Imperial College London
Department of Chemical Engineering

Multi-scale simulation of multiphase multi-component flow in porous media using the Lattice Boltzmann Method

Jianhui Yang

Supervised by Dr. Edo Boek

Submitted in part fulfilment of the requirements for the degree of
Doctor of Philosophy in Chemical Engineering of Imperial College London
and the Diploma of Imperial College London

Declaration

I hereby declare that this thesis titled Multi-scale simulation of multiphase multi-component flow in porous media using the Lattice Boltzmann Method is entirely my own work under the supervision of Dr. Edo Boek otherwise it is appropriately acknowledged. This work has not been previously submitted in its entirety or in part to any other academic institute for a degree.

Jianhui Yang
Department of Chemical Engineering
Imperial College London

Abstract

This thesis consists of work mainly performed within the Qatar Carbonates and Carbon Storage Research Centre (QCCSRC) project, focusing on the prediction of flow and transport properties in porous media. The direct pore scale simulation of complex fluid flow on reservoir rocks is the main topic of this work. A simulation package based on the lattice Boltzmann method has been developed to study single and multiphase flow as well as thermal and solute dispersion in porous media. The simulator has been extensively validated by comparing simulation results to reference solutions. Various numerical experiments have been performed to study the single/multiphase/solute dispersion flow in reservoir rocks. The simulator successfully predicts various transport properties including single phase and relative permeability, capillary pressure, initial-residual saturation, residual cluster size distribution and dispersion coefficient. The prediction has been compared to available experimental data and was generally found to be in good agreement. The simulator is also ready for exploring the two-phase dynamic problem with coupled and nonlinear physical processes including the effect of wettability, surface tension and hysteresis.

To improve the efficiency of the lattice Boltzmann simulations, an optimised collision model and corresponding parallel operation are proposed and implemented. A sparse storage scheme which significantly reduces the memory requirement has been designed and implemented for complex porous media. Due to the application of these optimisation schemes, it is possible to perform simulations on large scale samples (Size >1024x512x512). The optimised code shows very good and promising performance, and nearly ideal scalability was achieved.

The publications as a result of this research are:

Yang, J. and Boek E. S. A comparison study of multi-component Lattice Boltzmann models for flow in porous media applications, *Computers & Mathematics with Applications* 65.6 (2013): 882-890.

Yang, J. and Boek, E. S. Quantitative determination of molecular propagator distributions for solute transport in homogeneous and heterogeneous porous media using lattice Boltzmann simulations, Submitted to *Water Resources Research* (Accepted).

Chapman, E. M., Yang, J., Crawshaw, J. P., and Boek, E. S. Pore Scale Models for Imbibition of CO_2 Analogue Fluids in Etched Micro-model junctions Using Micro-fluidic Experiments and Direct Flow Calculations, *Energy Procedia*, 37 (2013): 3680-3686.

Shah, S. M., Yang, J., Crawshaw, J., Gharbi, O., Boek, E. S. Predicting porosity and permeability of carbonate rocks from core-scale to pore-scale using Medical CT, Confocal Laser Scanning Microscopy and Micro CT, Proceedings of SPE Annual Technical Conference and Exhibition (2013), SPE 166252, New Orleans, Louisiana, USA.

Al-Ansi, N., Gharbi, O., Raeini, A. Q., Yang, J. Iglauer, S. Influence of Micro-Computed Tomography Image Resolution on the Predictions of Petrophysical Properties. Proceedings of International Petroleum Technology Conference (2013), IPTC 16600, Beijing, China.

Yang, J., Boek, E. S., Lattice-Boltzmann simulation of single phase flow in reservoir rocks, to be submitted.

Yang, J., Boek, E. S., Lattice-Boltzmann simulation of immiscible fluid flow in reservoir rocks and relative permeability calculation, to be submitted.

Yang, J., Boek, E. S., Calculation of cluster size distributions for residual oil in sandstones using lattice Boltzmann method, to be submitted.

Acknowledgement

I wish to express my deepest gratitude to my supervisor Dr.Edo Boek for his professional guidance, encouragement and providing academic freedom during my study. Without his outstanding supervision, this work would not have been possible.

I would like to thank Dr.John Crawshaw for his constructive and practical comments which were crucial to my research. I also thank Nayef Al-Ansi, Oussama Gharbi and Ali Qaseminejad Raeini for offering me the opportunity to collaborate with them. A special thanks to Emily Chapman and Shah Saurabh, without their excellent support of experiment work, my research could not have been done.

I also would like to thank my office mate Farrel Gray, we had countless valuable discussions together. I thank him also for having lunch together every day and for his patience with my slow eating. I would like to thank my wonderful fellow colleagues Emily Chapman, Shah Saurabh, Farrel Gray, Daniel Ross, Christine Seifriedfor and Rudolf Umla for their kindly help, encouragement, valuable comments and I feel honored to work with them.

Furthermore, I would like to thank Professor Geoff Maitland and Professor Omar Matar for serving on my transfer examination committee, and providing constructive comments.

My gratitude is also expressed to every member in the Qatar Carbonates and Carbon Storage Research Centre (QCCSRC) for their support and wonderful environment for my research and social activity.

Last but not least, I deeply thank my parents and my girlfriend for their unconditional support and love, especially my girlfriend, for her understanding of my absence and her endless patience. Without their love and encouragement, it would not be possible for me to finish the study.

I gratefully acknowledge funding from the Qatar Carbonates and Carbon Storage Research Centre (QCCSRC), provided jointly by Qatar Petroleum, Shell, and Qatar Science & Technology Park.

Contents

[1] This chapter has been published in Computers & Mathematics with Applications 65.6 (2013): 882-890.

[2] This chapter has been submitted to Water Resource Research for possible publication

List of Tables

List of Figures

1. Introduction

The rapidly increasing demand for fossil fuel pushes the petroleum industry to get a deeper understanding of earth systems. To better understand the earth system, we are interested in determining and predicting the transport properties within the basic elements of earth systems, the rocks. In order to determine, or predict the transport properties within the rocks, the underlying microscopic physics should be properly understood. Therefore we are keen to understand, simulate and predict pore-scale physics in porous media. However, this is normally difficult. Firstly, the geometry of the rocks can be highly heterogeneous and complicated. What's more, the transport phenomena inside are normally nonlinear and coupled. As an example, we consider the flow of water and oil in the rocks. The transport properties of some porous media have been determined by experiments in laboratories. However, they are time-consuming and very costly. For example, the experimental measurement of relative permeabilities takes a long time (several months for a single rock core) and is therefore very expensive (costs are of the order of several 10^5 USD) in commercial petrophysics labs. The quality of measured data depends on the experience of operators. Some empirical relations were proposed to estimate the transport properties within the rocks [8, 9]. Due to the complexity of the structure, these empirical relations are not always effective, especially for multi-phase and multi-component flows.

