

Original article

On multi-block lattice Boltzmann method for high Knudsen number flows

Nijat Rustamov¹, Peyman Mostaghimi², Saman A. Aryana^{1,2,3}✉*

¹Department of Chemical and Biomedical Engineering, University of Wyoming, Laramie 82071, USA

²School of Civil and Environmental Engineering, University of New South Wales, Sydney 2052, Australia

³School of Energy Resources, University of Wyoming, Laramie 82071, USA

Keywords:

Grid refinement
high Knudsen number
lattice Boltzmann method

Cited as:

Rustamov, N., Mostaghimi, P., Aryana, S. A. On multi-block lattice Boltzmann method for high Knudsen number flows. *Advances in Geo-Energy Research*, 2025, 16(2): 143-157.
<https://doi.org/10.46690/ager.2025.05.06>

Abstract:

This work introduces a new computational framework aimed at advancing the modeling of gas transport in confined porous media, particularly shale and tight geological formations that are characterized by their intricate network of meso- and micro-scale fractures and a broad distribution of organic pores. Accurate simulation of gas behavior in such media is challenging due to the complex interactions occurring at high Knudsen numbers, where conventional continuum-based methods fail and kinetic-theory approach becomes more suitable. To tackle these complexities, this work presents a lattice Boltzmann framework tailored for large computational domains with multi-scale pore structures from nano to micro scales. This framework incorporates slip boundary conditions and features an innovative multi-block approach to enable efficient simulations over a wide range of pore sizes, from nanometers to micrometers. The novel contributions of this work include: A scale-informed grid refinement strategy, the incorporation of shear stress terms, multi-block evolution algorithm, and a novel classification method for implementing specular reflection boundary conditions on irregular surfaces. Validation against Direct Simulation Monte Carlo and Molecular Dynamics data from the literature confirms the model's accuracy in predicting gas behavior. Simulations of methane transport in tight porous media with irregular geometries highlight the framework's effectiveness in modeling gas permeability across varying pressure conditions. Apparent permeability results across a range of Knudsen numbers demonstrate the versatility of this framework in capturing the physics of gas transport in confined porous media.

1. Introduction

Shales hold significant promise as a source of low-carbon natural gas and potential seals for the storage of carbon dioxide and hydrogen fuel, making it crucial to better understand gas flows within these confined, complex systems (Abedi et al., 2024). However, the challenges that shales present, such as heterogeneity, non-classical transport mechanisms, and phase behavior, hinder accurate simulations of flow and transport in these tight media. Shales are primarily made up of ultra-fine-grained rocks with diverse chemical and structural

features, ranging from nanometers to millimeters (Loucks et al., 2012). In the most extreme cases, nanopores can approach the size of individual gas molecules, leading to confined behaviors (Zhao et al., 2022). This confinement intensifies fluid-particle interactions, causing significant deviations from bulk fluid properties. Furthermore, studies have shown that pore roughness plays a critical role in methane transport within these irregular geometries (Liu et al., 2021). One of the main challenges in understanding fluid flow in shales is the wide range of pore sizes, from nano-scale pathways to millimeter-

scale fractures (Carrillo et al., 2020; Wu et al., 2024). As described by Mehmani et al. (2021), in the absence of a clear separation of scales, this disparity makes scale translation techniques, such as homogenization methods ineffective. In such systems, it may be necessary not only to understand the various scales individually but also to study their collective behavior, which arises from intricate intra-scale interactions.

Transport mechanisms are typically characterized by the Knudsen number (K_n), which is the ratio of the mean free path of molecules to the characteristic length of the medium (Li et al., 2022). When $K_n < 0.001$, transport can be accurately described using continuum mechanics, such as the Navier-Stokes equations with no-slip boundary conditions. In the range between $0.001 < K_n < 0.1$, the flow enters the slip flow regime, where the Navier-Stokes equations with slip boundary conditions can still provide reasonable predictions (Dongari and Agrawal, 2012). As K_n increases to 0.1, the flow transitions into a regime between slip flow and free molecular flow, where kinetic theory becomes a reliable approach for modeling gas transport (Rapp, 2016). At $K_n > 10$, transport is in the free molecular regime. Given the vast range of scales present in shale systems, a multi-scale modeling approach is essential to account for the various scales and to understand their system-level behavior (Rustamov et al., 2023; Liu et al., 2024).

The lattice Boltzmann (LB) method has emerged as a promising alternative to continuum approaches for modeling transport over the last several decades (Su et al., 2017). The LB equation is a discrete formulation derived through certain simplifications of the Boltzmann transport equation (Succi, 2001). As a mesoscopic method, the LB approach describes the evolution of probability density functions of particles and can recover macroscopic properties such as pressure and velocity (Bocanegra et al., 2023). Due to its simplicity, LB is often more computationally efficient than other numerical techniques, such as finite volume methods, and lends itself to efficient parallel algorithms (Latt et al., 2020). While LB is well-suited for bridging the gap between free molecular and continuum scales, the challenge of solving high-fidelity models across multiple scales calls for additional strategies to handle the computational cost and memory requirements. One such technique is the multi-block method, which efficiently accelerates the convergence of numerical solutions by solving the problem on non-uniform grids. Our motivation for employing multi-block methods stems from the need to handle large domain sizes and irregular geometries commonly encountered in shales and other tight porous media. In such cases, the standard LB method can become computationally expensive as finer resolutions are required to capture small-scale details accurately. A similar approach to the multi-block method is the multi-grid method, which comes in two main types: Geometric and algebraic. Geometric multi-grid methods rely on predefined, structured grids typically generated based on the geometry of the problem. These methods use known geometric information to coarsen grids and interpolate between them (Chiu and Lin, 2023). In contrast, Algebraic Multi-Grid (AMG) methods do not depend on geometric information. Instead, AMG constructs coarse grids, and the

multi-grid hierarchy based on the algebraic properties of the matrix arising from problem discretization. This flexibility enables AMG to handle unstructured grids or irregular meshes, but it can also make the problem setup more computationally expensive (Adams and Chartier, 2005). By combining the LB equation with multi-grid or multi-block approaches, the computational load can be significantly reduced, allowing for faster convergence. However, multi-block methods are better suited for complex geometries, such as those found in shales and tight systems (Zhang et al., 2013). The aim of the current study is to develop a scale-informed grid refinement strategy and a multi-block LB method capable of simulating gas transport at high Knudsen numbers in complex geometries.

Over the years, several grid refinement strategies have been proposed for LB. One of the earliest attempts to extend LB to arbitrary geometries was by Nannelli and Succi (1992), who used a finite volume discretization method to solve flow problems in relatively simple geometries. Building on this foundation, Chen et al. (2005) developed grid refinement algorithms that incorporated locally embedded grids to conserve mass and momentum through a volumetric formulation. In their method, coarse cells "explode" into finer cells while conserving mass and momentum. The distribution functions of fine cells after explosion are computed using linear interpolation from the coarse cells. Following the streaming and collision operations in fine cells, the distribution functions are recombined to update the coarse cells. However, this method is limited to relatively simple geometries and lacked a detailed discussion on convergence and runtime compared to standard LB. Later, Cheng and Hung (2004) proposed a method to decouple the computational mesh from the discretization of momentum space, where the collision and streaming steps are performed at grid points of the computational mesh, and Lagrange interpolation is used to determine the new distribution function. Unlike the approach in Chen et al. (2005), their method eliminates the need for explosion and coalescence steps since lattice points are decoupled from the computational mesh, meaning that underlying lattice structure remains uniform for streaming operations. After streaming, Lagrange interpolation is used to compute density distribution at the non-uniform mesh points before collision steps are performed. Although interpolation introduces an additional computational step, it provides greater flexibility in mesh refinement. Moreover, interpolation methods have been shown to improve the accuracy of LB when transitioning between coarse and fine grids. This was highlighted by Guzik et al. (2013) who demonstrated that advanced interpolation techniques improved both computational performance and stability in simulations. Unlike previous studies, they implement different collision operators with non-uniform meshes and provide comprehensive error analysis based on the level of refinement. Other notable contributions include the work of Liu et al. (2022) who developed an Adaptive Mesh Refinement library in C++ for complex geometries and Ahmed et al. (2023) who extended the multi-block algorithm to moving boundaries for simulating scenarios such as of settling of a solid sphere in a fluid, the motion of an insect wing in air, and the hovering of elliptic airfoil. These studies demonstrate that the LB method is fully

capable of solving complex physics on non-uniform grids with sufficiently advanced grid refinement strategies.

A key challenge in the multi-block LB method is the construction of non-uniform meshes. The quadtree structure has been widely used for mesh refinement in the literature (Perumal and Koh, 2023; Prouvost et al., 2024). In two dimensions, the quad-tree structure, and in three dimensions, the octree structure, are hierarchical data structures based on the principle of recursive decomposition. These grids can be dynamically and automatically generated using simple algorithms that apply quad-tree refinement based on specific criteria (Shephard and Georges, 1991). Chen et al. (2011) implemented a quadtree structure for refining meshes in the Lattice Boltzmann method, using linear interpolation to determine unknown distribution functions at the interface between grid cells of different levels. They implemented back-and-forth error compensation and correction, where post-collision distribution streamed back and forth using positive and negative discrete velocity vectors. This process involved linear interpolation applied to post-streaming and post-collision distributions, and normal streaming is conducted on the post-interpolation distributions. Another significant implementation of quad-tree is by Foroughi et al. (2013) who used the "cell explosion and coalescence" technique similar to Chen et al. (2005), which avoids interpolation while simulating flow in porous media. One of the most comprehensive applications of the quadtree based multi-block LB method comes from Liu et al. (2023) who introduced a novel buffer zone layout between different grid levels and a novel interpolation mechanism.

