1. Local Knudsen number

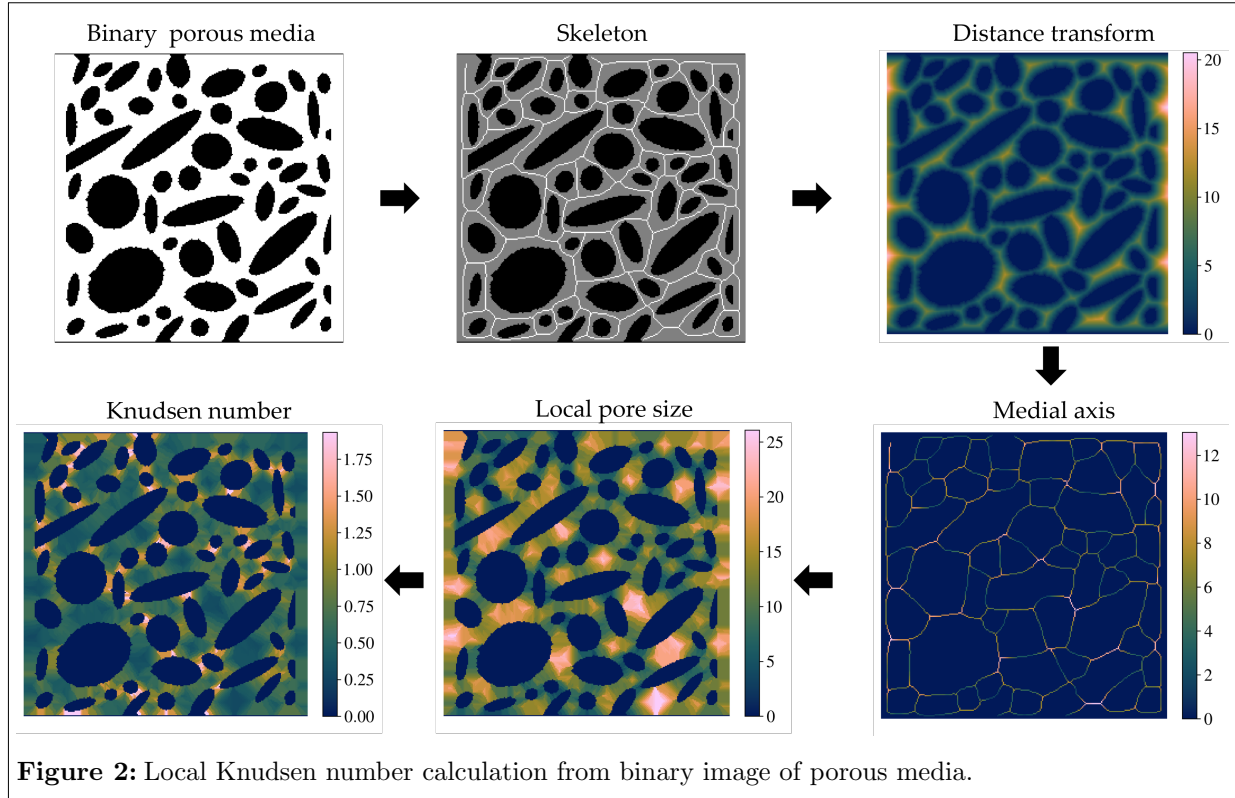
The Knudsen number is the ratio of the mean free path to a representative pore size. The mean free path in kinetic theory is calculated using:

$$\lambda = \frac{k_B T}{\sqrt{2} \pi d^2 P} \quad (8)$$

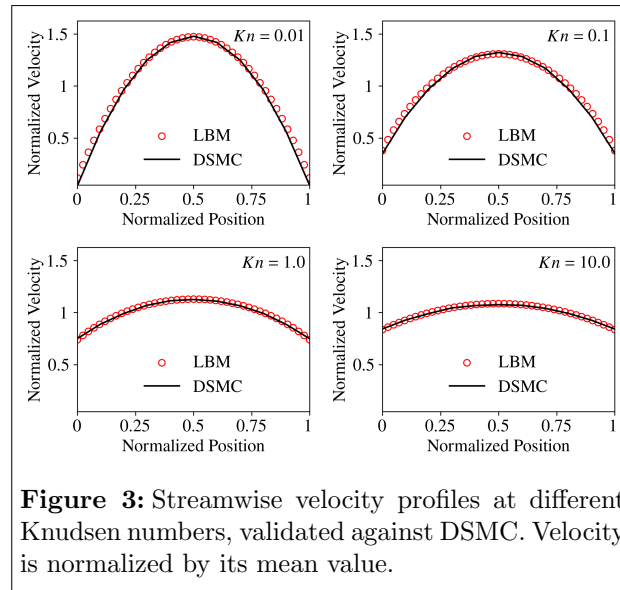
where  $k_B$  is the Boltzmann constant,  $T$  is temperature,  $d$  is the gas molecular diameter (methane in this work), and  $P$  is pressure. Consequently, the Knudsen number is calculated as:

$$Kn = \frac{\lambda}{H} \quad (9)$$

where  $H$  is the pore size. In simple channels the pore size is constant. However, in porous media this is not the case. To find the local pore-size distribution and Knudsen number, we employ image-processing techniques. As shown in Figure 2, the binary image, where pores are represented as white regions and grains as black, is first skeletonized using morphological thinning and then undergoes a Euclidean distance transform. Multiplying the distance transform by the skeleton yields the distance from the medial axis. Next, for each pore pixel, we identify the nearest point on the medial axis and define twice the distance from that point to the pore wall as the local pore size. Finally, the local Knudsen number is determined as the ratio of the molecule's mean free path to the local pore size.



## 2.2. Validation



To validate the accuracy of the described LBM model against Direct Simulation Monte Carlo (DSMC) data from Suga [32], simulations were performed at four Knudsen numbers ranging from 0.01 to 10. These cases span the slip, transitional, and free molecular flow regimes, with periodic boundary conditions applied at the inlet and outlet. The flow is driven by a constant body force.

$$\mathbf{u}(\mathbf{x}, t) = \frac{1}{\rho(\mathbf{x}, t)} \sum_{\alpha} f_{\alpha}(\mathbf{x}, t) \mathbf{e}_{\alpha} + \frac{\mathbf{F}}{2\rho(\mathbf{x}, t)} \quad (10)$$

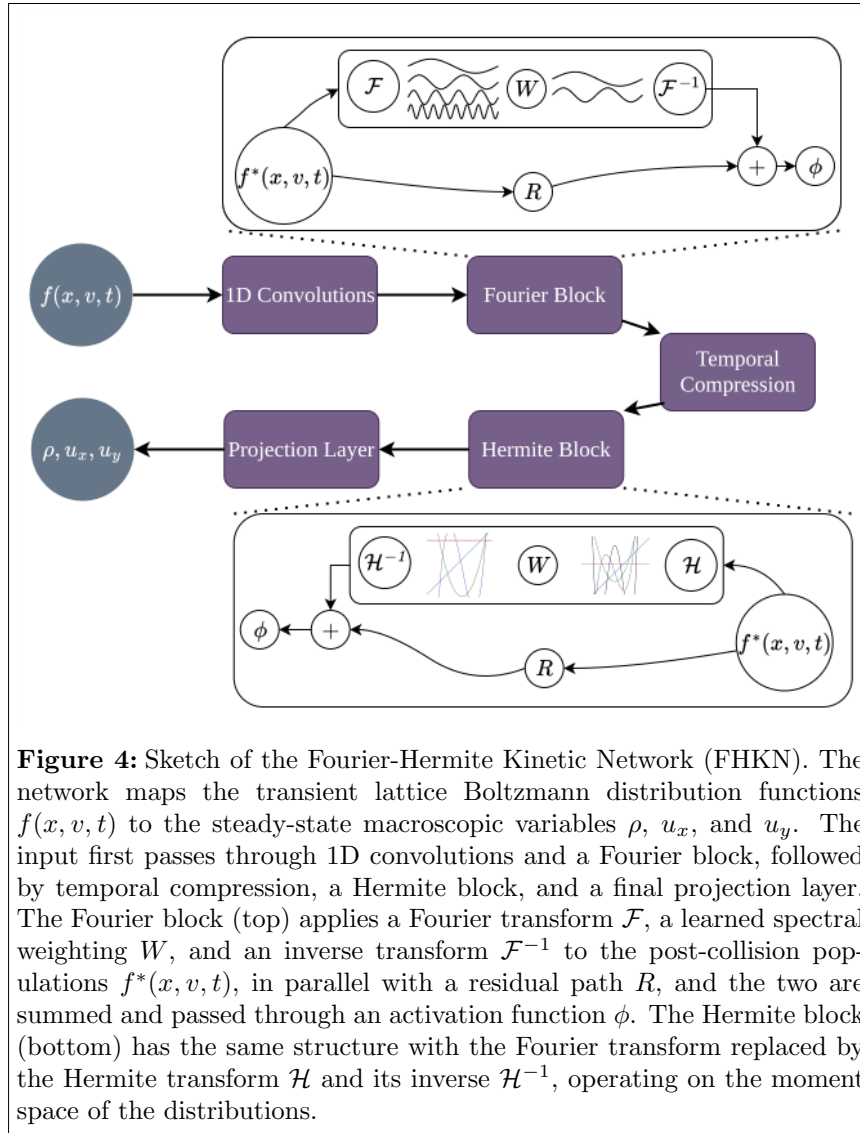
where  $\mathbf{u}(\mathbf{x}, t)$  is the lattice velocity,  $\rho(\mathbf{x}, t) = \sum_{\alpha} f_{\alpha}$  is the lattice density, and  $\mathbf{F}$  is the force density in lattice units, with magnitude  $10^{-8}$ . As shown in Figure 3, the velocity slip becomes increasingly significant as the Knudsen number grows, and the LBM profiles agree closely with the DSMC results across all four regimes. These simulations are carried out on a  $45 \times 45$  spatial domain at a pressure of 2.0 MPa and a temperature of 300 K. These conditions set the methane mean free path through Eq. (8), and fix the initial

**Table 1:** Combination coefficient at different Kn.

| $Kn$ | 0.01 | 0.1  | 1.0  | 10   |
|------|------|------|------|------|
| $r$  | 0.64 | 0.62 | 0.52 | 0.45 |

density, obtained from the Peng-Robinson equation of state, used to initialize the distribution functions at equilibrium. The corresponding values of the combination coefficient  $r$ , calculated using Eq. (7) for the different Knudsen numbers, are listed in Table 1.