With the continuing development of the computer, numerical simulation is rapidly becoming an alternative solution for this problem. Several commercial enterprises already offer "digital rock physics" services as a complement or even alternative to experimental permeability measurements [10]. However, this problem is far from solved: the complex geometry of some classes of porous media, such as carbonate rocks, makes it difficult to model and predict the transport properties. Several numerical methods have been proposed to determine the transport properties of porous media,

including Finite Element Method (FEM)[11, 12], Finite Differential Method (FDM)[13, 14] and Network Modelling [15, 16]. Network Modelling is an attractive option as it uses a simplification of the actual pore space geometry and therefore provides relatively simple numerical solutions [17, 18]. Network models work well for relatively simple rocks such as sandstone. However, for more complex porous media, such as carbonates, which have a very broad pore size distribution, the generation of network models is difficult [19]. For this reason, calculation of the transport properties directly on pore space images obtained from X-ray microtomography (XMT) has recently seen a tremendous development. The direct methods mentioned above, FEM and FDM, generally require extensive meshing to obtain reasonable accuracy. They often are difficult to implement and are of low efficiency. Moreover, multi-physics processes including multi-phase and multi-component flow in complex geometries are very challenging for these traditional CFD methods.

As a powerful alternative, the lattice Boltzmann method (LBM) has been introduced as a novel CFD technique which is able to handle extremely complex geometries without simplification [20]. The Navier-Stokes equations with complicated boundaries can be solved accurately by the LB method. These advantages match our requirement for the determination of transport properties in the pore space of the rocks. The LB method is easy to implement compared with other CFD algorithms. Most of the computation is local and thus it is ideal for parallel implementation. Due to the statistical physics background, it is easy to simulate multi-physics processes including multi-phase and multi-component flow, evaporation, condensation as well as cavitation. These advantages make the LB method an ideal numerical tool to study flow properties in porous media. In this PhD study, I developed a lattice Boltzmann method simulator for the transport phenomena associated with the flow of reservoir fluids, comprising concentrated brines, CO_2 and hydrocarbons at the pore scale. It is able to determine the transport properties in realistic rock geometries. This simulator is reliable, efficient and accurate compared with other existing numerical methods, and more importantly, has the capability of handling complex geometries. We will use upscaling algorithms to predict flow properties at the core scale. We have produced a working LB simulation code to help design CO_2 re-injection

operations in Qatari oil and gas fields. The code package is able to carry out direct simulation on images from Micro-CT scanning for:

- Single phase flow in porous media
  - Simulate single-phase flow within complex geometries.
  - Calculate the absolute permeability of porous media.
  - Visualise the pressure and velocity field.
- Multiphase/Multi-component flow in porous media
  - Simulate multiphase/multi-component flow with viscosity and density contrast, surface tension and wettability in porous media.
  - Calculate the relative permeability of porous media with binary immiscible fluids.
  - Compute the capillary pressure of binary immiscible fluids in porous media.
  - Simulate drainage/imbibition cycles.
  - Predict the initial-residual saturation and cluster size distributions at different reservoir conditions.
  - Visualise pressure, velocity and wetting/non-wetting distributions.
- Thermal/solute dispersion in porous media
  - Simulate coupled flow and thermal convection/diffusion in porous media.
  - Calculate dispersion coefficient of the porous media.

To obtain high accuracy and efficiency for the simulations of transport phenomena associated with the flow of reservoir fluids, we proposed several optimisation schemes to overcome several difficulties of the lattice Boltzmann method:

- Improve numerical stability of the lattice Boltzmann method for low viscosity fluid flow simulations.
- Eliminate viscosity dependence of absolute permeability calculations.

- Eliminate high spurious velocities in multi-phase and multi-component lattice Boltzmann method simulations.
- Reduce significantly the computing time in 3D lattice Boltzmann simulations.
- Reduce constraints on the viscosity ratio and density ratio of immiscible binary fluid systems that can be simulated by the lattice Boltzmann method.

The basic fluid mechanics concepts and derivation of core equations of the lattice Boltzmann are given in Chapter 3. Various different multiphase/multicomponent lattice Boltzmann models and thermal/solute dispersion lattice Boltzmann models are discussed in Chapter 4-7. A special optimisation scheme for pore scale LB simulation is given in Chapter 9. The code performance is also studied in Chapter 9.

2. Background and Literature Review

A porous medium consists of pore space and a solid matrix. The pores are generally interconnected, which allows the fluids to flow and mass transfer to take place. The transport properties within a porous medium can be determined accurately if the physical equations can be solved. However, since the geometry is often extremely complicated and the problems in the porous medium are nonlinear and coupled [2], it is not practical to achieve an analytical solution. The porous medium is normally simplified as a homogeneous medium, and the parameters of the transport properties are measured by experiments in laboratories at the macroscopic scale. The experiments are very time-consuming and costly. As computer power develops, several numerical methods have been proposed as an alternative solution for the estimation of transport properties within a porous medium.

Network modelling was first proposed by Fatt [15] to investigate the capillary pressure. This method has been widely used to estimate the flow and transport in porous medium in the past 20 years. It simplifies the pore space into nodes that represent pore junctions and bonds that connect pore nodes. This simplification keeps a certain amount of the geometry information at the pore scale, including the connectivity and pore size distributions. At the same time, it reduces the complexity of the geometry. A limitation of the network modelling method is that the algorithm that separates the pore space into pore junctions and bonds is not unique. The distinction between the pores and the throats is not always clear in reality. This makes the procedure of dividing somewhat arbitrary. The simplified network model cannot always reflect the real geometry, and therefore this affects the accuracy of the estimation.

More precise network models have been developed in the past 15 years to improve the performance. Maximal ball algorithms were proposed to construct the largest spheres centred on each void voxel that just fits in the pore space [21]. A network model extracting process using the Maximal ball algorithm was developed based on a 3D representation of the pore space, obtained by micro-CT scanning tools. Flow and transport can be computed within this idealised geometry [16].

Several conventional CFD techniques have also been used to study the transport properties of porous medium. Adler *et al.*[14] used an alternating-direction-implicit (ADI) finite differential method to study Newtonian flow in 3D Fontainebleau sandstones with different porosities. Comparison between simulated permeability and experimental data was carried out; although the general shape of the experimental permeability[1] versus porosity[2] curve was predicted in quite an accurate way, the calculated permeability differs by, at most, a factor of 5 from the measured one. A regular mesh can be used in the finite difference method, but accurate computation with complicated boundaries requires an intensive mesh. Saeger *et al.*[12] used the finite element method (FEM) to solve the Stokes equations in periodic porous medium. The calculated permeability achieved reasonable agreement with computed permeability given by other authors where the relative difference was from 0.31% to 15% [22]. However, a very complicated irregular mesh needs to be used in finite element simulations for porous medium which limits the use of this method.