2. Simulation framework

2.1 Lattice Boltzmann method

The simulation framework is based on the Boltzmann transport equation, which is expressed as (Mohamad, 2019):

$$\frac{\partial f}{\partial t} + \mathbf{c} \frac{\partial f}{\partial \mathbf{r}} + \frac{\mathbf{F}}{m} \frac{\partial f}{\partial \mathbf{c}} = \Omega \quad (1)$$

A system's statistical representation can be described by a distribution function denoted as $f(\mathbf{r}, \mathbf{c}, t)$ which describes the number of molecules at a given time t , with position \mathbf{r} and velocity \mathbf{c} . The derivation of Eq. (1) can be found in Mohamad, 2019. When an external force \mathbf{F} is applied to a gas molecule of mass m , it alters the molecule's velocity from \mathbf{c} to $\mathbf{c} + \mathbf{F}dt$, and its position from \mathbf{r} to $\mathbf{r} + \mathbf{c}dt$. The difficulty of solving the continuous Boltzmann equation arises from the collision term Ω . He and Luo (1997) proposed the following linearized solution:

$$f_\alpha(x + \mathbf{e}_\alpha \delta t, t + \delta t) - f_\alpha(x, t) = \Omega_\alpha(f(x, t)) + \delta t F_\alpha(x, t) \quad (2)$$

Here, $f_\alpha(x, t)$ represents the density distribution function in the direction α at position x and time t , Ω_α is the collision operator, and F_α is the force term. In the two-dimensional 9-velocity D_2Q_9 lattice model, the microscopic velocity vectors are defined as (Lallemand and Luo, 2000):

$$\mathbf{e}_\alpha = \begin{cases} (0, 0) & \alpha = 0 \\ (1, 0), (0, 1), (-1, 0), (0, -1) & \alpha = 1, 2, 3, 4 \\ (1, 1), (-1, 1), (-1, -1), (1, -1) & \alpha = 5, 6, 7, 8 \end{cases} \quad (3)$$

The collision term in Eq. (2) is the multi-relaxation time collision operator (Lallemand and Luo, 2000; D'Humières, 2002), which is given by:

$$\Omega_\alpha(f(x, t)) = - \sum_{\beta} (\mathbf{M}^{-1} \mathbf{S} \mathbf{M})_{\alpha\beta} (f_\beta - f_\beta^{eq}) \quad (4)$$

where \mathbf{M} is the projection matrix that maps discrete velocities onto moment space. Collisions occur in moment space with different relaxation times for each discrete direction, and the resulting moments are transformed back into velocity space. The multi-relaxation time operator is a generalization of the well-known Bhatnagar-Gross-Krook (BGK) collision model, and it addresses numerical stability concerns. The matrix \mathbf{S} is the diagonal relaxation matrix, given as:

$$\mathbf{S} = \text{diag}(\tau_\rho, \tau_e, \tau_\epsilon, \tau_j, \tau_q, \tau_j, \tau_q, \tau_s, \tau_s)^{-1} \quad (5)$$

In Eq. (5), τ_ρ and τ_j ensure the conservation of mass and momentum and are set to 1.0. τ_e and τ_ϵ are related to energy and its square, and are set to 1.1 and 1.2, respectively. τ_s is related to shear viscosity, while τ_q relates to slip velocity. The impact of confinement on shear viscosity and slip velocity is described by the following equations (Li et al., 2010; Michalis et al., 2010):

$$\tau_s = \frac{1}{2} + \sqrt{\frac{6}{\pi}} \frac{HK_n}{1 + 2K_n} \quad (6)$$

$$\tau_q = \frac{1}{2} + \frac{3 + A_2\pi(2\tau_s - 1)^2}{8(2\tau_s - 1)} \quad (7)$$

Here, H represents the local characteristic length (i.e., pore size), K_n is the local Knudsen number, and A_2 is the coefficient for the second order slip velocity boundary condition (Wang and Aryana, 2020) which will also appear in Eq. (12) in section 2.3. The equilibrium distribution in Eq. (2) is obtained via the Hermite expansion of the Maxwell-Boltzmann distribution function (Chen et al., 1992):

$$f_\alpha^{eq}(x, t) = \omega_\alpha \rho(x, t) \left[1 + \frac{\mathbf{e}_\alpha \mathbf{u}(x, t)}{c_s^2} + \frac{(\mathbf{e}_\alpha \mathbf{u}(x, t))^2}{2c_s^4} - \frac{\mathbf{u}(x, t)^2}{2c_s^2} \right] \quad (8)$$

In the equation above, the weighting factors $\omega_\alpha = 4/9$ for $\alpha = 0$, $\omega_\alpha = 1/9$ for $\alpha = 1, \dots, 4$, and $\omega_\alpha = 1/36$ for $\alpha = 5, \dots, 8$, are for the D_2Q_9 lattice. Here, $c_s = 1/3$ represents the lattice speed of sound, and $\rho(x, t)$ and $\mathbf{u}(x, t)$ are the macroscopic density and velocity respectively.

2.2 Regularization

To minimize errors introduced by confinement, the non-equilibrium distributions are regularized by projecting them onto the second-order Hermite space using the following equation (Zhang et al., 2006):

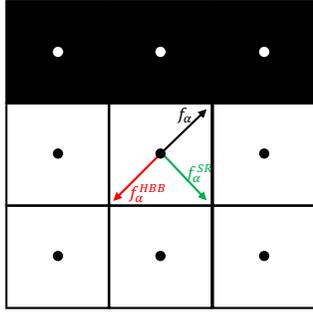


Fig. 1. Combination of halfway bounce-back and specular reflection: Black shows solids and white shows pores.

$$\tilde{f}_\alpha = \omega_\alpha \left[\frac{1}{2c_s^2} \mathcal{H}^{(n)} \left(\frac{\mathbf{e}_\alpha}{c_s} \right) : \sum_\beta f_\beta^{\text{noneq}} \mathbf{e}_\beta \otimes \mathbf{e}_\beta^T \right] \quad (9)$$

where f_β^{noneq} is the non-equilibrium part of the distribution function, and $\mathcal{H}^{(n)}$, with $n = 2$, represents the second-order Hermite polynomial expressed as:

$$\mathcal{H}^{(2)} = \mathbf{e} \otimes \mathbf{e}^T - \delta \quad (10)$$

In the above equation, δ is the Kronecker delta function. Substituting Eq. (9) into Eq. (4), Eq. (2) can be rewritten in the following form, excluding the force term:

$$f_\alpha(\mathbf{x} + \mathbf{e}_\alpha \delta t, t + \delta t) = f_\alpha^{\text{eq}} + \tilde{f}_\alpha - \sum_\beta (\mathbf{M}^{-1} \mathbf{S} \mathbf{M})_{\alpha, \beta} \tilde{f}_\beta \quad (11)$$

2.3 Boundary conditions

The second order slip boundary condition is given by:

$$u_s = A_1 K_n \left. \frac{\partial u}{\partial n} \right|_w - A_2 K_n^2 \left. \frac{\partial^2 u}{\partial n^2} \right|_w \quad (12)$$

where A_1 and A_2 are the first and second order slip coefficients, respectively, \mathbf{n} is wall normal vector and w denotes derivative of the quantity at the wall. The values of slip coefficients $A_1 = 0.6$ and $A_2 = 0.9$ are adapted from (Wang and Aryana, 2020). In confined systems, to capture this slip effect using LBM, we apply a combination of the halfway bounce-back and specular reflection boundary conditions, as proposed by (Succi, 2002; Guo et al., 2007):

$$\begin{aligned} f_\alpha^{\text{HBB}} &= r f_\alpha \\ f_\alpha^{\text{SR}} &= (1-r) f_\alpha \end{aligned} \quad (13)$$

In the above equations, f_α is the distribution function colliding with the wall, f_α^{HBB} is the halfway bounce-back component, and f_α^{SR} is the specular reflection component. A schematic of this setup is shown in Fig. 1. The combination coefficient r is derived from the second order slip boundary condition by (Guo et al., 2008):

$$r = \left[1 + \sqrt{\frac{\pi}{6}} A_1 + \frac{\tau_s'(0)}{8(\tau_s(0) - \frac{1}{2})} \right]^{-1} \quad (14)$$

where $\tau_s'(0)$ is the first-order derivative of the shear viscosity term with respect to y at the wall, hence (0). As indicated by Eq. (13), when r approaches unity, the halfway bounce-

back component dominates, leading to the disappearance of slip velocities. The f_α^{HBB} component in Fig. 1 is simply the opposite of f_α , but determining the direction of specular reflection becomes challenging in irregular geometries. Liu et al., 2021 classified the boundaries into 12 different types based on geometry and calculated the components of the basis vector using the following equation:

$$\mathbf{e}_\alpha^{\text{SR}} = \mathbf{e}_\alpha - 2(\mathbf{e}_\alpha \mathbf{n}) \mathbf{n} \quad (15)$$

where $\mathbf{e}_\alpha^{\text{SR}}$ is the basis vector for specular reflection, \mathbf{e}_α is the incoming particle direction, and \mathbf{n} is the vector normal to the boundary, which is calculated as follows:

$$\vec{n} = \frac{\sum_{\alpha=2}^9 \vec{e}_\alpha}{\left| \sum_{\alpha=2}^9 \vec{e}_\alpha \right|} \quad \text{if } \vec{x} + \vec{e}_\alpha \equiv \vec{1} \quad (16)$$

Here, \vec{x} represents the position of f_α , and $\vec{1}$ denotes the boundary position. In many cases, irregular geometries do not fit into any of the 12 boundary types identified by Liu et al. (2021). In this work, we extend the boundary classification, identifying 52 different types. As shown in Fig. 1, a 3×3 section of the image is used to determine the directions of bounce-back and specular reflection. Suppose boundary cells are marked as 0, and fluid cells as 1; the image section represents a 3×3 Boolean matrix. Column-wise linearization of this matrix produces a Boolean vector, which can be represented as a 9-digit binary number ranging from 0 (all boundary cells) to 511 (all fluid cells). Out of 512 possible cases, we select those where the central nodes are fluid and have at least one adjacent boundary cell, reducing the possibilities by half. Additionally, we ignore cases where the local pore size Eq. (1), i.e., when a fluid cell is sandwiched between two boundary cells, as this lacks sufficient resolution. For the remaining 138 cases, the normal vector direction is calculated using Eq. (16). If the magnitude of the normal vector is not a real number (due to division by zero), those cases are ignored since specular reflection cannot be determined. Otherwise, the boundary type is stored as a decimal number corresponding to its binary format, along with the axis-aligned normal and specular reflection direction vectors. This process is illustrated in Fig. 2. The stored data is later used as a lookup table for porous media simulations. We scan the large binary image in a 3×3 window, convert it to its decimal equivalent, and compare it against the pre-calculated table. If the number is not present, that section is removed from the fluid region. This approach has two major advantages. First, storing boundary types as decimal values significantly reduces the computational time required to identify the correct type and specular reflection direction. Second, it enhances the robustness of the simulations in porous media by allowing for more complex boundary shapes. Sample 3×3 images representing boundary types are shown in Fig. 3.