### 3. FOURIER-HERMITE KINETIC NETWORK



This section presents the Fourier-Hermite Kinetic Network (FHKN), a surrogate model mapping a short temporal sequence of lattice Boltzmann distribution functions to macroscopic steady-state observables. The network input is a tensor  $\mathbf{f} \in \mathbb{R}^{9 \times T}$ , representing the  $D_2Q_9$  lattice populations over a temporal window of length  $T$ . The target output is  $\hat{\mathbf{y}} = [\hat{\rho}, \hat{u}_x, \hat{u}_y]^\top$ , containing the steady-state density and velocity components.

The architecture initiates with a temporal lifting stage using standard 1D temporal convolutions to extract high-dimensional features  $X \in \mathbb{R}^{C \times T_{\text{FFT}}}$ . To model long-range temporal correlations, the network applies a sequence of spectral blocks. Within each block, the temporal sequence is transformed into the frequency domain via a discrete Fourier transform ( $\mathcal{F}_t$ ). Only the lowest  $K$  frequency modes are retained. For each

**Table 2:** Methane properties and physical conditions used in the simulations.

| Property             | Symbol   | Value                    |
|----------------------|----------|--------------------------|
| Temperature          | $T$      | 300 K                    |
| Pressure             | $P$      | 2.0 MPa                  |
| Critical temperature | $T_c$    | 190.564 K                |
| Critical pressure    | $P_c$    | 4.5992 MPa               |
| Molecular diameter   | $d$      | $3.80 \times 10^{-10}$ m |
| Acentric factor      | $\omega$ | 0.011                    |
| Molecular weight     | $M_w$    | 16.043 g/mol             |

frequency  $k < K$ , a learned complex-valued channel-mixing matrix  $W(k) \in C^{C \times C}$  is applied to the Fourier coefficients  $X(k)$ :

$$Y(k) = X(k) W(k). \quad (11)$$

Frequencies  $k \geq K$  are zeroed out. This is because the relaxation toward steady state is dominated by low-frequency temporal modes, whereas high frequencies mainly carry early-time transients and noise. Retaining only the lowest  $K$  modes therefore acts as a low-pass filter and limits the parameter count. We select  $K = 48$  experimentally as the smallest value that preserves accuracy on the validation set. The signal is returned to the time domain using the inverse Fourier transform ( $\mathcal{F}_t^{-1}$ ) and combined with a pointwise linear residual pathway  $R(t)$  to produce the block's output:

$$X'(t) = \phi(\mathcal{F}_t^{-1}[Y](t) + R(t)), \quad (12)$$

where  $\phi$  is a smooth nonlinearity.

Following the spectral blocks, the temporal dimension is collapsed into a purely spatial feature vector  $\mathbf{s} \in R^C$  through a learned linear temporal aggregation. This vector serves as the input to the Hermite stage, which operates in the kinetic moment space associated with the  $D_2Q_9$  lattice. Using the standard discrete velocities and quadrature weights  $w_i$ , the features are projected onto Hermite polynomials up to second order  $(H_0, H_1^\alpha, H_2^{\alpha\beta})$ , generating the corresponding discrete moments  $a_0, a_1^\alpha$ , and  $a_2^{\alpha\beta}$ .

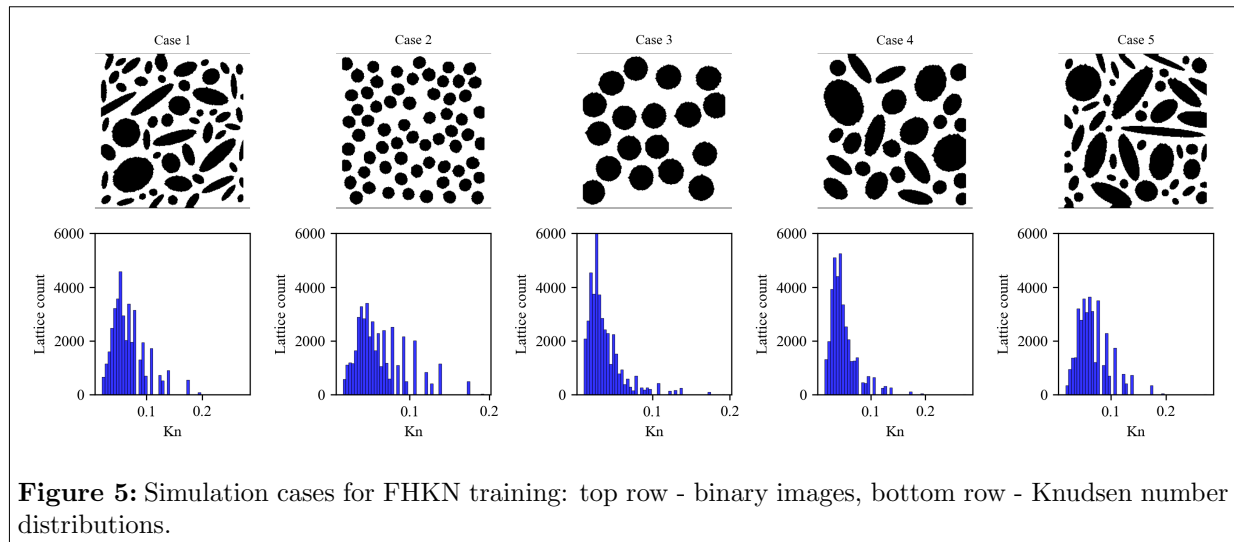
To automatically capture standard Hermite normalization factors during training, these moments are elementwise rescaled using learnable coefficients  $\omega_0, \omega_1$ , and  $\omega_2$ . The feature representation is then reconstructed by projecting back onto the  $D_2Q_9$  basis and adding a parameterized residual connection:

$$\psi_i = w_i \left( \omega_0 a_0 + \sum_{\alpha \in \{x, y\}} H_1^\alpha(i) \omega_1^\alpha a_1^\alpha + \frac{1}{2} \sum_{\alpha, \beta \in \{x, y\}} H_2^{\alpha\beta}(i) \omega_2^{\alpha\beta} a_2^{\alpha\beta} \right) + \gamma_i s_i, \quad (13)$$

where  $i \in \{1, \dots, 9\}$  and  $\gamma$  is a learnable diagonal weight vector. Finally, the Hermite-stage representation  $\psi$  is mapped to the predicted macroscopic observables  $\hat{\mathbf{y}}$  via a standard multilayer perceptron (MLP).

## 4. DATA DESCRIPTION

Five simulation scenarios were generated on  $256 \times 256$  grids with a spatial resolution of 1 nm as shown in Figure 5. The pore geometries were constructed by randomly placing circular and elliptical obstacles while enforcing a non-overlap constraint. The resulting Knudsen number ranges from 0.02 to 0.2 spanning slip and transitional flow regimes. For each case, methane transport was simulated at a pressure of 2 MPa and a temperature of 300 K under a constant body force. Methane properties are given in Table 2. The lattice Boltzmann method employed here is effectively incompressible with  $Ma < 1$ . From each simulation, the lattice Boltzmann distribution functions in all nine discrete velocity directions were recorded for the first 1000 iterations, together with the steady-state macroscopic fields, namely density and the velocity components in the  $x$  and  $y$  directions. A summary of the number of active lattice cells and the number of iterations required to reach steady state is reported in Table 3. Among cases 1–5, case 2 contains the largest number of active cells (43,359) and exhibits the highest per-iteration cost (approximately



**Figure 5:** Simulation cases for FHKN training: top row - binary images, bottom row - Knudsen number distributions.

**Table 3:** Simulation details

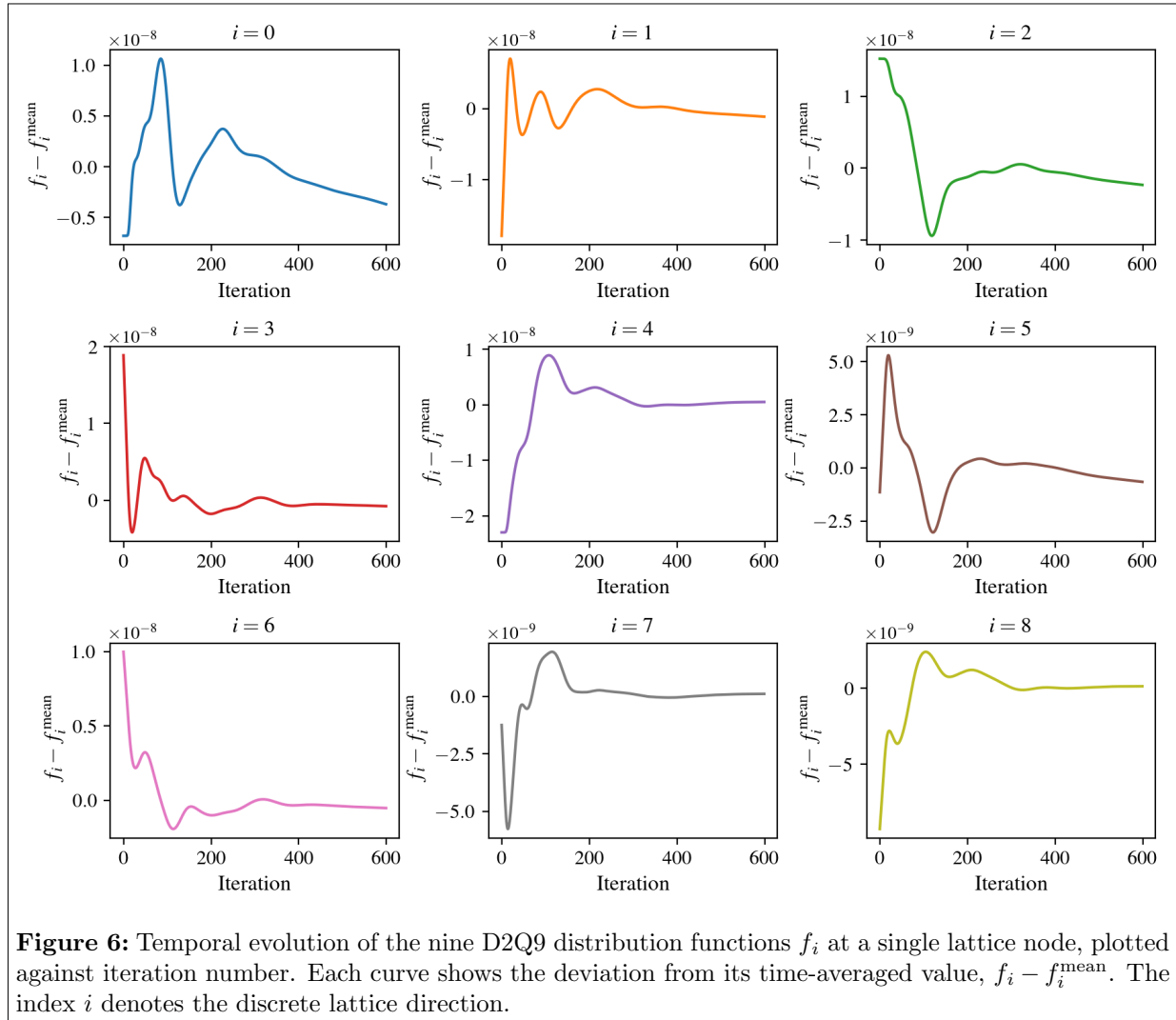
| Case | Total wall-time (ms) | Total timesteps | Active cells | Wall-time             |  |
|------|----------------------|-----------------|--------------|-----------------------|--|
|      |                      |                 |              | per LB iteration (ms) |  |
| 1    | 14324                | 7811            | 39152        | 1.833                 |  |
| 2    | 14406                | 7558            | 43359        | 1.906                 |  |
| 3    | 15402                | 8418            | 41286        | 1.830                 |  |
| 4    | 9788                 | 5891            | 37059        | 1.661                 |  |
| 5    | 17178                | 10645           | 36258        | 1.614                 |  |