The lattice Boltzmann method [23, 24] uses a different approach to simulate the fluid flow. The LB method describes the fluid system by interactions of fictitious particle groups which reside on the lattice nodes. These particle groups are much bigger than the real fluid molecules, but show the same behaviour in density and velocity as the real fluid at the macroscopic scale. A number of studies showed that the LB method can recover the Navier Stokes equations at the macroscopic scale [25, 13, 26]. The LB method uses a unique way to simulate the flow. While the conventional CFD method

[1]Permeability is a measure of the ability of a porous media to allow fluids to pass through it.

[2]Porosity is a measure of the void spaces in a material.

discretizes the governing equation in a top-down approach, the lattice Boltzmann method recovers the governing equations from the rules for discretized models in a bottom-up approach.

Since all the operations are taking place on the lattice nodes in the LB method, the lattice model is essential. Several lattice models were proposed for the LB method [26, 27, 28]. d'Humieres *et al.* [27] and Chen *et al.* [26] showed that in order to have isotropic relaxation of the stress tensor, the grid should be formed in 4D and then be projected into 2D and 3D. The grid in 2D and 3D used in this work was proposed by Qian *et al.*[29]. The 3D lattice model contains 19 velocities with the name D3Q19; the 2D lattice model contains 9 velocities with the name D2Q9. They are the most used lattice models in lattice Boltzmann simulations. Two limits of the lattice Boltzmann method were proposed by Qian *et al.*[30] and Ladd [13]. They showed that the lattice Boltzmann method can recover the Navier-Stokes equations for low Mach number Ma and low Knudsen number Kn. This means the grid spacing has to be much smaller than the characteristic length ($Kn << 1$) and the fluid velocity should be much lower than the speed of sound ($Ma << 1$).

In its simplest incarnation, the LB method uses particle distribution functions which relax back to equilibrium using a single relaxation time (SRT) model. This is known as the BGK approximation and will be detailed in Chapter 3. To overcome the disadvantages of the BGK model, such as numerical instabilities, multiple-relaxation-time (MRT) lattice Boltzmann was developed by d'Humieres [31]. The main idea of MRT is using different relaxation time parameters for different moments of macroscopic quantities. The collision step is carried out in the momentum space which involves density, energy, momentum energy flux, diagonal and off diagonal components of the stress tensor. The streaming step, in which the particles propagate to the neighbouring nodes, is still done in velocity space as in the common LB method. The MRT model can overcome several common defects of the LB method such as fixed Prandtl number [3] (Pr=1 for common LBM) and fixed

[3]The Prandtl number Pr is a dimensionless number which describes the ratio of viscous diffusivity to thermal diffusivity. It is defined as $Pr = \frac{C_p \mu}{k}$, where C_p is the specific heat, μ is the dynamic viscosity, k is the thermal conductivity

ratio between the kinematic and bulk viscosities. The stability improvement by using the MRT scheme would reduce the computational effort by at least one order of magnitude while maintaining the accuracy of the simulations [32]. The numerical stability of MRT has been studied by (Lallemand and Luo 2000)[33] in detail. Their analysis showed that the MRT scheme led to a large improvement of numerical stability compared with the single relaxation time LB model.

There are three multi-component LB models for multi-phase and multi-component fluid simulations. The first multi-component LB model was proposed by Gunstensen *et al.*[34] with the name colour gradient model. Two components represent two types of fluid with their own distribution functions, and follow their own evolution equation. They are named red particles and blue particles. The collision step includes self-interactions and cross-interactions with other types of particles. A colour function gradient was introduced to calculate the surface tension between different phases [35]. To segregate the phases, mixing near the interface should be minimised. A procedure called recolouring is proposed for this minimising process [36, 37]. This procedure is a very time-consuming step, and this model also has some numerical stability problems for high density ratio and large surface tension. We found that this model is capable of simulating immiscible binary fluids with viscosity contrast but the same density (see Chapter 5).

The free energy model was developed by Swift *et al.*[38]. This model includes thermodynamic equilibrium functions of phases, and a term describing the surface tension is added to the equilibrium distribution function. This allows the free energy model to specify the surface tension more easily than other multiphase multi-component models. It is also a fully thermodynamically consistent binary fluid lattice Boltzmann model. To reduce spurious currents, Pooley *et al.* [39] proposed a modified distribution function for the free energy model, decreasing spurious velocities by an order of magnitude compared to previous models. We found that the free energy model is able to simulate binary fluid systems with viscosity contrast but the same density (see Chapter 5).

A pseudo potential lattice Boltzmann model was developed by Shan and

Chen [40]. The principal characteristic of this model is an interaction force between particles that is introduced to have a consistent treatment of the equation of state for a non-ideal gas. Shan [1] reported that spurious velocities are due to the lack of sufficient isotropy in the calculation of the gradient term for the interaction force. Finite difference gradient operators with higher order of isotropy were proposed and spurious currents were found to decrease significantly [1, 41, 42]. We found that the Shan-Chen model is able to simulate a binary fluid system with density contrast, but the viscosity of fluids must be the same (see Chapter 5).

A variety of studies on the transport properties of porous medium using the LB method has been carried out. The absolute permeability of a reconstructed geometry from Fontainebleau sandstone was calculated using the lattice Boltzmann method by Jin [43]. The porosity of the sandstone sample was 39.8%. Good agreement with the empirical formula was obtained. However, the permeability calculation was observed to be dependent on the viscosity. The viscosity dependence poses a severe problem for computing the permeability. Because the permeability is a characteristic of the physical properties of the porous medium, it should only be related to the geometry of porous medium. This viscosity dependence can be reduced by using the MRT scheme [44]. Pan *et al.*[45] used a Shan-Chen multi-component lattice Boltzmann model to simulate multi-component flow in an idealised porous medium which comprises spheres with identical radius. The capillary pressure head was measured as a function of saturation. Acceptable agreement with experimental data was obtained. The densities of the binary fluids were the same, and a small viscosity ratio of 1.8 was used in the simulation. This value of the viscosity ratio is generally too low for our systems containing oil, water and supercritical CO_2 (the viscosity ratio of water and supercritical CO_2 is around 20). The efficiency of their code seems to be rather low, as the biggest geometry computed is a domain of 128^3 voxels. Ramstad *et al.*[46] used a color gradient lattice Boltzmann model to simulate an immiscible binary fluid system with equal viscosity in a Bentheimer sandstone where the geometry was taken from X-ray microtomography. Good agreement between the experimental data and simulated relative permeability of the wetting phase was obtained. The simulation under-predicts the relative permeability of the non-wetting phase at high

wetting phase saturation. The viscosity and the density were kept the same in this study.