3. Multi-Block LB method

3.1 Resolution analysis

The Lattice Boltzmann Method is primarily designed for uniform meshes. Hence, for non-uniform meshes, special att-

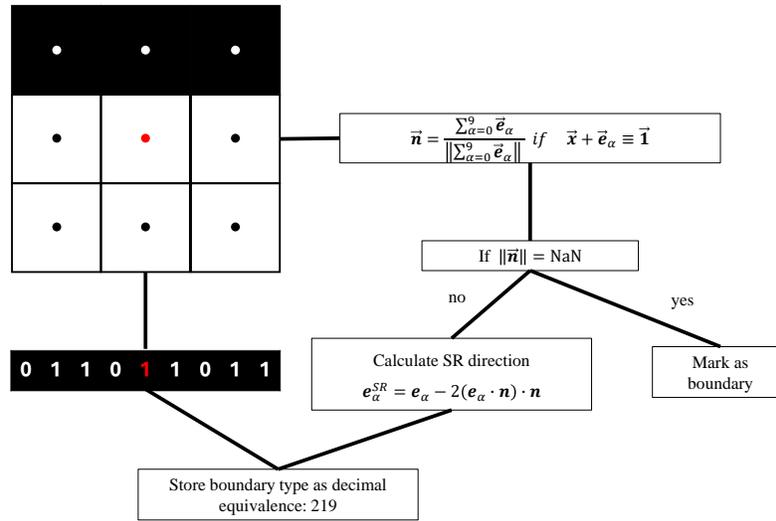


Fig. 2. Boundary classification methodology: black shows solids and white shows pores.

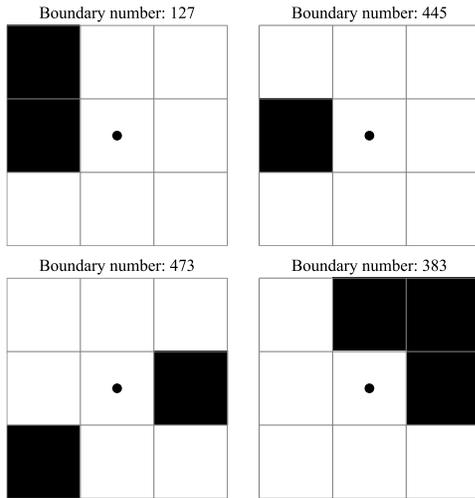


Fig. 3. Example boundary types: Black shows solids and white shows pores.

ention is required to conserve mass and momentum. To address this, we first decompose the uniform mesh using a quad-tree algorithm. The standard quad-tree algorithm, however, creates abrupt transitions at the interface between grids of different levels, which reduces the accuracy of interpolation. Additionally, tight pores require higher resolution to accurately capture slip velocity near the boundary.

When the simulation domain contains a wide range of pore sizes, the grid cell resolution *delta x*.*a.k.a* lattice spacing, must be selected according to the smallest pore, i.e., the highest Knudsen number. This implies that the required lattice spacing is inversely proportional to Knudsen number. This means that domain decomposition should account for the minimum required resolution for each pore size. To address this, we conducted a study to correlate grid resolution to Knudsen number under constant pressure and temperature. An ensemble of simulation cases for methane flow in simple channels with diameters ranging from 5 to 1,000 nm was generated. Each

channel was discretized into grids with mesh sizes ranging from 11 to 511 in the *y*-direction. The resulting grid resolution for a 5 nm channel ranged between 0.45 nm and 0.005 nm. Around 1,300 simulations were executed, and the results were collated for analysis.

As shown in Fig. 4, increasing the mesh size in the *y*-direction leads to more accurate velocity profiles, particularly near the boundary. To find the optimal mesh size, we analyzed the second-order derivative of velocity at the boundary with respect to mesh size for each channel size (see Fig. 5(a)). By selecting a threshold of 10^{-5} , we determined the required mesh size in the *y*-direction for each channel from 5 nm to 1000 nm. In the final stage of the resolution analysis, we calculated the critical resolution as D/N_y ,

where D is the channel width, and N_y is the *y*-direction mesh size, determined from the second derivative of velocity. The Knudsen number for each channel width was calculated using $Kn = \lambda/D$, where λ is the mean free path. As seen in Fig. 5(b), there is a strong correlation between the critical resolution and the Knudsen number. Thus, from Fig. 5(b), it can be concluded that:

$$\frac{D}{N_y} = 0.00956Kn^{-1} \tag{17}$$

The above correlation does not suggest that the Knudsen number is inherently related to mesh resolution or that it possesses a direct physical interpretation. Instead, it indicates that for a given Knudsen number, an adequately fine mesh is required to accurately capture transport behavior. As the Knudsen number increases, the mesh must be refined further to maintain accuracy. The purpose of the resolution analysis presented in this section is to determine the level of mesh refinement necessary for the LB method to accurately model slip velocity. In the subsequent section, this correlation will be utilized to develop scale-informed grid refinement.

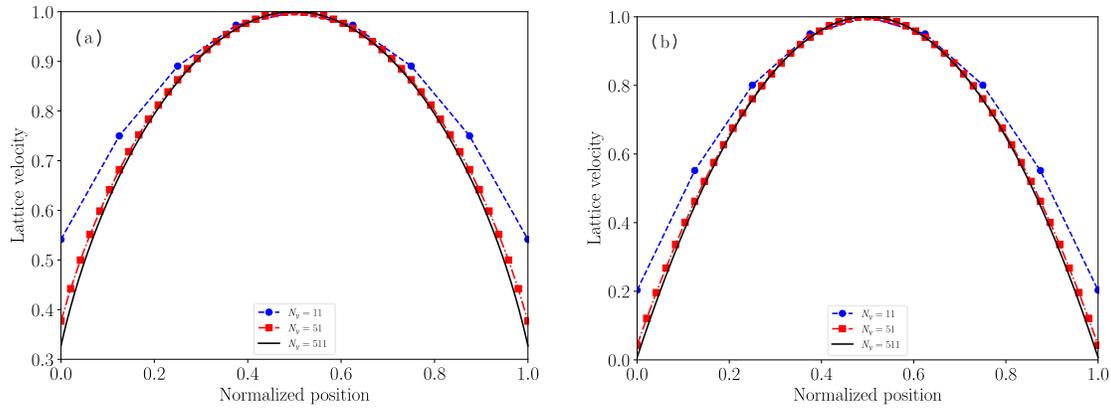


Fig. 4. Velocity profiles in different sized slit nano-channels: (a) 5 nm and (b) 1,000 nm.

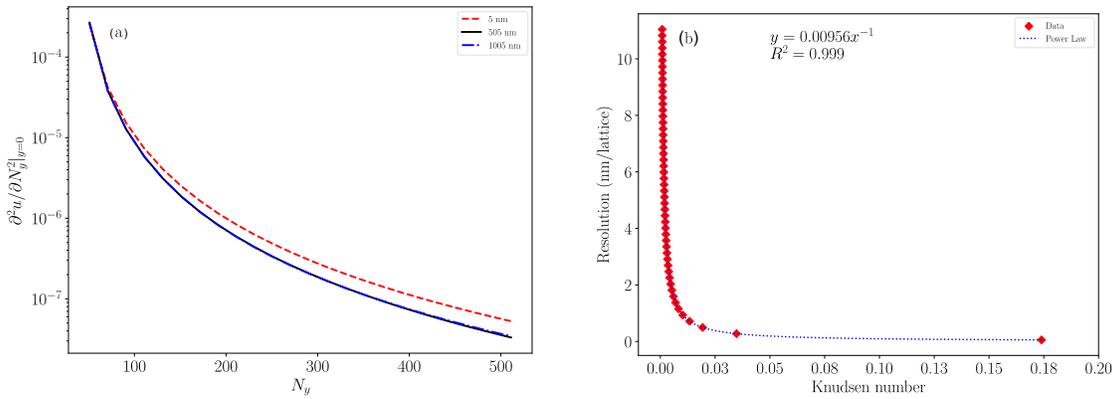


Fig. 5. Results of resolution analysis: (a) second-order derivative of velocity at the boundary and (b) correlation between critical resolution and Knudsen number.