1.91 ms), whereas case 5 has the lowest cost per iteration (about 1.61 ms) but requires substantially more iterations (10,645), leading to the longest overall runtime (around 17 s). In contrast, case 4 reaches steady state in the fewest iterations (5,891) and therefore completes fastest (roughly 9.8 s), despite only a moderate reduction in per-step cost ( $\sim 1.66$  ms). Overall, the computational cost per iteration scales with the number of active cells, as expected for a linear LB solver, while the total wall-time is governed primarily by the number of iterations required for convergence rather than by the per-step expense. Case 3 combines a relatively large grid with a high iteration count, making it slower than case 1, which occupies an intermediate position across most metrics.

Figure 6 illustrates the temporal evolution of the distribution functions at a representative lattice site from case 1. As expected, the stationary population exhibits the largest magnitude, followed by the primary directions  $i = 1, 2, 3, 4$ , while the diagonal directions  $i = 5, 6, 7, 8$  remain smallest. The sampled  $D_2Q_9$  populations display short transient behavior at early times: during approximately the first 10–100 iterations, several components overshoot or undershoot and undergo damped oscillations before stabilizing. This transient response arises from the abrupt application of forcing in a confined pore and the temporary formation of local recirculation regions, causing populations aligned or misaligned with the eventual mean flow to momentarily increase or decrease. As collision and streaming redistribute momentum and the bounce-back mechanism damps fast kinetic modes, these oscillations decay over the relaxation timescale, after which the slower hydrodynamic mode dominates and the distributions settle into near-steady plateaus.

Figure 6 also provides an example of the input supplied to the Fourier-Hermite Kinetic Network (FHKN). The underlying assumption is that the temporal signatures of the directional populations encode the pathway by which a lattice site approaches steady state. In the initial stage, one-dimensional convolutions along the iteration axis are used to modulate short-lived transients and to mix information across the  $D_2Q_9$  channels. The subsequent Fourier block isolates the dominant low-frequency components associated with steady behavior, while the Hermite block projects the features onto low-order kinetic moments, namely

density and momentum, thereby filtering out high-frequency kinetic fluctuations and maintaining a direct connection to  $\rho$  and  $\mathbf{u}$ . In the present study, adsorption effects are neglected, and the prediction head is therefore trained to output only the velocity components  $u_x$  and  $u_y$ .



## 5. MODEL TRAINING

For training, simulation cases 1 and 2 provide 82,511 samples, where each sample represents an individual lattice cell. Each sample includes 1,000 lattice Boltzmann (LB) time steps. The dataset is split into 80% training (66,008 cells) and 20% validation (16,503 cells). Data is normalized to the interval  $[0, 1]$  fitted strictly on the training set to prevent leakage, and applied independently to each discrete velocity direction across all time steps.

The network is optimized using PyTorch's SmoothL1 (Huber) loss, chosen for its robustness against outliers and stable gradients. Optimization is performed using Adam with standard hyperparameters ( $\beta_1 = 0.9$ ,  $\beta_2 = 0.999$ ,  $\varepsilon = 10^{-8}$ ) and a base learning rate of  $\alpha = 0.001$ . A ReduceLROnPlateau scheduler is employed to reduce the learning rate by a factor of 0.5 if the validation loss fails to improve for 5 consecutive epochs.

To evaluate the trade-off between predictive accuracy and the computational cost of generating input data, ten models were trained using varying temporal input lengths  $T \in \{100, 200, 300, \dots, 1000\}$ . Training was executed on two NVIDIA RTX A6000 GPUs (48 GB each) utilizing PyTorch's Distributed Data Parallel (DDP) framework.

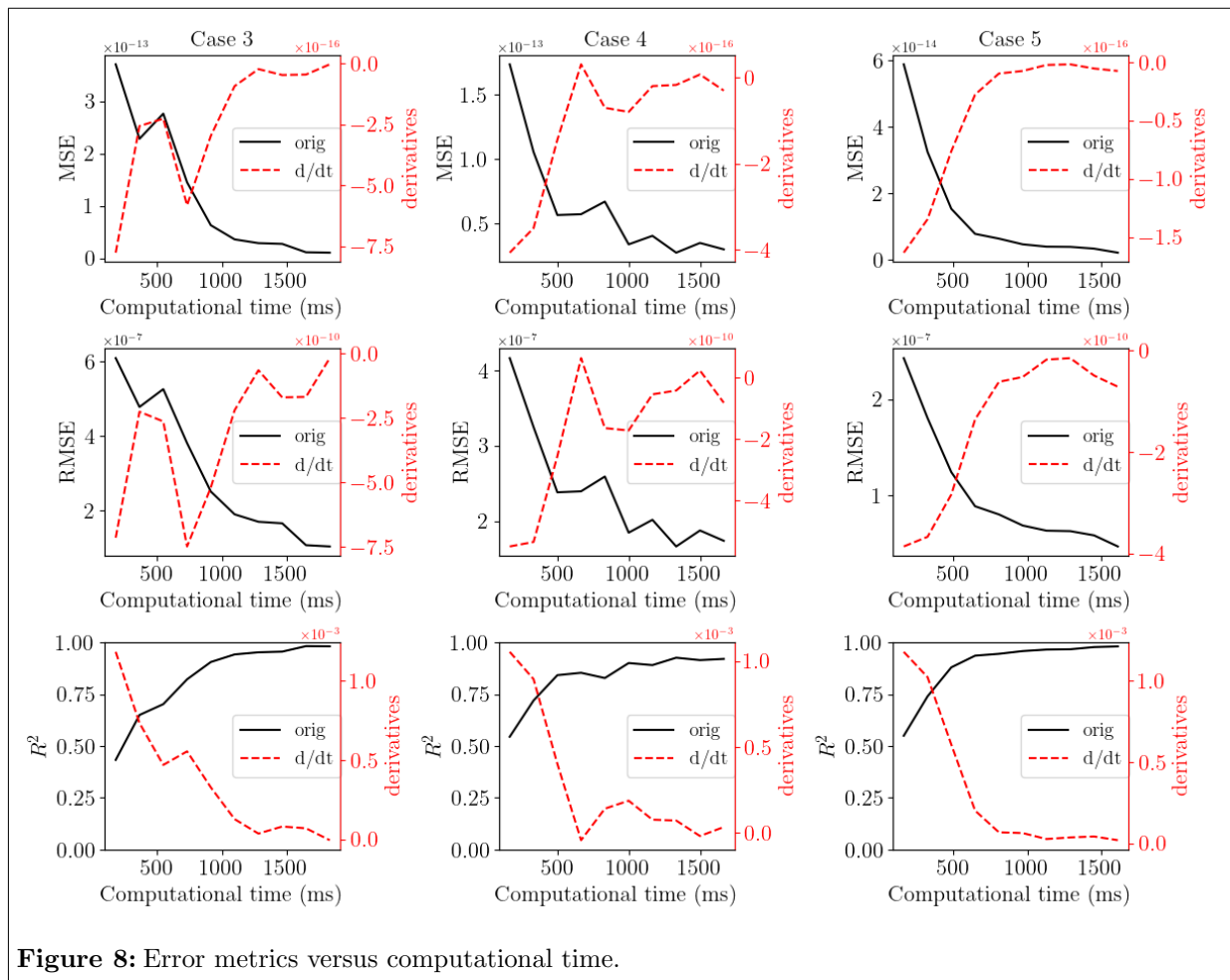
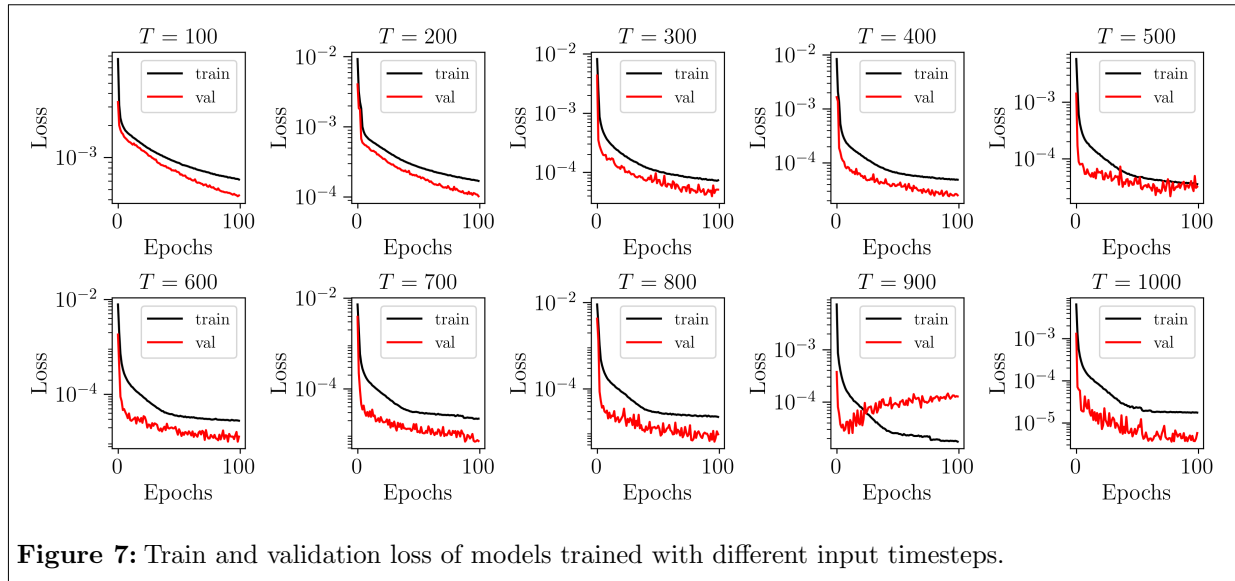
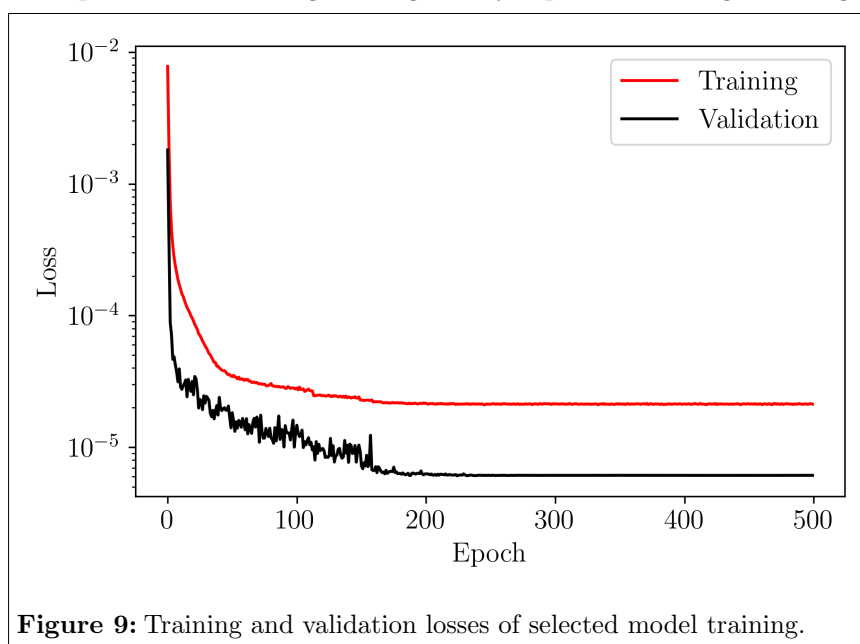