Since 3D simulations normally require a large amount of computation, the efficiency of the algorithm becomes essential. A number of different lattice Boltzmann method implementation schemes were proposed to improve the efficiency of the algorithm. A parallel implementation can significantly improve the efficiency of the lattice Boltzmann method code. Several partition strategies [47, 48, 49, 50] were proposed to divide the computation domains into slices or boxes. These decomposition strategies are only efficient when the workload is distributed homogeneously over the lattice [35]. Kandhai *et al.*[11] proposed a new approach based on the Orthogonal Recursive Bisection (ORB) method. The ORB method can be used to generate approximately balanced decompositions by taking into account the workload on each lattice point. It is found to be 12% to 60% more efficient compared to conventional parallelization strategies.

A standard lattice Boltzmann method stores distribution functions for both fluid nodes and solid nodes. This wastes a large amount of memory and computing time, since no operation is carried out on solid nodes. A sparse data structure is required to reduce the memory and computing source usage. Implementation schemes with the name SHIFT were proposed to improve the efficiency of data storage in the computation [51]. The information of solid nodes is no longer stored in these schemes, and the information regarding adjacent nodes is compressed to save memory. Numerical tests were carried out on four rock samples with porosities ranging from 10% to 38% to verify the performance of these schemes. The results showed that the memory required is 36% to 82% smaller [51]. In this study, a optimised sparse storage scheme based on SHIFT is proposed (Chapter 9). This scheme is easy to implement and the numerical experiments carried out has proved that this scheme is of high efficiency.

3. Methodology

Concepts of the lattice Boltzmann method and its extensions including multiphase/multi-component, and thermal/solute dispersion will be introduced in this chapter; numerical examples are given in the next chapter to demonstrate the viability of the lattice Boltzmann method for the simulation of transport properties in porous media at the pore scale.

3.1. Kinetic theory and the Boltzmann equation

The Boltzmann equation was introduced by Ludwig Boltzmann to describe the behaviour of a real gas using a statistical distribution of the gas particles. The gas is composed of interacting particles that can be described by classical mechanics. But because the number of particles is extremely large ($\sim 10^{23}$), a statistical treatment was introduced to describe the average behaviour of the particles.

A function $f(x, \xi, t)$ is introduced to describe the average distribution of particles. It represents the number density of particles at time t and position x with velocity ξ. We consider the gas in a control volume $dV = [x, x + dx]$. An external force $F = ma$ is applied to the system. At time t, the number of particles with velocity $[\xi, \xi + d\xi]$ is $dN = f(x, \xi, t)d\xi dV$. After a time dt, if there is no collision, the location of these particles will be $x' = x + \xi dt$, the velocity will be $\xi' = \xi + adt$. Therefore, we have the following equations:

$$f(x + \xi dt, \xi + d\xi, t + dt)d\xi dx V - f(x, \xi, t)d\xi dx = 0 \tag{3.1}$$

or

$$\frac{\partial f}{\partial t} + \xi \cdot \nabla_x f + a \cdot \nabla_\xi f = 0. \tag{3.2}$$

Collisions may result in a change of the velocity of particles, so that the number of particles in the control volume $[x, x+dx] \times [\xi, \xi+d\xi]$ and $[x', x'+dx] \times [\xi', \xi'+d\xi]$ (which means particles are within $[x, x+dx]$, and the velocity of particles are within $[\xi, \xi+d\xi]$) will not be the same any more. A collision term $\Omega(f)$ is introduced to account for the change of distribution of particles. For an ideal gas, the collision term can be described as:

$$\Omega(f(\xi)) = \int [f'f'_1 - ff_1] B(\theta, |V|) d\theta d\epsilon d\xi_1 \tag{3.3}$$

where $f(x, \xi, t)$ and $f'(x, \xi', t)$ are the number densities before and after the collision, respectively. $V = \xi_1 - \xi$ is the relative velocity between two particles, θ is the angle between $\xi_1 - \xi$ and the the line linking the centres of two particles, ϵ is the projection angle of one particle over another, $B(\theta, |V|)$ is a non-negative function related to the interactions between the particles.

Let us introduce $\psi(\xi)$ as an arbitrary function of ξ. If we set $\psi = (1, \xi, |\xi^2|)$, we can prove that:

$$\int \Omega(f)\psi(\xi) d\xi = 0 \tag{3.4}$$

Any function which satisfies equation 3.4 is called collision invariant [52].

3.2. H-theorem

The H-theorem was introduced by Boltzmann in 1872 to describe the increase in the entropy of an ideal gas in an irreversible process. The H function is defined as:

$$H(t) = \overline{lnf} = \frac{\int flnf d\xi}{\int f d\xi} = \frac{1}{n} \int flnf d\xi \tag{3.5}$$

Boltzmann proved that the H function is a monotonically decreasing function with time:

$$\frac{\partial H}{\partial t} \leq 0. \tag{3.6}$$

When the H function reaches a minimum value (Equation 3.7), the system reaches equilibrium:

$$\frac{\partial H}{\partial t} = 0 \tag{3.7}$$

3.3. Maxwell distribution

Maxwell derived the probability equilibrium distribution for the speed of gas particles in 1860. It gives us the probability of a particle's speed being near a specified value as a function of temperature, velocity, and mass.

$$f = n\frac{1}{(2\pi R_g T)^{2/3}} exp[-\frac{(\xi - u)^2}{2R_g T}], \tag{3.8}$$

where R_g is gas constant and T is the thermodynamic temperature.

3.4. Boltzmann-BGK equation

In 1954, Bhatnagar, Gross and Krook [53] introduced an approximate expression Ω_f for the collision term $\Omega(f)$. They proved that a simplified expression Ω_f which replaces the collision term $\Omega(f)$ should satisfy the following two properties:

- For collision invariant $\psi = (m, m\xi, \frac{1}{2}m\xi^2)$, the equation

$$\int \psi \Omega_f d\xi = 0 \tag{3.9}$$

  should be satisfied.