3.2 Local Knudsen number

In simple channels, such as those used in our resolution analysis, determining the Knudsen number is a relatively straightforward task. However, in complex geometries, variations in pore diameter complicate the computation of the Knudsen number. Since our focus is primarily on complex porous media, it is important to define how the local Knudsen number is calculated. We begin with a binary uniform representation of the porous media, such as the one shown in Fig. 6, where pores are represented as white regions and grains as black. Using a morphological thinning operation in MATLAB, we generate a skeleton of the pore network. This process can also be performed using the OpenCV library in Python. Once the skeleton is obtained, we calculate the Euclidean distance from the skeleton to the pore walls (shown in the top-right image of Fig. 6). By multiplying the distance transform by the skeleton, we obtain the distance from the medial axis (i.e., the skeleton) to the pore wall, as shown in the bottom-right image of Fig. 6. Next, for each pore pixel, we identify the nearest point on the medial axis and define twice the distance from that point (on the medial axis) 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, as illustrated in the bottom-left image of Fig. 6.

3.3 Domain decomposition

The core of our domain decomposition algorithm is based on the quad-tree structure. A binary image representing the simulation domain is used as input, where cells with a value of 0 correspond to solid regions, and cells with a value of 1 represent fluid nodes. The ratio of solid nodes to all nodes in a quadrant is denoted by β . In its simplest form, the algorithm divides the image into four quadrants at each iteration, recalculating β until a desired condition is met, without considering the critical resolution. Once the algorithm converges, the resulting quadrants form the cells of the computational mesh. As depicted in Fig. 7, we add an additional condition for grid resolution dx . If the resolution of a given quadrant exceeds the critical value, the recursive function further divides it, ensuring that the required accuracy is maintained.

3.4 Relaxation parameter

As described in Eq. (6), the shear viscosity term τ_s is a function of the Knudsen number. In the multiblock LB method, this term also depends on the grid size. In the literature, the relationship between the shear viscosity on coarse and fine g-

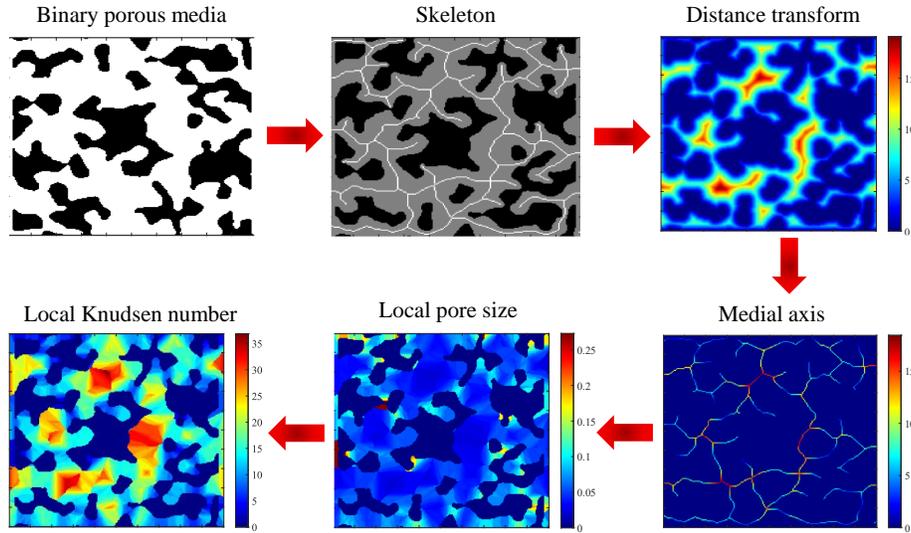


Fig. 6. Local Knudsen number calculation from binary image of porous media.

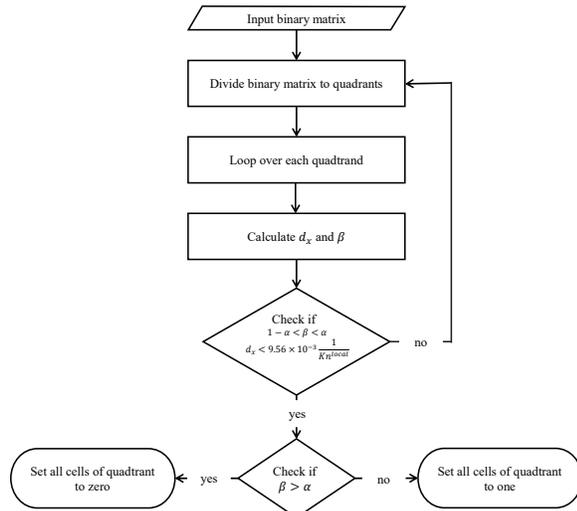


Fig. 7. Binary image decomposition flowchart.

rids, τ_s , is described as (Foroughi et al., 2013):

$$\tau_s^f = \left(\tau_s^c - \frac{1}{2} \right) \frac{\delta t^c}{\delta t^f} + \frac{1}{2} \quad (18)$$

where the superscripts f and c denote fine and coarse grid cells, respectively. This method works when the Knudsen component of the relaxation parameter is neglected. However, in confined media, rescaling relaxation parameters does not result in a smooth velocity profile near the walls. To address this issue, we combine Eqs. (6) and (18). Consider a simple 2D flow domain, as shown in Fig. 8, consisting of three levels of grid cells. To calculate the τ_s distribution in this domain, we create three copies of the same domain with uniform meshes at different levels. At the base lies the image with the coarsest mesh and its respective resolution. Using image skeletonization, we calculate the distance transform, which is then used to determine the local pore size and, ultimately, the local Knudsen number. Finally, using Eq. (6), we calculate t -

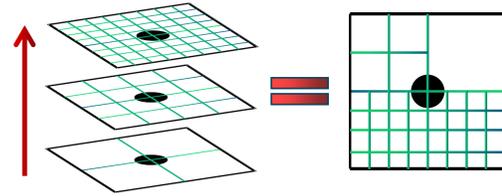


Fig. 8. Schematic of building the shear viscosity term.

he local τ_s distribution and Eq. (18) to rescale it at the correct level. Next, the image is resized to achieve a finer grid, and the process is repeated. In the end, all three scales are combined by selecting the corresponding values from each scale. This method of calculating τ_s results in a more robust and accurate multi-scale local shear stress distribution. The image skeletonization, resizing, and distance transform were performed using MATLAB and OpenCV's C++ API (Bradski and Kaehler, 2000).

3.5 Simulation algorithm

Due to the nature of the LB method, distribution functions cannot stream across the interface between cells of different sizes, because the cell size and time step are inherently linked. Specifically, the time step for coarse cells is n times the time step for fine cells, where n represents the ratio of the coarse to fine cell size. Consequently, fine cells must stream and collide n times for every single streaming and collision event in coarse cells. Furthermore, at the interface between coarse and fine cells, distribution functions cannot stream directly without interpolation. Inspired by the interface layout proposed by Liu et al. (2023), we developed a method that eliminates the need for time interpolation by introducing additional cells in the buffer layers, as shown in Fig. 9. The buffer zone consists of both fine and coarse grid cells at the interface. The main coarse cells (bottom) and their mirror image, referred to as coarse buffer or ghost cells (top), are highlighted in red in F-

Algorithm 1: Evolution step of multi-block LB method

```

1 Function evolutionStep( $L$ )
   Input   : Level  $L$ 
   Output : Updated Grid
2    $R \leftarrow 2^{M-L}$ ;
3   for  $i = 0$  to  $R - 1$  do
4     Call collide( $L$ );
5      $f_{\text{temp}} \leftarrow f$ ;
6     Call stream( $L$ );
7     Call boundaryConditions( $L$ );
8   end
9   if  $L > 1$  then
10    Recursive Call: evolutionStep( $L - 1$ );
11  end
12  if  $L \neq M$  then
13    Interpolation:
14    interpolateBlockInterface( $L, L + 1$ );
15  end

```

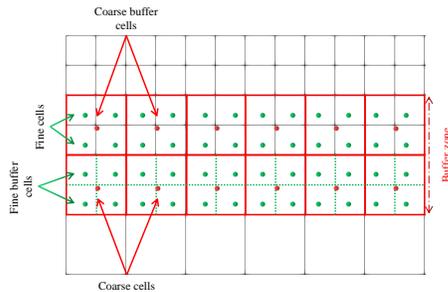


Fig. 9. Buffer layout at the interface of different grid levels.

Fig. 9. Similarly, the top two layers of green cells represent the main fine grid cells and their mirror image, the fine buffer cells, which are located below the main cells. In this setup, coarse ghost/buffer cells encapsulate the fine main cells, while fine ghost/buffer cells reside within the main coarse cells. This arrangement allows the streaming of coarse cell distribution functions into the fine region and vice versa. At the end of a time step, undetermined distribution functions exist within the buffer cells. For coarse buffer cells, these distribution functions can be determined by averaging the values of the constituent fine cells. However, fine buffer cells require a more rigorous approach, which is achieved using a 9-point bi-cubic interpolation method implemented with the AlgLib open-source C++ library (Bochkanov, 2024). Attempts were made to implement the interpolation equations from Liu et al. (2023), but their validity could not be verified. In practice, the velocity profiles could not be accurately recovered using those equations. It is important to note that the thickness of the buffer zone depends on the number of decomposition levels. For example, if there are only two levels of grid cells, the buffer zone will consist of 4 fine cell layers. For more than two levels, the general rule for determining the buffer zone thickness is $Z = 2^M$, where M is the number of levels. For a three-level grid, the buffer zone will be as thick as 8 fine-level cells. This is because, in a 3-level grid, a single streaming

and collision operation in the coarsest cells corresponds to 4 operations in the mid-level cells and 8 operations in the finest-level cells. To bring everything together, Algorithm 1 outlines a single iteration of the multi-block LB method. We define the coarsest cells as level $L = M$ and the finest cells as level $L = 0$. Initially, the *evolutionStep* function is called with $L = M$ to perform collision and streaming operations in the coarsest cells, where the number of collision and streaming steps in each level is defined by $R = 2^{M-L}$. The *evolutionStep* function is then recursively called for finer levels as long as $L > 1$. Interpolation begins at the interface between the finest cells ($L = 0$) and $L = 1$ cells, working its way up until all undetermined buffer zones are interpolated. This method is highly effective, regardless of the number of levels involved.