Figure 7 shows the training and validation losses. Models trained on longer input sequences generally achieve lower validation losses. The  $T = 900$  case is an exception, where the validation loss reaches a minimum early and then rises while the training loss continues to decrease, indicating overfitting. Since the neighboring  $T = 800$  and  $T = 1000$  models use the same architecture and do not exhibit this behavior, we attribute the  $T = 900$  divergence to training run variance.

Figure 8 illustrates FHKN predictive accuracy against the computational time required for the initial LB iterations. Across test cases 3–5, mean squared error (MSE) and root mean squared error (RMSE) decrease rapidly at low computational times before saturating. The coefficient of determination ( $R^2$ ) provides a scale-independent metric, rising sharply and stabilizing near unity. Establishing  $R^2 = 0.9$  as a threshold for acceptable accuracy, the models reach this target at an input generation cost of approximately 1000 ms. Beyond this point, the rate of change drops below  $10^{-4}$ , indicating that essential flow-field dynamics are captured and further LB input iterations yield negligible improvements.

## 6. RESULTS AND DISCUSSIONS

Analysis of the data in Figure 8 indicated that 1000 ms of LB computation—approximately 600 iterations—provides sufficient data for the FHKN network. Following this determination, the selected model was trained for 500 epochs. The resulting training history is presented in Figure 9. Figure 10 illustrates

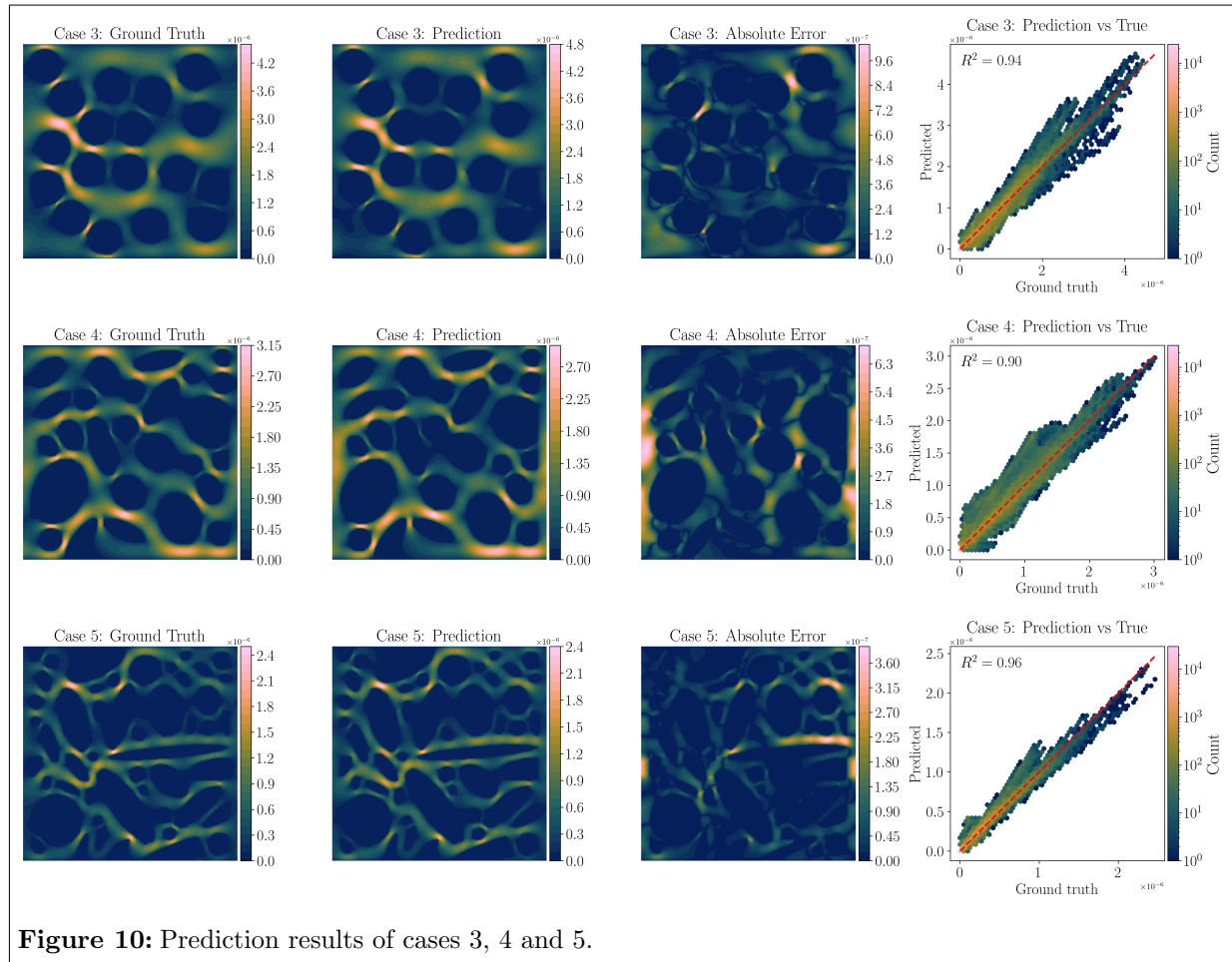


a comparison between LBM-generated velocity magnitude fields, acting as the ground truth, and the fields predicted by FHKN for three representative cases. The figure also includes absolute error maps and pixel-wise scatter density plots. This analysis evaluates the precision with which FHKN reconstructs steady-state flow from truncated LBM data. The observed visual and statistical consistency between LBM and FHKN outputs suggests that the model successfully identifies global and local flow features across different geometries and flow states.

In the three cases shown, FHKN velocity fields (Figure 10, second column) replicate the spatial configurations of the LBM reference (first column). The model preserves high-velocity channels as well as low-flow regions. The absolute error maps in the third column indicate that discrepancies are mostly confined to pore boundaries or areas with steep velocity gradients. These regions are naturally difficult to model due to nonlinear interactions. However, total error remains low—typically an order of magnitude smaller than the predicted values, demonstrating stable performance.

The scatter plots in the final column of Figure 10 confirm this quantitative precision. Data points align with the 1:1 diagonal, and  $R^2$  values between 0.90 and 0.96 indicate a high correlation with the LBM ground truth. Overall, case 5 achieves the highest  $R^2$ , while case 4 is the lowest. The convergence profiles in Figure 11, where the residual is the L2 norm of the difference between the velocity magnitude at each iteration and the steady-state value, help explain this. The case 4 profile shows a non-monotonic rise near iteration 650, outside the network input window, which is indicative of transient backflow. A similar feature appears in cases 3 and 5, but it is more subtle and occurs earlier, within the network input.