- It should satisfy Boltzmann's H-theorem:

$$\int (1 + lnf)\Omega_f d\xi \leq 0 \tag{3.10}$$

They obtained the simplified term with the idea that the collision will lead the system to its equilibrium distribution f^{eq}. The rate of change is proportional to the difference of f^{eq} and f. The scale factor ν is a constant:

$$\Omega_f = \nu[f^{eq}(x, \xi) - f(x, \xi, t)] \tag{3.11}$$

The Boltzmann equation is then simplified to:

$$\frac{\partial f}{\partial t} + \xi \cdot \nabla_x f + a \cdot \nabla_\xi f = \nu(f^{eq} - f) \tag{3.12}$$

Equation 3.12 is called the Boltzmann-BGK equation. Bhatnagar, Gross and Krook also proved that Ω_f satisfies the properties 3.9 and 3.10. The equation includes a collision time $\tau = \frac{1}{\nu}$, also called relaxation time, indicating the time interval between two collisions. The Boltzmann-BGK equation is given as:

$$\frac{\partial f}{\partial t} + \xi \cdot \nabla_x f + a \cdot \nabla_\xi f = -\frac{1}{\tau}(f - f^{eq}) \tag{3.13}$$

The macroscopic fluid density, velocity and energy can be calculated from the microscopic distribution function:

$$\rho(x,t) = mn(x,t) = \int f(x,\xi,t)d\xi \tag{3.14}$$

$$n\mathbf{u}(x,t) = \int f(x,\xi,t)d\xi \tag{3.15}$$

$$nR_gT(x,t) = \frac{1}{D}\int (v-\mathbf{u})^2 f(x,\xi,t)d\xi \tag{3.16}$$

where m is the particle mass, D is the dimension of the space, ρ is the macroscopic density and T is the temperature. By applying a Chapman-Enskog expansion, the macroscopic equations for mass, momentum and energy can be derived from the Boltzmann equation. The bulk viscosity is derived as:

$$\nu = \frac{\tau R_g T}{m} \tag{3.17}$$

The equation of state (EOS) relating pressure and density is given by:

$$p = \rho R_g T \tag{3.18}$$

3.5. Single-phase lattice Boltzmann method

The lattice Boltzmann method (LBM) is a special discretization of the Boltzmann-BGK equation. Discretization of space, velocity and time are carried out in LBM. This procedure greatly simplifies the original Boltzmann equation. The location of particle distribution functions (PDFs) in space is restrained on the nodes of the lattice grid, and the particle velocity is simplified into a very limited number of lattice velocities. The number of discrete velocities is not unique, but the LB model using this discrete velocity model should be able to recover the macroscopic equations (in our study, the Navier-Stokes equations). We take a 2D model as an example.

This model, which is proposed by Qian *et al.* [29] is well known and widely used. It contains 9 velocities and is known as D2Q9. In LBM, we assume that all the particle distribution functions (PDFs) have the same uniform mass (normally taken as 1 for simplicity). The lattice unit (lu) and time steps (ts) are important length and time units in LBM. We only discuss uniform mesh in this chapter ($\Delta x = \Delta y$).

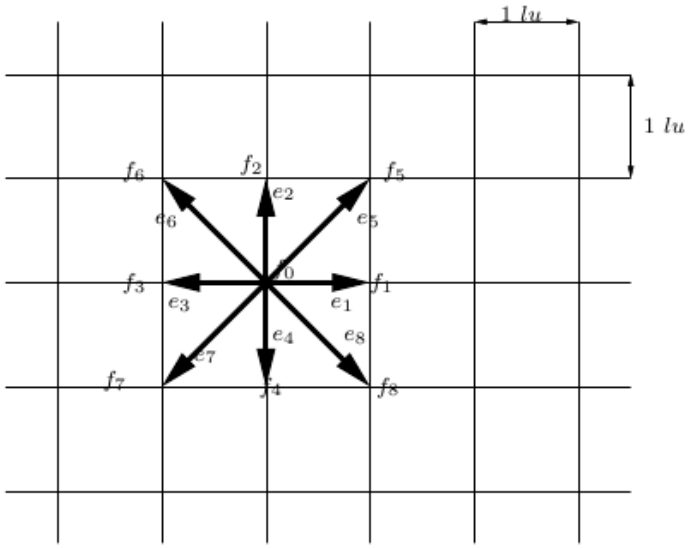


Figure 3.1.: D2Q9 lattice and velocities

Figure 3.1 shows the discretized velocity space $\{\mathbf{e}_i\}$, $(i = 0..8)$. The lattice velocity can be written as:

$$\mathbf{e} = e \begin{bmatrix} 0 & 1 & 0 & -1 & 0 & 1 & -1 & -1 & 1 \\ 0 & 0 & 1 & 0 & -1 & 1 & 1 & -1 & -1 \end{bmatrix} \tag{3.19}$$

where $e = \Delta x/\Delta y$ is the local lattice speed with a unit of $lu \cdot ts^{-1}$ (where lu is the lattice unit for length, ts is the time unit in LBM). The relation with the local speed of sound is given as $c_s = \frac{e}{\sqrt{3}}$.

The continuous distribution functions associated with velocity are written as $f_i(x,t)$, $(i = 0..8)$. We can obtain the lattice Boltzmann equations for

the D2Q9 model (Single Relaxation Time BGK) as:

$$f_i(\mathbf{x} + \mathbf{e}_i\Delta t, t + \Delta t) = f_i(\mathbf{x}, t) - \frac{f_i(\mathbf{x}, t) - f_i^{eq}(\mathbf{x}, t)}{\tau} \tag{3.20}$$

Collision of the particles can be considered as a relaxation process towards equilibrium. The equilibrium distribution function for the D2Q9 model is a truncated Maxwell-Boltzmann distribution (Equation 3.8) and is defined as [29]:

$$f_i^{eq}(x) = w_i\rho(x)[1 + 3\frac{\mathbf{e}_i \cdot \mathbf{u}}{e^2} + \frac{9(\mathbf{e}_i \cdot \mathbf{u})^2}{2e^4} - \frac{3\mathbf{u}^2}{c^2}] \tag{3.21}$$

where the weight coefficients for the D2Q9 model are:

$$w_i = \begin{cases} 4/9 & i = 0 \\ 1/9 & i = 1..4 \\ 1/36 & i = 5..8 \end{cases} \tag{3.22}$$

The macroscopic transport equations for mass, momentum and energy can be derived from the Boltzmann equation using a Chapman-Enskog expansion [54]. The kinematic viscosity ν in the D2Q9 model is obtained as:

$$\nu = c_s^2(\tau - \frac{1}{2})\Delta t \tag{3.23}$$

Note that $\tau > 1/2$ for positive viscosity. Numerical difficulties can arise as τ approaches $1/2$. The pressure is given by the equation of state for an ideal gas [6]:

$$P = \frac{nRT}{V} \tag{3.24}$$

In single phase LBM, $RT = c_s^2 = \frac{1}{3}$, so that the pressure can be computed as:

$$p = \rho c_s^2 \tag{3.25}$$

To implement a lattice Boltzmann simulation, four major steps should be included in the code:

- Initialisation of distribution function $f_i(\mathbf{x}, 0)$

- Collision step

$$f_i'(\mathbf{x},t) = f_i(\mathbf{x},t) - \frac{f_i(x,t) - f_i^{eq}(x,t)}{\tau} \tag{3.26}$$

- Streaming step

$$f_i(\mathbf{x} + \mathbf{e}_i\Delta t, t + \Delta t) = f_i'(\mathbf{x},t) \tag{3.27}$$

- Computation of macroscopic hydrodynamic quantities

$$\rho(\mathbf{x},t) = \textstyle\sum_i f_i(\mathbf{x},t) \tag{3.28}$$

$$\rho\mathbf{u}(\mathbf{x},t) = \textstyle\sum_i \mathbf{e}_i f_i(\mathbf{x},t) \tag{3.29}$$

where $f_i'(x,t)$ represents the value of the distribution function $f_i(x,t)$ after collision.

3.6. Bounce-back Boundary Conditions

Bounce-back boundary conditions play a major role in the LBM simulation due to their simplicity, versatility and powerful capability of dealing with extremely complex boundaries. This boundary condition is usually used at fluid-solid interfaces due to its correspondence to the no-slip condition. This boundary condition is illustrated in Figure 3.2. The densities moving toward the solid are bounced back into the fluid domain along reversed incoming directions. In the D2Q9 model, as shown in Figure (3.1) the bounce-back condition can be described in terms of equations as:

$$f_2 = f_4, \quad f_5 = f_7, f_1 = f_3, \quad f_6 = f_8 \tag{3.30}$$

The standard bounce-back condition places the boundary on the lattice nodes. Although mass and momentum are conserved, the accuracy is first order, while LBM is of second order [54]. Inamuro [55] found that the error produced by single relaxation time LBM with bounce-back condition is sufficiently small and of second order if the relaxation parameter τ is close enough to 2. The bounce back conditions can be used without any influence on the order of the LBM, if τ is chosen in the range (0.5,2). Bounce-back

conditions in multi-relaxation-time LBM are always of second order and not dependent on the selection of relaxation parameters [44]. Furthermore, the bounce back condition is the most efficient one for arbitrarily complex geometries [56]. Many researchers contributed to this ongoing discussion [50, 57]. A second order scheme with the name half-way bounce back condition was proposed by Ziegler [57]. In this boundary condition, the surface is a solid boundary placed between two neighbouring lattice sites with the same distance $\Delta x/2$. It is illustrated in Figure (3.2). This half-way boundary condition has been implemented in our simulator.

The boundary condition can be integrated in the collision step. The particles in the fluid domain will follow the collision law (Equation 3.26), whereas the particles in the solid domain follow the bounce-back rule defined in Equation (3.30).

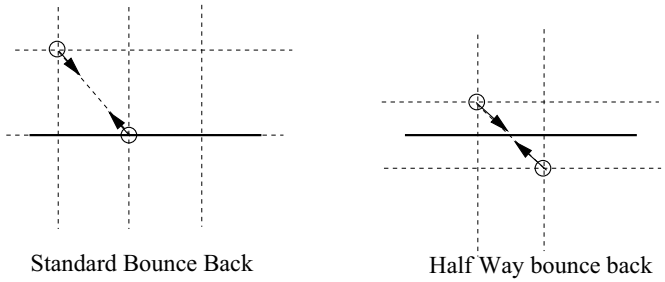


Figure 3.2.: Bounce-back condition

3.7. Periodic Boundary Condition

The periodic boundary condition could be the simplest boundary condition for LBM. In the periodic system, the fluid flowing out through one face will re-enter into the opposite face of the domain. So the edges of the simulation domain could be treated as if they are attached to the other side of the domain. For the boundary nodes, their neighbouring nodes are located at the opposite side of the boundary. (Fig 3.3)

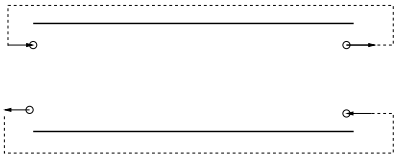


Figure 3.3.: Periodic boundary condition

Assuming that the length of computational domain is L, the periodic boundary condition is described by the formula (taking the x direction as an example, the derivation for the y direction is straightforward):

$$f_i(0, y, z, t + \Delta t) = f_i'(L, y, z, t) \tag{3.31}$$

$$f_i(L, y, z, t + \Delta t) = f_i'(0, y, z, t) \tag{3.32}$$

3.8. Fixed Pressure or Velocity Boundary

The fixed pressure or velocity boundary can be achieved by updating the distribution function after the streaming step with an equilibrium distribution function that has the desired pressure/velocity.

Guo *et al.*[58] proposed another approach to apply a pressure/velocity boundary. The distribution function after collision was decomposed into two parts; equilibrium and non-equilibrium: $f(x,t) = f^{eq}(x,t) + f^{neq}(x,t)$. The distribution functions of boundary points are updated using the non-equilibrium distribution from the neighbouring points, and the equilibrium distribution value computed using the desired pressure/velocity. He showed that this scheme is numerically stable and of second order.

3.9. Multi-Relaxation-Time (MRT) scheme for the lattice Boltzmann method

MRT allows independent adjustment of bulk [1] and shear viscosities which significantly improves the numerical stability for a low viscosity fluid. In the single relaxation time LB model, the collision term is relaxed by a single parameter τ, while it could be relaxed instead by a matrix Λ:

$$f_i(x + c_i\Delta t, t + \Delta t) - f_i(x, t) = -\Lambda_{ij}[f_j - f_j^{eq}], \quad i = 1, 2, \ldots, b \tag{3.34}$$

The matrix Λ is a full matrix of constants. For the D2Q9 model, the transformation matrix M is given as [33]:

$$M = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ -4 & -1 & -1 & -1 & -1 & 2 & 2 & 2 & 2 \\ 4 & -2 & -2 & -2 & -2 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & -1 & 0 & 1 & -1 & -1 & 1 \\ 0 & -2 & 0 & 2 & 0 & 1 & -1 & -1 & 1 \\ 0 & 0 & 1 & 0 & -1 & 1 & 1 & -1 & -1 \\ 0 & 0 & -2 & 0 & 2 & 1 & 1 & -1 & -1 \\ 0 & 1 & -1 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & 1 & -1 \end{pmatrix} \tag{3.35}$$