3.6 Post-processing

The final stage of the multi-block LB method involves interpolating velocity fields from multi-block simulations to refine the solutions and enhance simulation accuracy. Each coarse cell is subdivided into four child cells, and the same interpolation technique described in the previous section is applied to compute the velocity and density in each child cell. This interpolation uses all eight neighbors of the parent cell and is based on a 9-point bi-cubic method. This interpolation step is performed only at the end of the simulation, just before outputting the results, and it incurs minimal computational expense. Omitting this step results in a slight decrease in the accuracy of velocity fields but significantly reduces memory demands, which can be advantageous for applications such as solute transport and reactive transport simulations (Mostaghimi et al., 2016; Liu and Mostaghimi, 2017).

To highlight the significance of interpolation-based post-processing, we simulate flow in a 32×32 grid using both uniform and multi-block LB methods. As shown in Figs. 10(a) and 10(b), post-processing smooths the velocity distribution, resulting in a profile indistinguishable from that of the uniform grid LB method. However, in Fig. 10(c), the child cell velocities are set equal to the parent cell without interpolation, leading to a less accurate representation. Furthermore, Fig. 10(d) demonstrates that without post-processing, the velocity profile becomes stair stepped. For the remainder of this work, we use the post-processed results of the multi-block LB method.

4. Results

4.1 Grid refinement

We begin by comparing different levels of grid refinement within a 512×512 domain. This domain represents a two-dimensional cross-section of a narrowing tube with sinusoidal boundaries, as illustrated in Fig. 11(a). This domain was specifically designed to capture the sharp variation in Knudsen number while introducing additional geometric complexity through the sinusoidal boundaries. At the fluid entrance (left) and exit (right), a buffer zone with a thickness of 50 pixels is applied to maintain periodicity. By setting the resolution to $\delta x = 10^{-9}$ m the local pore size at each pixel can be calculated as described in Fig. 6. Once the local pore size is determined,

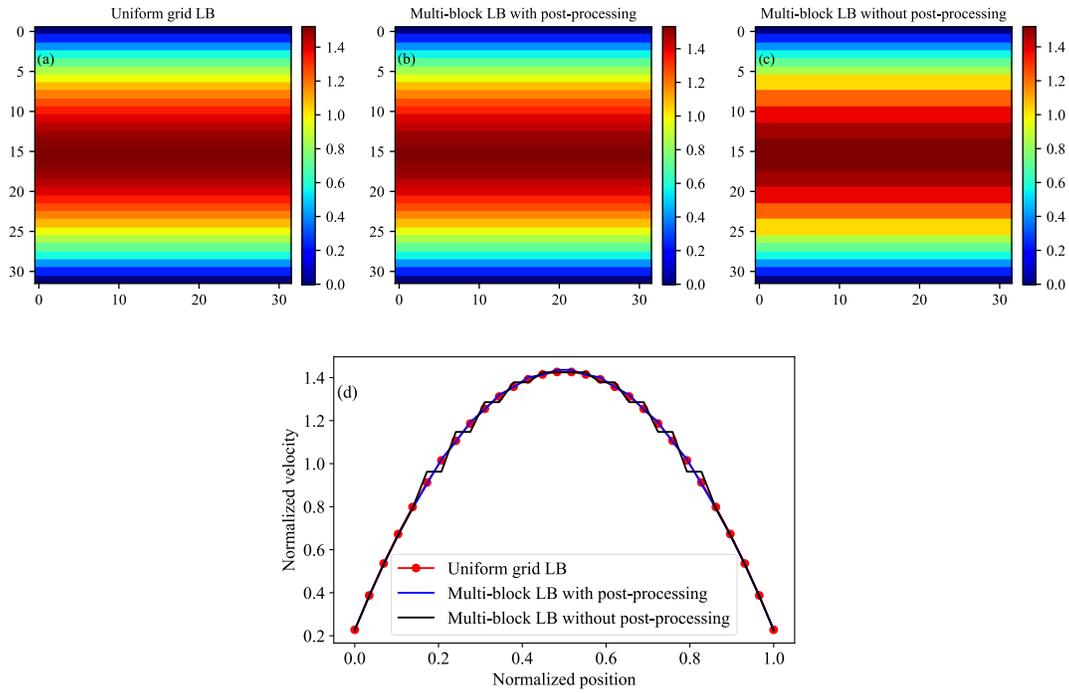


Fig. 10. Multi-block LB method with and without post-processing: normalized stream-wise velocity distribution of (a) uniform grid LB, (b) multi-block LB with post-processing, (c) multi-block LB without post-processing and (d) cross sectional velocity profiles.

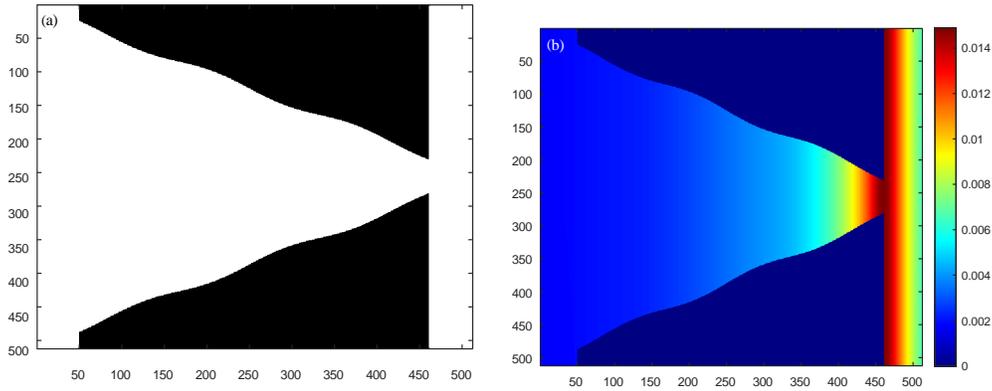


Fig. 11. Narrowing pore: (a) binary image, and (b) Knudsen number distribution.

the local Knudsen number can be easily computed, which, in this case, lies entirely within the slip flow regime (Fig. 11(b)). Using the domain decomposition flowchart described earlier (see Fig. 7), the image is refined at different levels based on the Knudsen number.

The finest cell is defined as level 0. As the cell size increases, the level increases correspondingly, with the maximum level configurable during the refinement process. In Fig. 12(a), the maximum level is set to 1, resulting in only two levels of grid cells. In Figs. 12(b) and 12(c), the maximum levels are set to 2 and 3, respectively. The results clearly demonstrate that cells in regions with high Knudsen numbers are maintained at finer levels to prioritize accuracy over computational performance, achieving the intended objective.

4.2 Validation

Now that we have established the mesh refinement and simulation algorithm, we must validate the accuracy of both the multi-block and single-block LB methods. First, we validate the methods against results from the Direct Simulation Monte Carlo (DSMC) method at different Knudsen numbers. We set up a square 64×64 domain with the top and bottom boundaries defined by a combination of halfway bounce-back and specular reflection, while the inlet and outlet are set to periodic conditions. At 2.0 MPa pressure and 300.0 K temperature, the resulting mean free path is calculated using the following equation:

$$\lambda = \frac{M_w}{\sqrt{2}\rho N_A \pi d^2} \quad (19)$$

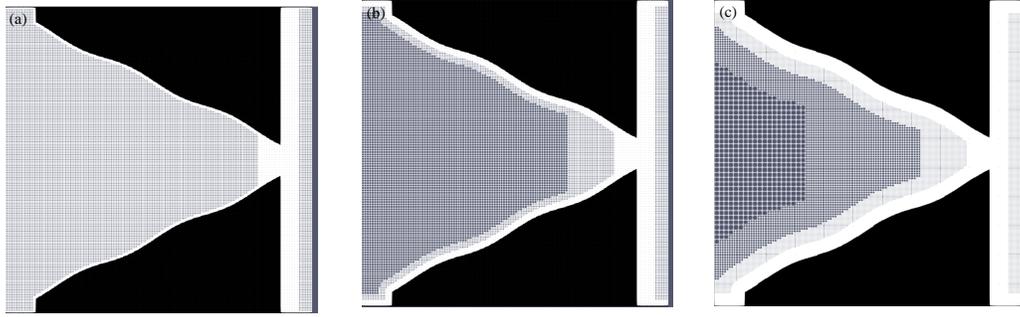


Fig. 12. Mesh refinement at different levels: (a) 2-level grid, (b) 3-level grid and (c) 4-level grid.

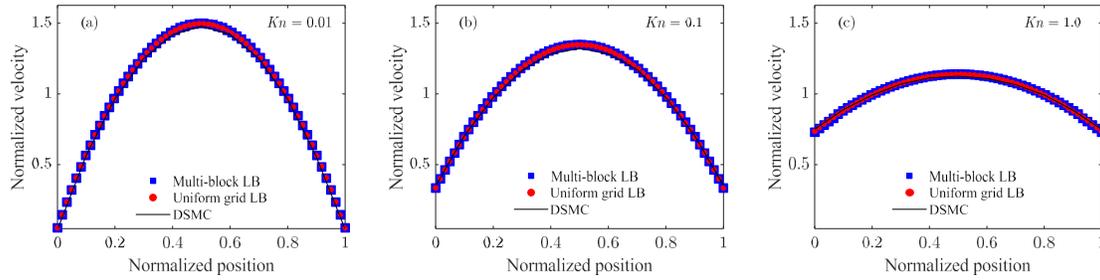


Fig. 13. Validation of stream-wise velocity profiles at different Knudsen numbers using single-block and multi-block LB methods: (a) $K_n = 0.01$, (b) $K_n = 0.1$ and (c) $K_n = 1$. DSMC results are obtained from Suga (2013).

where M_w is the molecular weight of methane, N_A is Avogadro's constant, d is the molecular diameter of methane, and ρ is the density, which is calculated using the Peng-Robinson equation of state. We set the Knudsen number to 0.01, 0.1, 1.0 by adjusting the pixel resolution in the 64×64 domain. We ran 3 simulations, each corresponding to one of the Knudsen numbers, using both the single-block and multi-block LB methods. The stream-wise velocity profiles from these simulations were then compared with DSMC data obtained from Suga (2013). As expected, the velocity profile flattens as the Knudsen number increases due to the pronounced slip effect at the boundaries (see Fig. 13). There is no noticeable difference between the results from the single-block and multi-block LB methods and the DSMC results with an average error of 1.6%.