This is consistent with case 5 achieving the highest prediction accuracy and case 4 the lowest, since the backflow feature in case 4 falls outside the transient window available to the network. Collectively, these findings show that FHKN reaches near-physical fidelity despite limited temporal input. The model serves as an efficient tool for inferring steady-state flow fields while lowering computational requirements compared to full LBM simulations. Table 4 details the quantitative results for the Fourier Network



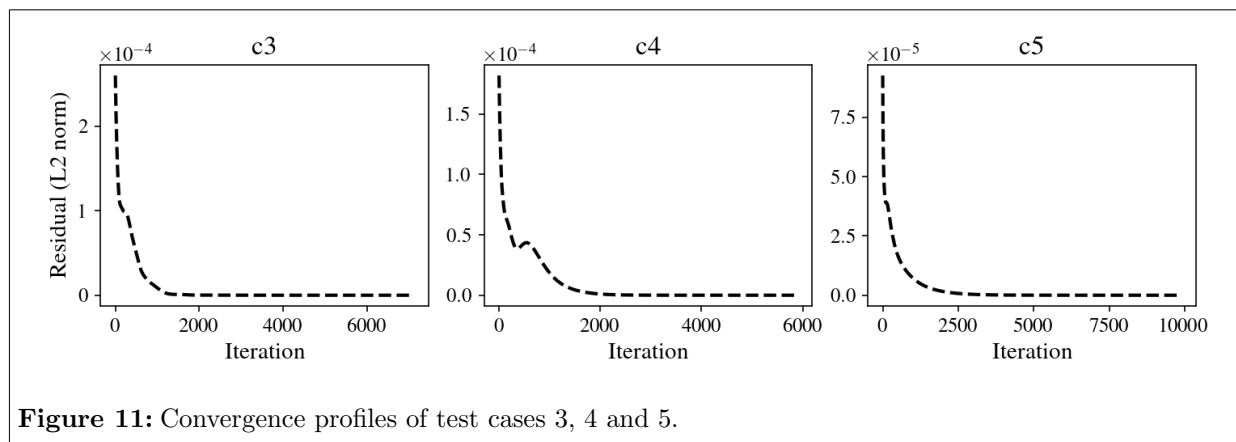
**Figure 10:** Prediction results of cases 3, 4 and 5.

(FN), Hermite Network (HN), and FHKN models. Accuracy was measured via the  $R^2$  coefficient and relative  $L_2$  error between the predicted and LBM-derived velocity fields. FHKN consistently yielded the highest  $R^2$  and lowest  $L_2$  values across all resolutions ( $256^2$ ,  $512^2$ , and  $1024^2$ ). It is worth mentioning that we achieve mesh refinement through cell division. Each cell is split into four, so that a  $256^2$  domain at 1 nm becomes  $512^2$  at 0.5 nm, and a further division yields  $1024^2$  at 0.25 nm. To keep the simulation results consistent, the body force is reduced by the same factor as the resolution. Hybrid framework provides a superior mapping from transient states to steady-state solutions. These performance gains were systematic across test cases, implying that improvements are due to the network architecture rather than training stochasticity. While Fourier Network performed well, it showed greater variability between cases and reduced accuracy at higher resolutions. Hermite Network remained competitive in specific instances but proved sensitive to geometric complexity. On average, FHKN maintained an  $R^2$  of approximately 0.93, surpassing FN and HN by roughly 0.02–0.03. Its relative  $L_2$  errors were the lowest, ranging from 0.16 to 0.17.

The integration of Fourier and Hermite representations drives these results. Fourier transforms capture large-scale spatial correlations, while Hermite expansions efficiently describe local velocity-space moments. Their combination in FHKN allows for a hierarchical representation of the space-velocity dynamics in Boltzmann transport, resulting in physically consistent reconstructions. Furthermore, FHKN showed minimal dependence on spatial resolution, suggesting strong generalization on grids four times finer than

**Table 4:** Comparison of FN, FHKN, and HN across resolutions. Metrics reported are  $R^2$  and Relative  $L_2$  error.

| Resolution         | Metric               | Case    | FN    | FHKN  | HN    |
|--------------------|----------------------|---------|-------|-------|-------|
| $256 \times 256$   | $R^2$                | c3      | 0.93  | 0.94  | 0.95  |
|                    |                      | c4      | 0.85  | 0.90  | 0.83  |
|                    |                      | c5      | 0.95  | 0.96  | 0.96  |
|                    |                      | average | 0.910 | 0.933 | 0.913 |
|                    | Relative $L_2$ error | c3      | 0.16  | 0.15  | 0.13  |
|                    |                      | c4      | 0.24  | 0.19  | 0.26  |
|                    |                      | c5      | 0.13  | 0.13  | 0.12  |
|                    |                      | average | 0.177 | 0.157 | 0.170 |
| $512 \times 512$   | $R^2$                | c3      | 0.93  | 0.94  | 0.95  |
|                    |                      | c4      | 0.85  | 0.90  | 0.82  |
|                    |                      | c5      | 0.95  | 0.96  | 0.96  |
|                    |                      | average | 0.910 | 0.933 | 0.910 |
|                    | Relative $L_2$ error | c3      | 0.17  | 0.15  | 0.13  |
|                    |                      | c4      | 0.24  | 0.19  | 0.26  |
|                    |                      | c5      | 0.14  | 0.14  | 0.13  |
|                    |                      | average | 0.183 | 0.160 | 0.173 |
| $1024 \times 1024$ | $R^2$                | c3      | 0.93  | 0.93  | 0.95  |
|                    |                      | c4      | 0.85  | 0.90  | 0.83  |
|                    |                      | c5      | 0.95  | 0.95  | 0.96  |
|                    |                      | average | 0.910 | 0.927 | 0.913 |
|                    | Relative $L_2$ error | c3      | 0.17  | 0.16  | 0.14  |
|                    |                      | c4      | 0.25  | 0.20  | 0.27  |
|                    |                      | c5      | 0.16  | 0.15  | 0.14  |
|                    |                      | average | 0.193 | 0.170 | 0.183 |



the training set. In contrast, FN and HN exhibited declining  $R^2$  and rising  $L_2$  errors during refinement. Consequently, FHKN provides a balanced solution for spatial generalization, stability, and accuracy. Tables 5 and 6 compare runtime performance for the LBM solver and the neural network surrogates (FN, FHKN, HN) across varying resolutions and hardware. LBM simulations were performed on a 28-core Intel Xeon Gold processor. Surrogate inference was tested on both the CPU (Table 5) and an NVIDIA RTX A6000 GPU (Table 6). Results represent the mean of three trials. Hybrid entries (e.g., FHKN+LBM) include the pure inference time plus the LBM prefix steps (600, 1200, and 2400 for increasing grid sizes) needed for input generation.

The optimized C++ LBM solver is efficient on the CPU, completing faster than neural network at  $256^2$  and for some  $512^2$  cases. This performance is due to low-level optimizations and hybrid MPI/OpenMP parallelization. Conversely, neural networks are hindered on the CPU by Python and framework overhead at smaller problem sizes. However, at  $1024^2$  resolution, neural surrogates outperform the full LBM solver

**Table 5:** CPU Runtime comparison between LBM and neural models (FN, FHKN, HN) across test cases and resolutions. All times are in seconds.

| Case | Resolution | Active cells | Total LBM iterations | LBM wall-time | Pure FN wall-time | FN+LBM wall-time | Pure FHKN wall-time | FHKN+LBM wall-time | Pure HN wall-time | HN+LBM wall-time |
|------|------------|--------------|----------------------|---------------|-------------------|------------------|---------------------|--------------------|-------------------|------------------|
| c1   | 256        | 39152        | 7811                 | 6.40          | 11.70             | 12.19            | 15.68               | 16.17              | 8.30              | 8.79             |
|      | 512        | 154986       | 14392                | 44.60         | 44.65             | 58.72            | 60.00               | 63.72              | 34.03             | 37.75            |
|      | 1024       | 624890       | 28340                | 454.69        | 207.10            | 245.61           | 272.70              | 311.21             | 150.11            | 188.62           |
| c2   | 256        | 43359        | 7558                 | 7.10          | 13.104            | 13.70            | 16.86               | 17.42              | 9.00              | 9.56             |
|      | 512        | 171936       | 13889                | 56.10         | 55.00             | 59.85            | 67.50               | 71.42              | 41.33             | 46.18            |
|      | 1024       | 692324       | 28087                | 452.89        | 219.61            | 258.31           | 296.28              | 334.98             | 166.45            | 205.15           |
| c3   | 256        | 41286        | 8418                 | 7.20          | 12.50             | 13.01            | 16.70               | 17.21              | 8.68              | 9.19             |
|      | 512        | 164196       | 18436                | 70.00         | 51.76             | 56.32            | 67.04               | 71.60              | 40.61             | 45.17            |
|      | 1024       | 659708       | 28462                | 463.00        | 209.75            | 248.79           | 272.07              | 311.11             | 157.94            | 196.98           |
| c4   | 256        | 37059        | 5891                 | 4.70          | 10.12             | 10.60            | 14.58               | 15.06              | 7.84              | 8.32             |
|      | 512        | 147077       | 15540                | 4.40          | 46.00             | 46.34            | 61.00               | 61.34              | 35.30             | 35.64            |
|      | 1024       | 591865       | 25443                | 349.10        | 177.90            | 210.83           | 243.31              | 276.24             | 140.05            | 172.98           |
| c5   | 256        | 36258        | 10645                | 8.60          | 9.76              | 10.24            | 14.23               | 14.71              | 7.51              | 7.99             |
|      | 512        | 143506       | 19738                | 54.00         | 45.40             | 48.68            | 59.52               | 62.41              | 33.41             | 36.69            |
|      | 1024       | 578682       | 40330                | 589.33        | 173.02            | 208.09           | 239.44              | 274.51             | 139.10            | 174.17           |

by roughly  $1.5\times$  to  $3\times$ . In these cases, Hermite Network is the fastest and FHKN the slowest, although FHKN exhibits superior generalization across new flow regimes.