The single relaxation time LB model can be obtained by specifying Λ as a diagonal matrix with identical value:

$$\Lambda_{ij} = \frac{1}{\tau}\delta_{ij} \tag{3.36}$$

The macroscopic quantities are calculated in the same way as in the LBGK model. Instead of considering distribution functions, MRT employs

[1] Bulk viscosity, also called volume viscosity is important for simulations where fluid compressibility is essential. It appears in the compressible Navier-Stokes equation:

$$\rho(\frac{\partial v}{\partial t} + v \cdot \nabla v) = -\nabla\rho + \mu\nabla^2 v + f + \mu^v \nabla(\nabla \cdot v) \tag{3.33}$$

where μ^v is the bulk viscosity. In the incompressible Navier-Stokes equation, this term disappears because the divergence of the velocity of an incompressible fluid, $\nabla \cdot v$, equals 0

several moments corresponding to macroscopic quantities and their flux. These quantities can be relaxed with different time scales. A matrix M transforms the distribution functions f_i from the distribution space to the moment space:

$$m = M \cdot f, \quad f = M^{-1} \cdot m \tag{3.37}$$

The moment space for the D2Q9 model is:

$$m = (\rho, e, e^2, j_x, q_x, j_y, q_y, p_{xx}, p_{xy})^T \tag{3.38}$$

where e is the energy, j_x, j_y are the momenta in x and y directions, q_x, q_y are energy fluxes and p_{xx}, p_{xy} is the stress tensor. The collision is carried out in the moment space by multiplying the transformation matrix M; the left and right hand side of Equation 3.39 can be transformed into the moment space as:

$$f_i'(x,t) = f_i(x,t) - \Lambda_{ij}[f_j - f_j^{eq}] \tag{3.39}$$

$$\mathbf{m}' = \mathbf{m} - \mathbf{S}[\mathbf{m} - \mathbf{m}^{eq}] \tag{3.40}$$

where $\mathbf{m}^{eq} = \mathbf{M}\mathbf{f}^{eq}$ is the equilibrium equation in moment space. $S = M\Lambda M^{-1} = diag(s_1, s_2, \ldots, s_b)$. The corresponding relaxation time for moment m_i is s_i^{-1}. After the collision step, the moment m' is transformed back into distribution function space by multiplying M^{-1} for the streaming step $f_i(x + e_i\Delta t, t + \Delta t) = f_i'(x,t)$ which will be carried out in the same way as in the single relaxation time LB model.

The relaxation parameters and equilibrium functions of the moments are:

$$S = (0, s_e, s_{e^2}, 0, s_q, 0, s_q, s_\nu, s_\nu) \tag{3.41}$$

$$m^{eq} = \rho(1, -2 + 3u^2, \alpha + \beta u^2, u_x, -u_x, u_y, -u_y, u_x^2 - u_y^2, u_x u_y)^T \tag{3.42}$$

where α and β are adjustable parameters and may be chosen as $\alpha = 1, \beta = -3$ [31]. The kinematic viscosity and volume viscosity are given by

$$\nu = c_s^2(\tfrac{1}{s_\nu} - \tfrac{1}{2})\Delta_t \tag{3.43}$$

$$\zeta = c_s^2(\tfrac{1}{s_e} - \tfrac{1}{2})\Delta_t \tag{3.44}$$

where c_s is the local sound speed that equals $1/\sqrt{3}$ in the single phase LB model, s_ν and s_e are parameters given by users to set the kinematic viscosity and volume viscosity.

3.10. Multi-component lattice Boltzmann models

3.10.1. The Shan-Chen pseudo potential model

We start with the standard LBM using the Bhatnagar-Gross-Krook (BGK) collision term (see Chapter 3.5). In the lattice Boltzmann method (LBM), fictional particle groups on lattice nodes with discrete velocities are used to describe fluids. A distribution function $f_i(\mathbf{x}) = f(\mathbf{x}, \mathbf{e}_i)$ is used to describe the occupation of each lattice site. Each lattice velocity $\mathbf{e}_i$ on each lattice node $\mathbf{x}$ has a distribution function $f_i(\mathbf{x})$.

We recall the evolution of the distribution functions is described by the BGK collision terms in Chapter 3.5:

$$f_i(\mathbf{x} + \mathbf{e}_i\Delta t, t + \Delta t) = f_i(\mathbf{x}, t) - \frac{f_i(\mathbf{x}, t) - f_i^{eq}(\mathbf{x}, t)}{\tau} + K_i. \tag{3.45}$$

To incorporate a body force $\mathbf{F}$, an extra term K_i is included in the LBGK model. In this paper we use Guo's force term [58]:

$$\mathbf{K} = (1 - \frac{1}{2\tau})w_i[3\frac{\mathbf{e}_i - \mathbf{u}}{e^2} + 9\frac{\mathbf{e}_i \cdot \mathbf{u}}{e^4}\mathbf{e}_i] \cdot \mathbf{F}, \tag{3.46}$$

and the Exact Difference Method (EDM) [59, 41]:

$$K_i = f_i^{eq}(\rho, \mathbf{u} + \Delta\mathbf{u}) - f_i^{eq}(\rho, \mathbf{u}), \tag{3.47}$$

where the term $\mathbf{F}$ is the body force vector and $\Delta\mathbf{u} = \mathbf{F}\Delta t/\rho$. The macroscopic velocity $\mathbf{u}$ is computed as

$$\rho\mathbf{u}(\mathbf{x}, t) = \sum_i \mathbf{e}_i f_i(\mathbf{x}, t) + \frac{\Delta t\mathbf{F}}{2}. \tag{3.48}$$

Macroscopic transport equations for mass, momentum and energy can be derived from the Boltzmann equation using a Chapman-Enskog expansion [54]. The kinematic viscosity ν in the D2Q9 model is obtained as

$$\nu = c_s^2(\tau - \frac{1}{2})\Delta t. \tag{3.49}$$

An interaction force term between the particles is used to describe the interparticle forces:

$$\mathbf{F}(\mathbf{x}, t) = -G\psi_\sigma(\mathbf{x}, t)\sum_{i=1}^{8}\omega(|\mathbf{e}_i|)\psi_{\bar{\sigma}}(\mathbf{x} + \mathbf{e}_i\Delta t, t)\mathbf{e}_i, \tag{3.50}$$

where σ and $\bar{\sigma}$ denote the different fluid components. G is a parameter determining the interaction strength $F(\mathbf{x}, t)$ between neighbouring particles. It also determines whether the interaction is attractive or repulsive. To simulate a binary immiscible fluids system, the value of G should be kept positive so that a force will be generated to separate the fluids away from the interface. ψ_σ is the effective number density which is taken as the component density : $\psi_\sigma = \rho_\sigma$ [40]. $\omega(|\mathbf{e}_i|)$ is a parameter related to the strength and order of isotropy of the interaction forces, as shown in Table 3.1.