Although the LB method, with appropriate boundary treatments and Knudsen corrections, is well-suited for simulating gas flow in the slip-flow regimes, it becomes less reliable as the Knudsen number increases. In the transitional flow regime ($K_n > 0.1$), the influence of the Knudsen layer become significant, making LBM less reliable for accurate simulations (Suga, 2013). In this regime, molecular dynamics (MD) is generally more suitable technique. The $K_n = 1.0$ case shown in Fig. 13 is intended merely as a demonstration of how LBM can approximate DSMC results and does not necessarily reflect the method's effectiveness at higher Knudsen numbers.

The second validation case involves the domain shown in Fig. 14(a). Molecular simulation data for this case was obtained from Suga (2013). The simulation was conducted in a 1024×1024 domain with $K_n = 0.15$, representing

the transitional flow regime. The presence of a triangular obstacle creates resistance, reducing the overall velocity. In Fig. 14(b), the normalized stream-wise velocity profiles from the uniform grid and multi-block LB simulations (with a maximum of 2 grid levels) are compared with MD results obtained from literature. Overall, there is a good match among the profiles, though slight deviations are observed with an average deviation of 3.3%. These deviations are likely due to restricting the Knudsen number to a constant value of 0.15 across the entire domain, whereas the Knudsen number should ideally vary depending on the pore tightness.

4.3 Computational performance

To assess the computational efficiency of the multi-block LB method compared to the uniform grid LB method, we set up several uniform computational domains. Each domain contains a circular obstacle at the center to simulate complex boundaries. The domain sizes vary from 64×64 to 1024×1024 , with the obstacle radius set to $1/8$ of the domain size. The uniform grid resolution is 1 nm. Each domain is refined using the method shown in Fig. 7 to generate a multi-block grid with varying grid resolutions. An example pair of 64×64 domains is shown in Fig. 15. We generate five pairs of domains by doubling the size up to 1024×1024 . Next, we perform simulations of methane transport with periodic inlet and outlet boundary conditions under a constant pressure of 2 MPa and a temperature of 300 K for both uniform and multi-block domains. In total, 10 test simulations are run: Five using uniform domains and five using multi-block domains. The simulations are executed on a 64-bit Intel Xeon W-2275 processor, initially

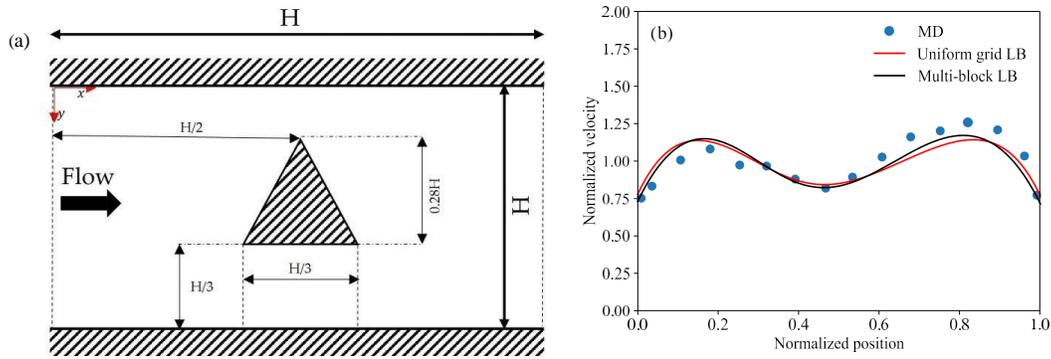


Fig. 14. Validation of stream-wise velocity profiles: (a) simulation domain setup and (b) normalized streamwise velocity profiles at the inlet. MD data was obtained from Suga (2013).

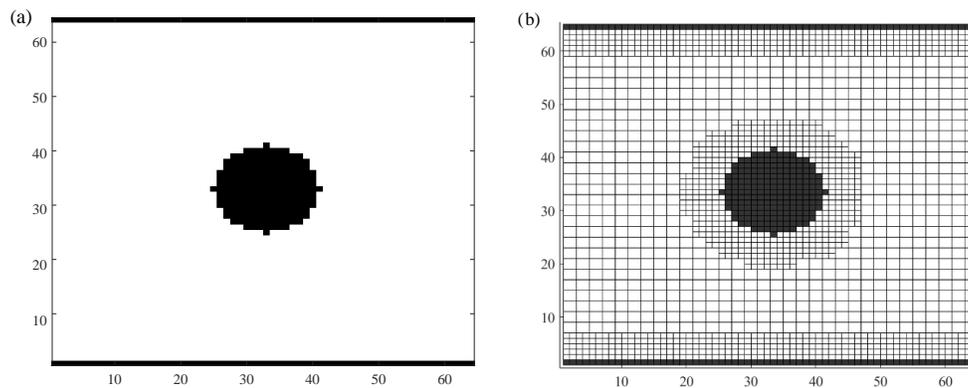


Fig. 15. Simulation domain for runtime tests: (a) uniform binary simulation domain with circular obstacle and (b) non-uniform domain after refinement.

using a single core for 100 timesteps, with the average CPU time recorded as the runtime per timestep. We then repeat the simulations using 4 and 8 CPU cores in parallel, calculating the average runtime per timestep for each configuration. The results, plotted in Figure 16, show the average runtime per timestep (y-axis) against the domain size in the flow direction (x-axis). For smaller domains, there is no discernible difference between the uniform grid LB and multi-block LB methods due to the overhead introduced by the interpolation operation in the multi-block LB method. However, as the domain size increases, the multi-block method significantly outperforms the uniform grid LB method. Additionally, Figs. 16(b) and 16(b) show that the multi-block method is as scalable as the uniform grid LB method. The key takeaway from the runtime study is that the multi-block method is most suitable for large domains, while the uniform grid LB method performs equally well in small domains. In the next section, we will demonstrate that the multi-block method generally converges to the same tolerance faster than the uniform LB method, which provides an additional computational advantage.

4.4 Flow in complex systems

In this section, we generate a complex porous medium using random Gaussian noise. The noisy image is filtered with

a 2D Laplacian of Gaussian filter and binarized within a 1024×1024 domain. The domain resolution is set to 1 nm per pixel, resulting in 1024 nm by 1024 nm. We then extract the skeleton and calculate the local pore size, which is used to determine the local Knudsen number. The binary image and its corresponding Knudsen number distribution are shown in Fig. 17.

We perform LB simulations of methane transport with and without grid refinement at 2 MPa pressure and 300 K temperature. The porosity of the binary domain is around 68.73%, resulting in 720,697 active cells. The 2-level grid refinement reduced the total number of active cells to 493,792, including the buffer cells, representing a 31.5% reduction from the uniform grid approach. Fig. 18 (top and middle rows) presents the distribution of methane flow velocity magnitude (m/s) for both the uniform grid and multi-block LB simulations. To highlight the primary flow pathways, we applied a threshold of 6.010^{-6} and filtered the velocity as shown in the right columns of the top two rows in Fig. 18. To compare the multi-block simulation to results with those from the uniform grid, we computed the absolute difference, which is shown in Fig. 18 (bottom left). Additionally, we calculated the errors between uniform grid and multi-block simulation results using Eq. (20):

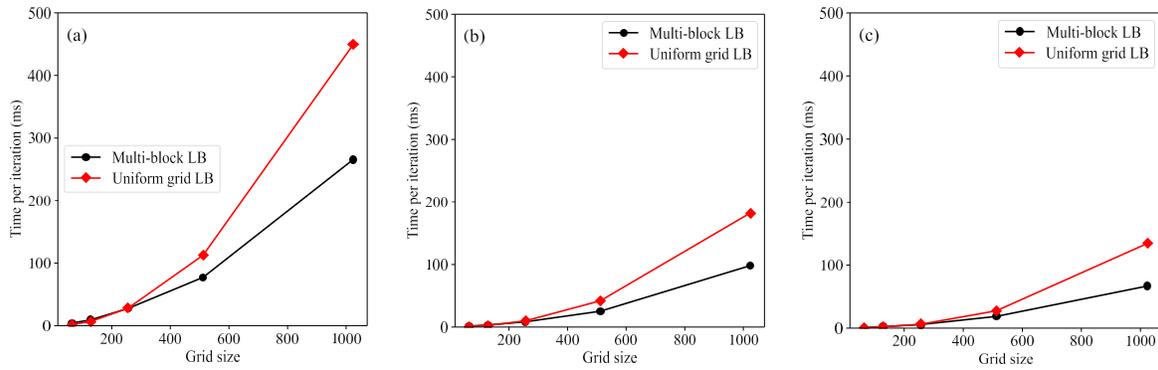


Fig. 16. Runtime of a single time-step for the proposed multi-block and uniform grid LB methods: (a) runtime on a single CPU, (b) 4 CPUs and (c) 8 CPUs.