On the GPU, performance differs significantly. Pure inference times fall to the millisecond range, and hybrid workflows remain over ten times faster than full LBM convergence. For  $1024^2$  grids, GPU-based inference achieves speedups between  $10\times$  and  $1000\times$ . While FHKN requires slightly more inference time than its counterparts, it remains more stable under extrapolative conditions. In summary, traditional solvers are optimal for smaller domains on CPU hardware, while neural networks—particularly when GPU-accelerated—become the superior choice for high-resolution simulations.

## 6.1. Blind Tests

To rigorously evaluate the zero-shot generalization of the FHKN architecture, two complex porous media domains ( $1024 \times 1024$ ) were generated from random Gaussian noise using PoreSpy [8] as shown in Figure 12. LBM simulations of methane transport were conducted at 2 MPa and 300 K. While the Knudsen number distributions remain consistent with the training set, the topological complexity is significantly higher, featuring highly tortuous pathways and dead-end pores. Consequently, reaching a converged steady state required 300,000 iterations for Case 1 and 66,000 iterations for Case 2. The FHKN model was restricted to using only the first 2,400 transient iterations as input to predict the final steady-state fields as shown in Figures 13 and 14. The model achieves an  $R^2$  of 0.93 for both cases. The point-wise absolute errors are marginally higher than in the in-distribution test cases, which is a direct consequence of the massive temporal gap between the early-transient input and the true steady state.

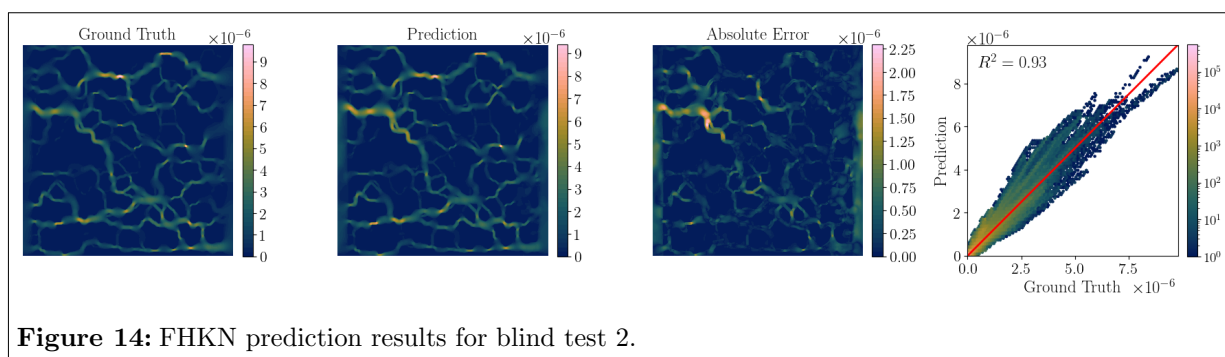
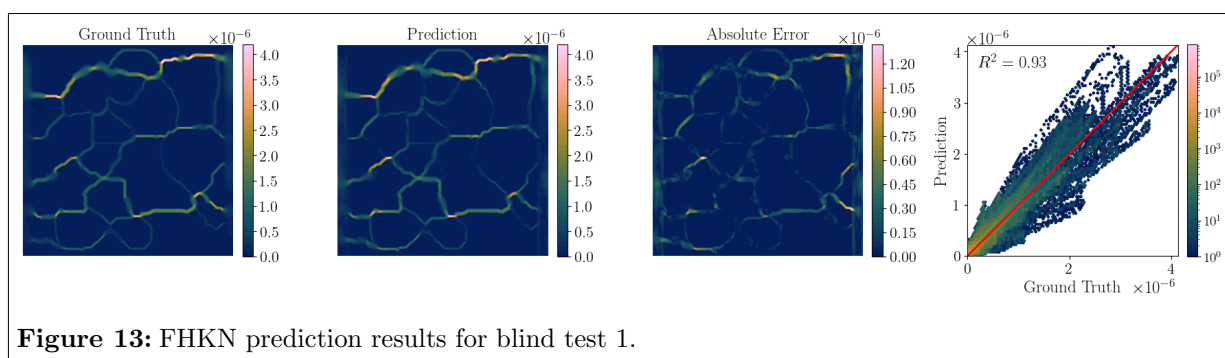
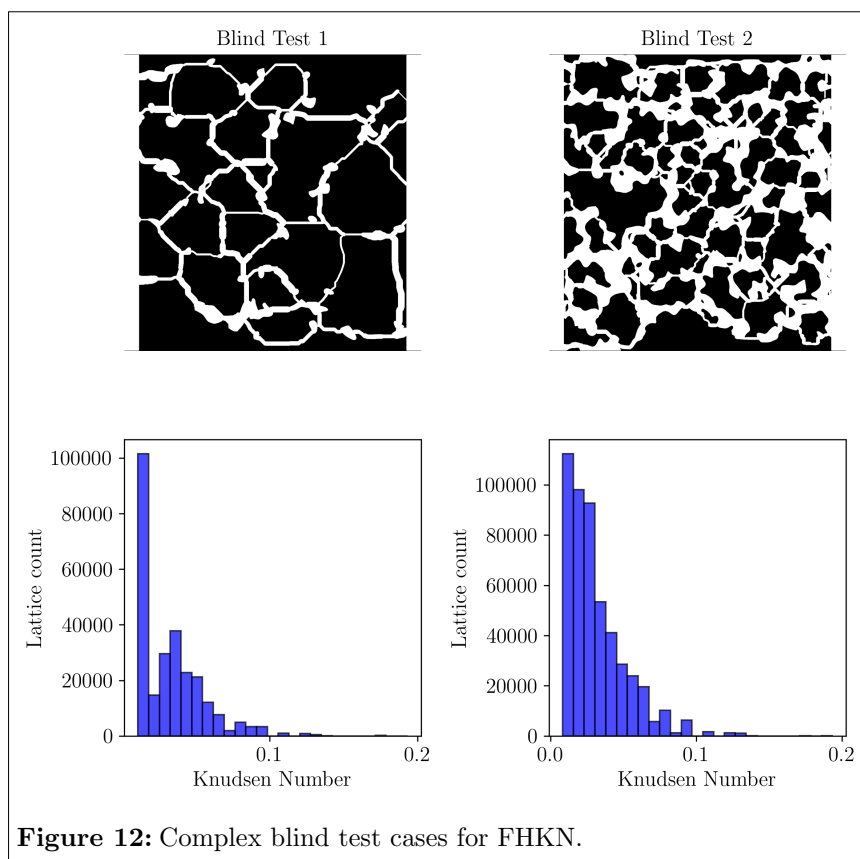
**Table 6:** Runtime comparison between LBM (CPU) and neural network models (FN, FHKN, HN) with GPU surrogate inference, across test cases and resolutions. All times are in seconds.

| Case | Resolution | Active cells | Total LBM iterations | LBM wall-time | Pure FN wall-time | FN+LBM wall-time | Pure FHKN wall-time | FHKN+LBM wall-time | Pure HN wall-time | HN+LBM wall-time |
|------|------------|--------------|----------------------|---------------|-------------------|------------------|---------------------|--------------------|-------------------|------------------|
| c1   | 256        | 39152        | 7811                 | 6.40          | 0.0072            | 0.50             | 0.010               | 0.50               | 0.004             | 0.50             |
|      | 512        | 154986       | 14392                | 44.60         | 0.0293            | 3.75             | 0.041               | 3.76               | 0.014             | 3.73             |
|      | 1024       | 624890       | 28340                | 454.69        | 0.2052            | 38.71            | 0.288               | 38.79              | 0.109             | 38.61            |
| c2   | 256        | 43359        | 7558                 | 7.10          | 0.013             | 0.58             | 0.016               | 0.58               | 0.007             | 0.57             |
|      | 512        | 171936       | 13889                | 56.10         | 0.0328            | 4.88             | 0.042               | 4.89               | 0.016             | 4.86             |
|      | 1024       | 692324       | 28087                | 452.89        | 0.2323            | 38.93            | 0.317               | 39.02              | 0.122             | 38.82            |
| c3   | 256        | 41286        | 8418                 | 7.20          | 0.011             | 0.52             | 0.014               | 0.53               | 0.006             | 0.52             |
|      | 512        | 164196       | 18436                | 70.00         | 0.0317            | 4.59             | 0.041               | 4.60               | 0.015             | 4.57             |
|      | 1024       | 659708       | 28462                | 463.00        | 0.227             | 39.27            | 0.295               | 39.34              | 0.116             | 39.16            |
| c4   | 256        | 37059        | 5891                 | 4.70          | 0.007             | 0.49             | 0.009               | 0.49               | 0.003             | 0.48             |
|      | 512        | 147077       | 15540                | 4.40          | 0.0289            | 0.37             | 0.038               | 0.38               | 0.014             | 0.35             |
|      | 1024       | 591865       | 25443                | 349.10        | 0.1977            | 33.13            | 0.262               | 33.19              | 0.103             | 33.03            |
| c5   | 256        | 36258        | 10645                | 8.60          | 0.007             | 0.49             | 0.009               | 0.49               | 0.030             | 0.51             |
|      | 512        | 143506       | 19738                | 54.00         | 0.0286            | 3.31             | 0.038               | 3.32               | 0.015             | 3.30             |
|      | 1024       | 578682       | 40330                | 589.33        | 0.1929            | 35.26            | 0.263               | 35.33              | 0.101             | 35.17            |

**Table 7:** Permeability prediction from transient and steady-state LBM and FHKN in  $m^2$ .

|        | LBM (Transient) | LBM (steady) | FHKN      | Error (FHKN) % | Error (Transient) % |
|--------|-----------------|--------------|-----------|----------------|---------------------|
| Case 1 | 3.147e-19       | 2.602e-19    | 2.764e-19 | 6.226          | 20.742              |
| Case 2 | 6.612e-19       | 5.730e-19    | 6.255e-19 | 9.162          | 15.393              |

Visual agreement of velocity fields is an insufficient metric for evaluating fluid dynamics models; the defining metric is macroscopic transport capacity, specifically permeability. To quantify the practical utility of the FHKN, macroscopic permeability was calculated from three states: the early-transient LBM at iteration 2,400 (the network input), the fully converged steady-state LBM (the ground truth), and the FHKN prediction. As shown in Table 7, extracting permeability directly from the transient LBM yields errors of 20% for Case 1 and 15% for Case 2. The larger baseline error in Case 1 correlates directly with its longer convergence requirement. In contrast, the FHKN successfully maps the transient input to the steady-state limit, reducing the permeability prediction error to 6% and 9% for Cases 1 and 2, respectively. This demonstrates the model's capacity to predict macroscopic steady-state permeability within a 10% error margin using only a fraction of the computational iterations required by the traditional solver.