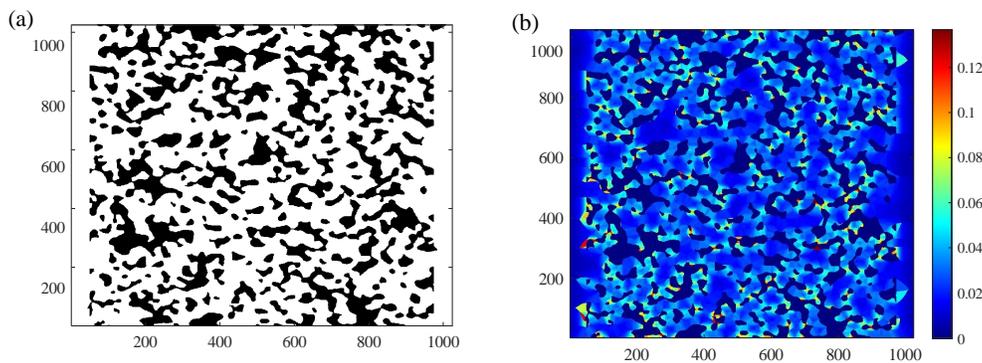


Fig. 17. Complex pore system: (a) binary media and (b) local Knudsen number distribution.

$$\text{Error} = \frac{|u_{ug} - u_{mb}|}{u_{ug,max} - u_{ug,min}} \times 100 \quad (20)$$

where u_{ug} and u_{mb} denote the velocity magnitudes from the uniform grid and multi-block LB simulations, respectively, while $u_{ug,max}$ and $u_{ug,min}$ represent the maximum and minimum velocity magnitudes in the uniform grid LB simulation. The error distribution is shown in Fig. 18 (bottom right), with a mean error of 0.07%.

As mentioned in Section 4.3, the multi-block LB method converges faster than uniform grid LBM. This is evident in Fig. 17(a), which shows that, for the same domain (Fig. 17(a)), the multi-block LB method reached a convergence tolerance of 10^{-6} in 8,850 steps, whereas the uniform grid LB method required 11,935 iterations to achieve the same tolerance. This shows that the multiblock method converged 25% faster. However, the extent of this improvement depends on the geometric complexity and the level of grid refinement applied. To further assess the capability of the multi-block method, we conducted a set of simulations in the same domain (Fig. 17(a)) under varying pressures from 2 to 6 MPa at a constant temperature of 300 K. We then calculated the permeability at each pressure and compared the results with those obtained using the uniform grid approach. Fig. 19(b) shows apparent permeability as a function of the inverse of mean pressure. The non-linear behavior of this curve is consistent with the

findings of Froué et al. (2020). The average relative error in permeability between the uniform grid and multi-block LB methods was approximately 0.47%, further confirming the accuracy and reliability of the multi-block method.

5. Conclusions

The proposed multi-block lattice Boltzmann method has been shown to be an accurate and computationally efficient approach for simulating gas transport in high Knudsen number regimes. The key contributions of our work are as follows:

- 1) A Knudsen number-driven grid refinement strategy is proposed to maintain accurate slip velocity in confined environments.
- 2) The shear stress component in the relaxation matrix is calculated by individually combining different grid levels and scaling them using the coarse and fine grid time steps, δt .
- 3) A modified evolution algorithm is introduced, which does not require time interpolation.
- 4) A new method is presented for classifying boundary types for the specular reflection boundary condition in irregular geometries.

Both multi-block and uniform grid approaches were validated using Direct Simulation DSMC and MD simulations from the literature 1.6% and 3.3% of average errors were

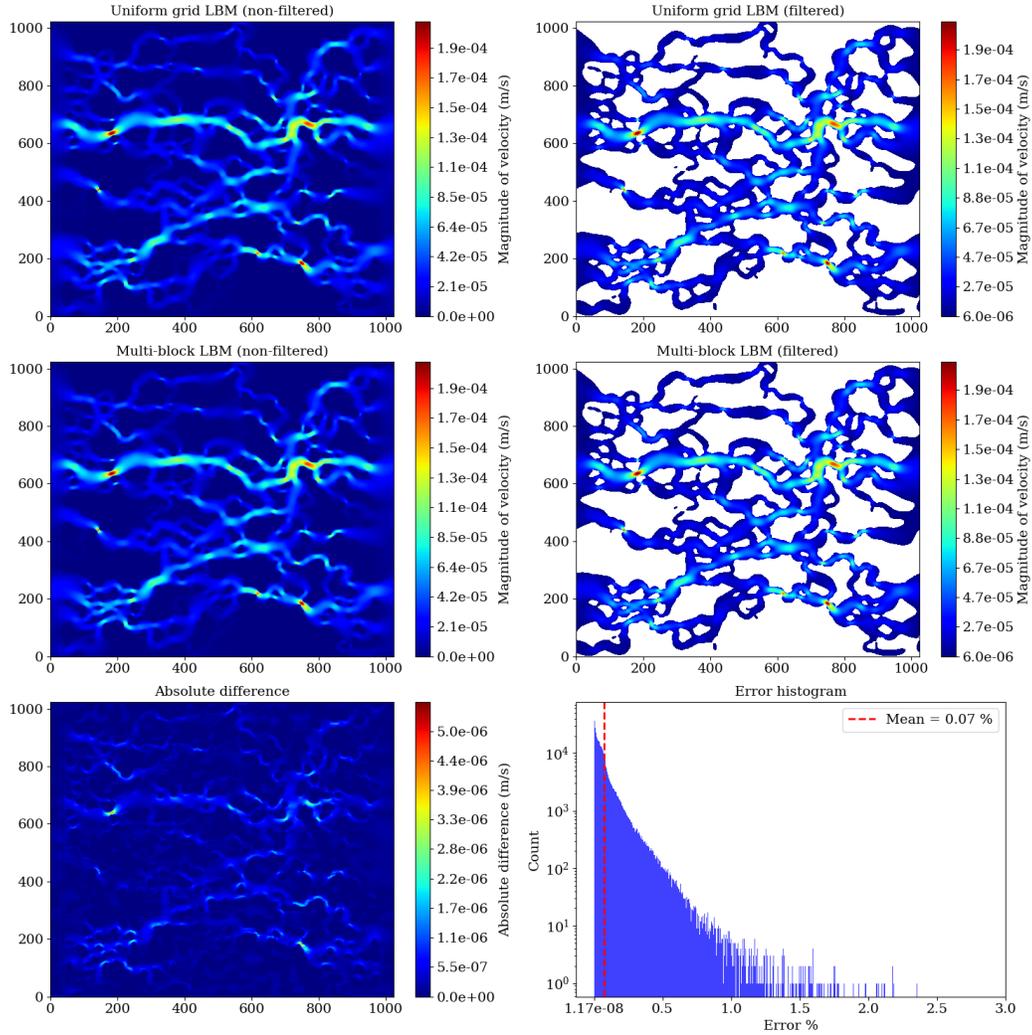


Fig. 18. Gas transport in complex porous media. Top row: simulation results of uniform grid LB, middle row: simulation results of multi-block LB, bottom row: absolute difference between uniform grid and multi-block LB methods (left) and error histogram (right).

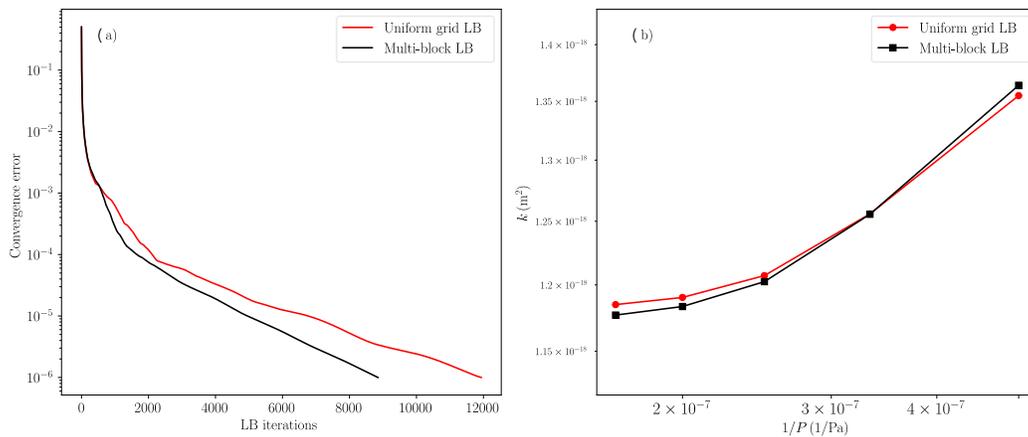


Fig. 19. Convergence profiles (a) and apparent gas permeability of the complex porous media versus inverse pressure (b).

reported in simple channel and flow around triangular obstacle, respectively. It was shown that not only is the multi-block LB method significantly faster per iteration, but it also converges in far fewer iterations, making it a more favorable approach. We also demonstrated simulation results for methane transport in complex porous media generated using random Gaussian noise. Additionally, we showed that the multi-block method accurately predicts the apparent gas permeability in complex porous media under varying pressures. That said, there are some limitations to our method that should be acknowledged. Firstly, although the LB method is well suited for rarefied systems, its reliability decreases at $Kn > 0.1$, and at $Kn > 1.0$, its suitability becomes questionable at best. Secondly, the proposed multi-block method is best suited for large systems. As demonstrated in the computational benchmark section, the runtime per iteration in small systems is not significantly different from the uniform grid approach. Furthermore, since our work focuses on high Knudsen number flows, the numerical scheme can become unstable if the Knudsen number requirement (see Fig. 7) is not met. Overall, our approach marks another step towards bridging the gap between system-level behavior of shale gas systems and confined pore-scale behavior. Future work will extend this approach to incorporate molecular interactions, capturing the coupled effects of adsorption and transport.

Acknowledgements

The corresponding author extends gratitude to the School of Energy Resources at the University of Wyoming for funding this work through its Mowry Shale initiatives.

Conflict of interest

The authors declare no competing interest.