## 6.2. Limitations and Future Work

While the proposed sequence-to-state architecture demonstrates that integrating native kinetic networks (Fourier and Hermite expansions) significantly accelerates steady-state prediction, several limitations remain inherent to the current proof-of-concept formulation.

First, the model is strictly a point-wise mapping. Because the network processes isolated temporal sequences ( $9 \times N_t$  tensors) to predict local macroscopic variables, it inherently lacks spatial coupling. Consequently, the network does not natively enforce macroscopic conservation laws. In its current form, introducing a divergence penalty during training is intractable because the isolated lattice input structure prevents the calculation of finite-difference spatial derivatives. We note, however, that despite the absence of an explicit conservation constraint, the low permeability error reported in the blind tests confirms that these point-wise predictions integrate into an accurate non-local transport coefficient, indicating that the point-wise errors do not accumulate destructively.

Second, the architecture is built on the existence of a stationary steady-state limit. By utilizing a 1D temporal Fast Fourier Transform to extract relaxation frequencies, the model assumes the transient kinetic noise will uniformly decay. This formulation is highly effective for laminar flows in porous media but the network is incapable of predicting unsteady high-Reynolds-number flows.

Furthermore, the training domains are random granular packs and the blind-test cases are generated from Gaussian noise. In both cases these domains are more porous and better connected than real shale. Assessing the model on real shale images therefore remains both a limitation and a direction for future work, since such structures exhibit lower porosity and isolated or dead-end pores, which significantly increase the time required to reach steady state. While one might argue that a model trained on random packed media and is able to generalize to more tortuous blind tests would also generalize to realistic shale, this is difficult to claim confidently without direct testing. The blind tests represent shale in terms of pore-wall roughness and tortuosity, but the full complexity of shale remains under-represented. Because prediction accuracy degrades with longer convergence times, we would overall expect somewhat reduced accuracy on realistic shale geometries.

Finally, the present study is focused on two-dimensional systems, which are enough to establish and validate the FHKN mapping method. The formulation extends to three dimensions in principle, but the larger discrete velocity set and the correspondingly higher-dimensional moment space increase the state size, so the offline generation, storage, and training of snapshots scale up substantially. This is a resource cost confined to the one-time training phase. The inference-time advantage of predicting the steady state from a short transient applies equally, and likely more strongly, in three dimensions.

Future work will proceed along the following directions. First, to address the lack of spatial coupling, the architecture can be extended from a purely localized temporal model to a spatially aware framework. Potential approaches include embedding the point-wise kinetic predictions into a Graph Neural Network (GNN) to explicitly penalize mass-flux imbalances across adjacent nodes, or incorporating spatial coordinates into the input tensor to enable automatic differentiation (autograd) for a Physics-Informed Neural Network (PINN) divergence loss. Enforcing strict global mass conservation without sacrificing the speed of the localized kinetic extraction remains the primary objective for subsequent iterations. Second, the model should be validated on realistic shale microstructures derived from SEM or FIB-SEM imaging, which exhibit the lower porosity, poorer connectivity, and smaller throats that are under-represented in the current domains. This will test whether the transient inputs the network relies on remain informative as convergence times lengthen and as local Knudsen numbers move toward the edge of the trained distribution. Finally, extending the framework to three-dimensional lattices, which is mechanically direct but considerably more demanding in data generation, storage, and training, would complete the transition from proof of concept to a general-purpose surrogate for kinetic transport.

## 7. SUMMARY

This study established the efficacy of the Fourier-Hermite Kinetic Network for mapping the transient prefix of high Knudsen number gas transport simulations to their corresponding steady-state solutions. By utilizing the first several hundred iterations of a Lattice Boltzmann Method (LBM) simulation as

input, the FHKN framework successfully reconstructs steady-state velocity fields in complex, synthetically generated porous media. The development of FHKN as motivated by Fourier Neural Operator (FNO) addresses the need to capture the unique dynamics of the Boltzmann distribution functions in rarefied environments like shales and tight nano-porous systems. While Fourier transforms are well-suited for capturing global spatial correlations, the integration of Hermite transforms allows the model to efficiently represent the velocity-space moments that are critical in rarefied flow regimes.

A comparative analysis of three surrogate architectures—Fourier, Hermite, and FHKN—revealed distinct trade-offs between numerical accuracy and computational demand. Quantitative results demonstrated that while Hermite Network generally offers higher precision than Fourier network alone, the hybrid FHKN architecture provides the highest accuracy across all tested cases. Specifically, FHKN achieved a superior average relative  $L_2$  error of approximately 0.16. This performance gain justifies the architectural complexity of the hybrid network, as it more accurately represents the coupled space-velocity dependencies inherent in the Boltzmann transport equation.

Runtime benchmarks indicated that while HN is the most computationally efficient on CPU architectures, and FN offers a moderate balance of speed and precision, FHKN requires the longest execution time. However, this discrepancy in computational overhead is significantly mitigated when inference is performed on GPU hardware. Under GPU acceleration, the execution time differences between the three models become negligible. Consequently, the superior generalization capabilities of FHKN—particularly its ability to maintain accuracy across unseen geometries and higher spatial resolutions—position it as the most robust choice for high-fidelity surrogates.

The tests performed on more complex geometries showcase the engineering utility of the proposed surrogate model achieving under 9% error in permeability prediction.

Ultimately, this work demonstrates that kinetic networks can effectively bypass the high computational costs of reaching steady-state convergence in LBM. The ability of FHKN to generalize from coarse training data to high-resolution test cases suggests a scalable pathway for predicting the permeability of porous media.

## STATEMENTS AND DECLARATIONS

### Supplementary Material

All supplementary or supporting information provided by the authors will be published online alongside the main article. When such material is available, it will also be indicated here along with a link to the pdf for downloading.

## Acknowledgments

The corresponding author gratefully acknowledges support from the Mowry Shale initiative at the School of Energy Resources, University of Wyoming.

## Author Contributions

Nijat Rustamov: Conceptualization, methodology; software, investigation, data curation, validation, visualization, writing—original draft preparation. Saman A Aryana; Conceptualization, supervision, project administration, funding acquisition, writing—review and editing.

## Conflicts of Interest

The authors declare no conflict of interest

## Data and Code Availability

Training, testing, and blind test curated datasets are archived at the Wyoming Data Repository and will be publicly accessible at: xxxx. The model code and implementation are available on GitHub at: xxxx.

## Funding Received

This work was supported by the School of Energy Resources Mowry Shale initiative at the University of Wyoming

## Declaration of Generative AI and AI-Assisted Technologies in the Manuscript Preparation Process

During the preparation of this manuscript, the first author used Gemini for editorial revision. The authors reviewed and verified the output of these tools and take full responsibility for the content of the published article.

## ORCID IDs

Nijat Rustamov  <https://orcid.org/0009-0000-2278-8060>  
 Saman A. Aryana  <https://orcid.org/0000-0001-9509-7420>

## REFERENCES

- Behbood Abedi, Alirza Orujov, Ehsan Dabbaghi, Kam Ng, John Ackerman, and Saman A. Aryana. Containment strategy for subsurface h2 storage based on time-dependent soft solids. *International Journal of Hydrogen Energy*, 82:1001–1014, 9 2024. URL: <http://dx.doi.org/10.1016/j.ijhydene.2024.07.336>, doi:10.1016/j.ijhydene.2024.07.336.
- Alankaa Al-Harbi, Mohammed Al-Marri, Giuliano Carchini, Mohammed Saad, and Ibnelwaleed A. Hussein. Hydrogen underground storage in silica-clay shales: Experimental and density functional theory investigation. *ACS Omega*, 8(48):45906–45913, November 2023. URL: <http://dx.doi.org/10.1021/acsomega.3c06644>, doi:10.1021/acsomega.3c06644.
- Johan Augusto Bocanegra, Mario Misale, and Davide Borelli. A systematic literature review on lattice boltzmann method applied to acoustics. *Engineering Analysis with Boundary Elements*, 158:405–429, January 2024. URL: <http://dx.doi.org/10.1016/j.enganabound.2023.11.007>, doi:10.1016/j.enganabound.2023.11.007.
- Andreas Busch, Sascha Alles, Yves Gensterblum, Dirk Prinz, David N. Dewhurst, Mark D. Raven, Helge Stanjek, and Bernhard M. Krooss. Carbon dioxide storage potential of shales. *International Journal of Greenhouse Gas Control*, 2(3):297–308, July 2008. URL: <http://dx.doi.org/10.1016/j.ijggc.2008.03.003>, doi:10.1016/j.ijggc.2008.03.003.
- Francisco J. Carrillo, Ian C. Bourg, and Cyprien Soulaine. Multiphase flow modeling in multiscale porous media: An open-source micro-continuum approach. *Journal of Computational Physics: X*, 8:100073, 9 2020. URL: <http://dx.doi.org/10.1016/j.jcpx.2020.100073>, doi:10.1016/j.jcpx.2020.100073.
- Shuai Chen and Fushen Liu. On gas transport and permeability in shale across Knudsen regimes. *Acta Geotechnica*, 21(2):1011–1030, nov 12 2025. URL: <http://dx.doi.org/10.1007/s11440-025-02819-0>, doi:10.1007/s11440-025-02819-0.
- Nishanth Dongari and Amit Agrawal. Modeling of navier–stokes equations for high knudsen number gas flows. *International Journal of Heat and Mass Transfer*, 55(15-16):4352–4358, 7 2012. URL: <http://dx.doi.org/10.1016/j.ijheatmasstransfer.2012.04.002>, doi:10.1016/j.ijheatmasstransfer.2012.04.002.
- Jeff Gostick, Zohaib Khan, Thomas Tranter, Matthew Kok, Mehrez Agnaou, Mohammadamin Sadeghi, and Rhodri Jervis. Porespy: A Python Toolkit for Quantitative Analysis of Porous Media Images. *Journal of Open Source Software*, 4(37):1296, may 1 2019. URL: <http://dx.doi.org/10.21105/joss.01296>, doi:10.21105/joss.01296.
- Zhaoli Guo, Baochang Shi, T. S. Zhao, and Chuguang Zheng. Discrete effects on boundary conditions for the lattice boltzmann equation in simulating microscale gas flows. *Physical Review E*, 76(5), 11 2007. URL: <http://dx.doi.org/10.1103/PhysRevE.76.056704>, doi:10.1103/physreve.76.056704.