Open Access This article is distributed under the terms and conditions of the Creative Commons Attribution (CC BY-NC-ND) license, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

References

- Abedi, B., Orujov, A., Dabbaghi, E., Ng, K., et al. Containment strategy for subsurface H₂ storage based on time-dependent soft solids. *International Journal of Hydrogen Energy*, 2024, 82: 1001–1014.
- Adams, L., Chartier, T. P. A comparison of algebraic multigrid and geometric immersed interface multigrid methods for interface problems. *SIAM Journal on Scientific Computing*, 2005, 26(3): 762-784.
- Ahmed, F., Gupta, A., Arora, N. An Eulerian based geometry conforming grid-block dynamic mesh refinement for the lattice Boltzmann method. *Physics of Fluids*, 2023, 35(10): 107142.
- Bochkanov, S. [ALGLIB-Numerical Analysis Library, 2024.](#)
- Bocanegra, J. A., Misale, M., Borelli, D. A systematic literature review on lattice Boltzmann method applied to acoustics. *Engineering Analysis with Boundary Elements*, 2023, 158: 405-429.
- Bradski, G., Kaehler, A. [OpenCV Library, 2000.](#)
- Carrillo, F. J., Bourg, I. C., Soulaine, C. Multiphase flow modeling in multiscale porous media: An open-source micro-continuum approach. *Journal of Computational Physics X*, 2020, 8: 100073.
- Chen, H., Chen, S., Matthaeus, W. H. Recovery of the Navier-Stokes equations using a lattice-gas Boltzmann method. *Physical Review A*, 1992, 45(8): R5339-R5342.
- Chen, H., Filippova, O., Hoch, J., et al. Grid refinement in lattice Boltzmann methods based on volumetric formulation. *Physica A: Statistical Mechanics and Its Applications*, 2005, 362(1): 158-167.
- Chen, Y., Kang, Q., Cai, Q., et al. Lattice Boltzmann method on quadtree grids. *Physical Review E*, 2011, 83(2): 026707.
- Cheng, M., Hung, K. C. Lattice Boltzmann method on nonuniform mesh. *International Journal of Computational Engineering Science*, 2004, 5(2): 291-302.
- Chiu, T. -H., Lin, C. -A. Multigrid accelerated projection method on GPU cluster for the simulation of turbulent flows. *Journal of Mechanics*, 2023, 39: 199-212.
- D’Humières, D. Multiple-relaxation-time lattice Boltzmann models in three dimensions. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 2002, 360(1792): 437-451.
- Dongari, N., Agrawal, A. Modeling of Navier-Stokes equations for high Knudsen number gas flows. *International Journal of Heat and Mass Transfer*, 2012, 55(15-16): 4352-4358.
- Foroughi, S., Jamshidi, S., Masihi, M. Lattice Boltzmann method on quadtree grids for simulating fluid flow through porous media: A new automatic algorithm. *Physica A: Statistical Mechanics and Its Applications*, 2013, 392(20): 4772-4786.
- Frouté, L., Wang, Y., McKinzie, J., et al. Transport simulations on scanning transmission electron microscope images of nanoporous shale. *Energies*, 2020, 13(24): 6665.
- Guo, Z., Shi, B., Zhao, T. S., et al. Discrete effects on boundary conditions for the lattice Boltzmann equation in simulating microscale gas flows. *Physical Review E*, 2007, 76(5): 056704.
- Guo, Z., Zheng, C., Shi, B. Lattice Boltzmann equation with multiple effective relaxation times for gaseous microscale flow. *Physical Review E*, 2008, 77(3): 036707.
- Guzik, S. M., Weisgraber, T. H., Colella, P., et al. Interpolation methods and the accuracy of lattice-Boltzmann mesh refinement. *Journal of Computational Physics*, 2013, 259: 461-487.
- He, X., Luo, L. -S. Theory of the lattice Boltzmann method: From the Boltzmann equation to the lattice Boltzmann equation. *Physical Review E*, 1997, 56(6): 6811-6817.
- Lallemand, P., Luo, L. -S. Theory of the lattice Boltzmann method: Dispersion, dissipation, isotropy, Galilean invariance, and stability. *Physical Review E*, 2000, 61(6): 6546-6562.
- Latt, J., Malaspinas, O., Kontaxakis, D., et al. Palabos: Parallel lattice boltzmann solver. *Computers & Mathematics with Applications*, 2020, 81: 334-350.
- Li, Q., He, Y. L., Tang, G. H., et al. Lattice Boltzmann model-

- ing of microchannel flows in the transition flow regime. *Microfluidics and Nanofluidics*, 2010, 10(3): 607-618.
- Li, W., Wang, D., Wang, J. G. Improved mathematical model of apparent permeability: A focused study on free and multilayer adsorptive phase flow. *Journal of Natural Gas Science and Engineering*, 2022, 101: 104508.
- Liu, L., Frouté, L., Kovscek, A. R., et al. Scale translation yields insights into gas adsorption under nanoconfinement. *Physics of Fluids*, 2024, 36(7): 072011.
- Liu, L., Wang, Y., Aryana, S. A. Insights into scale translation of methane transport in nanopores. *Journal of Natural Gas Science and Engineering*, 2021, 96: 104220.
- Liu, M., Mostaghimi, P. High-resolution pore-scale simulation of dissolution in porous media. *Chemical Engineering Science*, 2017, 161: 360-369.
- Liu, Z., Tian, F. -B., Feng, X. An efficient geometry-adaptive mesh refinement framework and its application in the immersed boundary lattice Boltzmann method. *Computer Methods in Applied Mechanics and Engineering*, 2022, 392: 114662.
- Liu, Z., Li, S., Ruan, J., et al. A New Multi-Level grid multiple-relaxation-time lattice Boltzmann method with spatial interpolation. *Mathematics*, 2023, 11(5): 1089.
- Loucks, R. G., Reed, R. M., Ruppel, S. C., et al. Spectrum of pore types and networks in mudrocks and a descriptive classification for matrix-related mudrock pores. *AAPG Bulletin*, 2012, 96(6): 1071-1098.
- Mehmani, Y., Anderson, T., Wang, Y., et al. Striving to translate shale physics across ten orders of magnitude: What have we learned? *Earth-Science Reviews*, 2021, 223: 103848.
- Michalis, V. K., Kalarakis, A. N., Skouras, E. D., et al. Rarefaction effects on gas viscosity in the Knudsen transition regime. *Microfluidics and Nanofluidics*, 2010, 9(4-5): 847-853.
- Mohamad, A. A. *Lattice Boltzmann Method: Fundamentals and Engineering Applications with Computer Codes*. Berlin, Germany, Springer, 2019.
- Mostaghimi, P., Liu, M., Arns, C. H. Numerical simulation of reactive transport on micro-CT images. *Mathematical Geosciences*, 2016, 48: 963-983.
- Nannelli, F., Succi, S. The lattice Boltzmann equation on irregular lattices. *Journal of Statistical Physics*, 1992, 68(3-4): 401-407.
- Perumal, L., Koh, W. H. Techniques for element formulation and quadtree-based triangular mesh generation for strain-based finite elements. *MethodsX*, 2023, 10: 102027.
- Prouvost, L., Belme, A., Fuster, D. A metric-based adaptive mesh refinement criterion under constraints for solving elliptic problems on quad/octree grids. *Journal of Computational Physics*, 2024, 506: 112941.
- Rapp, B. E. *Microfluidics: Modeling, Mechanics and Mathematics*. Amsterdam, Netherlands, Elsevier, 2016.
- Rustamov, N., Aryana, S. GPU-accelerated simulations of gas transport and adsorption in complex shale systems. *ESS Open Archive*, 2024.
- Rustamov, N., Liu, L., Aryana, S. A. Scalable simulation of coupled adsorption and transport of methane in confined complex porous media with density preconditioning. *Gas Science and Engineering*, 2023, 119: 205131.
- Shephard, M. S., Georges, M. K. Automatic three-dimensional mesh generation by the finite octree technique. *International Journal for Numerical Methods in Engineering*, 1991, 32(4): 709-749.
- Su, W., Lindsay, S., Liu, H., et al. Comparative study of the discrete velocity and lattice Boltzmann methods for rarefied gas flows through irregular channels. *Physical Review E*, 2017, 96(2): 023309.
- Succi, S. *The Lattice Boltzmann Equation: For Fluid Dynamics and Beyond*. Oxford, UK, Oxford University Press, 2001.
- Succi, S. Mesoscopic modeling of slip motion at fluid-solid interfaces with heterogeneous catalysis. *Physical Review Letters*, 2002, 89(6): 064502.
- Suga, K. Lattice Boltzmann methods for complex micro-flows: Applicability and limitations for practical applications. *Fluid Dynamics Research*, 2013, 45(3): 034501.
- Wang, Y., Aryana, S. A. Pore-scale simulation of gas flow in microscopic permeable media with complex geometries. *Journal of Natural Gas Science and Engineering*, 2020, 81: 103441.
- Wu, Z., Sun, Z., Shu, K., et al. Mechanism of shale oil displacement by CO₂ in nanopores: A molecular dynamics simulation study. *Advances in Geo-Energy Research*, 2024, 11(2): 141-151.
- Zhang, L., Zhao, Z., Chang, X., et al. A 3D hybrid grid generation technique and a multigrid/parallel algorithm based on anisotropic agglomeration approach. *Chinese Journal of Aeronautics*, 2013, 26(1): 47-62.
- Zhang, R., Shan, X., Chen, H. Efficient kinetic method for fluid simulation beyond the Navier-Stokes equation. *Physical Review E*, 2006, 74(4): 046703.
- Zhao, Y., Luo, M., Liu, L., et al. Molecular dynamics simulations of shale gas transport in rough nanopores. *Journal of Petroleum Science and Engineering*, 2022, 217: 110884.