10. Zhaoli Guo, Chuguang Zheng, and Baochang Shi. Lattice boltzmann equation with multiple effective relaxation times for gaseous microscale flow. *Physical Review E*, 77(3), 3 2008. URL: <http://dx.doi.org/10.1103/PhysRevE.77.036707>, doi:10.1103/physreve.77.036707.
11. Xiaoyi He and Li-Shi Luo. Theory of the lattice boltzmann method: From the boltzmann equation to the lattice boltzmann equation. *Phys. Rev. E*, 56:6811–6817, Dec 1997. URL: <https://link.aps.org/doi/10.1103/PhysRevE.56.6811>, doi:10.1103/PhysRevE.56.6811.
12. Peng Hou, Feng Gao, Jian He, Jia Liu, Yi Xue, and Zhaopeng Zhang. Shale gas transport mechanisms in inorganic and organic pores based on lattice boltzmann simulation. *Energy Reports*, 6:2641–2650, 11 2020. URL: <http://dx.doi.org/10.1016/j.egy.2020.09.021>, doi:10.1016/j.egy.2020.09.021.
13. Tianli Hu, Chengsheng Wu, Jun Ding, Xing Wang, Yu Yang, and Jianchun Wang. Data-driven flow initialization framework for CFD acceleration of underwater vehicle in vertical-plane oblique motion. *Ocean Engineering*, 362:126291, 7 2026. URL: <http://dx.doi.org/10.1016/j.oceaneng.2026.126291>, doi:10.1016/j.oceaneng.2026.126291.
14. Zuhao Kou and Haitao Wang. Transient pressure analysis of a multiple fractured well in a stress-sensitive coal seam gas reservoir. *Energies*, 13(15):3849, 7 2020. URL: <http://dx.doi.org/10.3390/en13153849>, doi:10.3390/en13153849.
15. Utpalendu Kuila and Manika Prasad. Specific surface area and poresize distribution in clays and shales. *Geophysical Prospecting*, 61(2):341–362, jan 29 2013. URL: <http://dx.doi.org/10.1111/1365-2478.12028>, doi:10.1111/1365-2478.12028.
16. Jonas Latt, Orestis Malaspinas, Dimitrios Kontaxakis, Andrea Parmigiani, Daniel Lagrava, Federico Brogi, Mohamed Ben Belgacem, Yann Thorimbert, Sébastien Leclaire, Sha Li, Francesco Marson, Jonathan Lemus, Christos Kotsalos, Raphaël Conradin, Christophe Coreixas, Rémy Petkantchin, Franck Raynaud, Joël Beny, and Bastien Chopard. Palabos: Parallel lattice boltzmann solver. *Computers & Mathematics with Applications*, 81:334–350, 1 2021. URL: <http://dx.doi.org/10.1016/j.camwa.2020.03.022>, doi:10.1016/j.camwa.2020.03.022.
17. Wenrui Li, Dengke Wang, and J.G. Wang. Improved mathematical model of apparent permeability: A focused study on free and multilayer adsorptive phase flow. *Journal of Natural Gas Science and Engineering*, 101:104508, 2022. URL: <https://www.sciencedirect.com/science/article/pii/S1875510022000981>, doi:10.1016/j.jngse.2022.104508.
18. Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier Neural Operator for Parametric Partial Differential Equations. 2020. URL: <https://arxiv.org/abs/2010.08895>, doi:10.48550/ARXIV.2010.08895.
19. Wei Xian Lim, Loh Sher En Jessica, Yu Lv, Adams Wai-Kin Kong, and Wai Lee Chan. Accelerating fluid simulations with graph convolution network predicted flow fields. *Aerospace Science and Technology*, 164:110414, 9 2025. URL: <http://dx.doi.org/10.1016/j.ast.2025.110414>, doi:10.1016/j.ast.2025.110414.
20. Lingfu Liu, Yuhang Wang, and Saman A. Aryana. Insights into scale translation of methane transport in nanopores. *Journal of Natural Gas Science and Engineering*, 96:104220, 12 2021. URL: <http://dx.doi.org/10.1016/j.jngse.2021.104220>, doi:10.1016/j.jngse.2021.104220.
21. Robert G. Loucks, Robert M. Reed, Stephen C. Ruppel, and Ursula Hammes. Spectrum of pore types and networks in mudrocks and a descriptive classification for matrix-related mudrock pores. *AAPG Bulletin*, 96(6):1071–1098, Jun 2012. doi:10.1306/08171111061.
22. Yashar Mehmani, Timothy Anderson, Yuhang Wang, Saman A. Aryana, Ilenia Battiato, Hamdi A. Tchelepi, and Anthony R. Kavscek. Striving to translate shale physics across ten orders of magnitude: What have we learned? *Earth-Science Reviews*, 223:103848, December 2021. URL: <http://dx.doi.org/10.1016/j.earscirev.2021.103848>, doi:10.1016/j.earscirev.2021.103848.
23. Vasilis K. Michalis, Alexandros N. Kalarakis, Eugene D. Skouras, and Vasilis N. Burganos. Rarefaction effects on gas viscosity in the knudsen transition regime. *Microfluidics and Nanofluidics*, 9(4-5):847–853, 3 2010. URL: <http://dx.doi.org/10.1007/s10404-010-0606-3>, doi:10.1007/s10404-010-0606-3.
24. Chenjia Ning, Jiaqing Kou, and Weiwei Zhang. An effective convergence accelerator of fluid simulations via generative diffusion probabilistic model. *Aerospace Science and Technology*, 158:109917, 3 2025. URL: <http://dx.doi.org/10.1016/j.ast.2024.109917>, doi:10.1016/j.ast.2024.109917.
25. 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, 2 2019. URL: <http://dx.doi.org/10.1016/j.jcp.2018.10.045>, doi:10.1016/j.jcp.2018.10.045.
26. Bastian E. Rapp. Fluids. In *Microfluidics: Modelling, Mechanics and Mathematics*, pages 243–263. Elsevier, 2017. URL: <http://dx.doi.org/10.1016/B978-1-4557-3141-1.50009-5>, doi:10.1016/b978-1-4557-3141-1.50009-5.

27. Nijat Rustamov, Lingfu Liu, and Saman A. Aryana. Scalable simulation of coupled adsorption and transport of methane in confined complex porous media with density preconditioning. *Gas Science and Engineering*, 119:205131, 11 2023. URL: <http://dx.doi.org/10.1016/j.jgsce.2023.205131>, doi:10.1016/j.jgsce.2023.205131.
28. Nijat Rustamov, Peyman Mostaghimi, and Saman A. Aryana. On multi-block lattice Boltzmann method for high Knudsen number flows. *Advances in Geo-Energy Research*, 16(2):143–157, apr 5 2025. URL: <http://dx.doi.org/10.46690/ager.2025.05.06>, doi:10.46690/ager.2025.05.06.
29. Wei Su, Scott Lindsay, Haihu Liu, and Lei Wu. Comparative study of the discrete velocity and lattice boltzmann methods for rarefied gas flows through irregular channels. *Physical Review E*, 96(2), August 2017. URL: <http://dx.doi.org/10.1103/PhysRevE.96.023309>, doi:10.1103/physreve.96.023309.
30. S. Succi. *The Lattice Boltzmann Equation*. Oxford University Press, 6 2001. URL: [https://books.google.com/books/about/The.Lattice.Boltzmann.Equation.html?hl=&id=OC0Sj\\_xgnhAC](https://books.google.com/books/about/The.Lattice.Boltzmann.Equation.html?hl=&id=OC0Sj_xgnhAC).
31. Sauro Succi. Mesoscopic modeling of slip motion at fluid-solid interfaces with heterogeneous catalysis. *Physical Review Letters*, 89(6), 7 2002. URL: <http://dx.doi.org/10.1103/PhysRevLett.89.064502>, doi:10.1103/physrevlett.89.064502.
32. K Suga. Lattice Boltzmann methods for complex micro-flows: applicability and limitations for practical applications. *Fluid Dynamics Research*, 45(3):034501, may 14 2013. URL: <http://dx.doi.org/10.1088/0169-5983/45/3/034501>, doi:10.1088/0169-5983/45/3/034501.
33. Yuhang Wang and Saman A. Aryana. Pore-scale simulation of gas flow in microscopic permeable media with complex geometries. *Journal of Natural Gas Science and Engineering*, 81:103441, 9 2020. URL: <http://dx.doi.org/10.1016/j.jngse.2020.103441>, doi:10.1016/j.jngse.2020.103441.
34. Qingsong Xu, Yilei Shi, Jonathan Bamber, Ye Tuo, Ralf Ludwig, and Xiao Xiang Zhu. Physics-aware Machine Learning Revolutionizes Scientific Paradigm for Machine Learning and Process-based Hydrology. 2023. URL: <https://arxiv.org/abs/2310.05227>, doi:10.48550/ARXIV.2310.05227.
35. Raoyang Zhang, Xiaowen Shan, and Hudong Chen. Efficient kinetic method for fluid simulation beyond the navier-stokes equation. *Physical Review E*, 74(4), 10 2006. URL: <http://dx.doi.org/10.1103/PhysRevE.74.046703>, doi:10.1103/physreve.74.046703.
36. Yulong Zhao, Mingyao Luo, Lingfu Liu, Jianfa Wu, Man Chen, and Liehui Zhang. Molecular dynamics simulations of shale gas transport in rough nanopores. *Journal of Petroleum Science and Engineering*, 217:110884, 10 2022. URL: <http://dx.doi.org/10.1016/j.petrol.2022.110884>, doi:10.1016/j.petrol.2022.110884.