Order of Isotropy	$\omega(1)$	$\omega(2)$	$\omega(3)$	$\omega(4)$	$\omega(5)$	$\omega(6)$	$\omega(8)$
4	1/3	1/12					
6	4/15	1/10		1/120			
8	4/21	4/45		1/60	2/315		1/5040

Table 3.1.: Weights that yield fourth, sixth and eighth order of isotropy interaction forces [1]

To incorporate the interaction force, Shan and Chen proposed a force term scheme which only shifts the equilibrium velocity [60]. However, this force term scheme has been reported to be correct only if the relaxation time $\tau = 1$ [59, 42]. This unfavourable feature can be eliminated using Guo's force term or the Exact Difference Method [58, 61]. Both force terms have been incorporated in the Shan-Chen LB model and found to be independent of the equilibrium properties of τ; the difference between the Shan-Chen force term was shown to be on the order $O(F^2)$ [59, 62, 63]. To avoid the dependence of the equilibrium properties on the relaxation time, Guo's force term is used for the Free Energy and Colour Gradient Models whereas the Exact Difference Method is used for the Shan-Chen model in this paper.

Shan [1] studied the spurious currents phenomenon of the Shan-Chen model. He concluded that spurious currents are due to insufficient isotropy of the discrete gradient operator. For this reason, a new method of obtaining the discrete gradient operator with higher order isotropy was proposed. However, more adjacent nodes are needed to calculate the gradient operator. This will increase the amount of message exchanges required in parallel computing. Unfortunately, this method reduces the level of inherent parallelism of the lattice Boltzmann method. In this paper, we use a fourth and eighth order interaction force scheme [1] for the Shan-Chen multi-component model.

3.10.2. The Free Energy Model

Swift *et al.* [38] developed a thermodynamically consistent binary fluid LB model by introducing an equilibrium state associated with a free energy functional, corresponding pressure tensors and chemical potentials. A correct choice of the collision rules ensures that the system evolves towards minimisation of the free energy functional.

The thermodynamic properties of a binary fluid system can be described by a Landau free energy functional

$$\Psi = \int_V (\psi_b + \frac{\kappa}{2}(\partial_\alpha \phi)^2) dv + \int_S \psi_s ds, \tag{3.51}$$

where ψ_b is the bulk free energy density and has the form

$$\psi_b = \frac{c^2}{3}\rho \ln \rho + \frac{A}{4}\phi^2(-2+\phi^2). \tag{3.52}$$

ψ_s is the surface energy density and has the form:

$$\psi_s = -\frac{3}{2}\sigma sign(\frac{\pi}{2} - \theta)\sqrt{\cos(\frac{\alpha}{3} - \cos(\frac{\alpha}{3}))}\phi \tag{3.53}$$

A is a constant and was set as 0.04 in all simulations. σ is the surface thension, θ is the equilibrium contact angle and $\alpha = \cos^{-1}(\sin^2\theta)$. ϕ is the order parameter representing the concentration of components, defined as

$$\phi = \frac{\rho_a - \rho_b}{\rho_a + \rho_b}. \tag{3.54}$$

The hydrodynamics and thermodynamics of the binary fluids are described by the Navier-Stokes equations and a convection-diffusion equation:

$$\frac{\partial \phi}{\partial t} + \nabla(\phi \mathbf{u}) = \mu \nabla^2 \phi, \tag{3.55}$$

where the parameter μ determines the diffusivity of the binary fluids system.

A new distribution function $g_i(x,t)$ is introduced to describe the concentration $\phi = \sum_i g_i$ and is related to the convection and diffusion. The distribution function $f_i(x,t)$ is related to the fluid density and momentum as usual.

The time evolution equation uses the Multiple-Relaxation-Time (MRT) scheme and can be described as:

$$\text{Collision Step:} \quad f_i'(\mathbf{x},t) = f_i(\mathbf{x},t) - M^{-1}SM(f_i - f_i^{eq}), \tag{3.56}$$

$$g_i'(\mathbf{x},t) = g_i^{eq}(\mathbf{x},t). \tag{3.57}$$

$$\text{Streaming Step:} \quad f_i(\mathbf{x} + \mathbf{e}_i\Delta t, t + \Delta t) = f_i'(\mathbf{x},t), \tag{3.58}$$

$$g_i(\mathbf{x} + \mathbf{e}_i\Delta t, t + \Delta t) = g_i'(\mathbf{x},t). \tag{3.59}$$

Using an appropriate choice of the equilibrium distribution functions, it is possible to reproduce the macroscopic equations in the continuum limit [38].

3.10.3. The Colour Gradient lattice Boltzmann Model

An immiscible fluid model developed from Lattice Gas Cellular Automata was introduced by Gunstensen *et al.* [34]. The particles in this model are coloured either red or blue and therefore it is normally called the colour gradient method. The surface tension is introduced by adding a perturbation to the collision operator while keeping the adherence to the Navier-Stokes equations in homogeneous regions. A recolouring step is invoked after the surface tension perturbation calculation in order to achieve zero diffusivity of one colour into the other.

We use f_i^r, f_i^b and f_i to denote the distribution functions of the red fluid, the blue fluid and their combination respectively. A perturbation is computed

to generate the surface tension. The surface tension can be expressed as a local anisotropy in the pressure: the pressure measured normal to the surface is larger than that tangential to the surface. The pressure in a LBM is proportional to the density, so surface tension can be generated by preferentially placing particles in directions normal to the interface rather than tangential. Mass and momentum should be conserved. The colour gradient $\mathbf{C}$ is defined as:

$$\mathbf{C}(\mathbf{x},t) = \sum_i \mathbf{e}_i \sum_j (f_j^r(\mathbf{x}+\mathbf{e}_i\Delta t, t) - f_j^b(\mathbf{x}+\mathbf{e}_i\Delta t, t)). \tag{3.60}$$

Perturbation of the populations gives

$$f_i''(\mathbf{x},t) = f_i'(\mathbf{x},t) + \sigma|\mathbf{C}(\mathbf{x},t)|(\frac{(\mathbf{e}_i\cdot\mathbf{C})}{\mathbf{C}\cdot\mathbf{C}} - \frac{1}{2}), \tag{3.61}$$

where σ is a parameter to set the surface tension.

An alternative optimised perturbation term for the Colour Gradient LB model has been developed by Kehrwald [64] to improve the stability and accuracy. A further improvement based on the MRT collision scheme was proposed by Tölke *et al.* [65], reducing the spurious velocity, increasing the numerical stability and removing the dependency on viscosity. A redistribution of colour forces the particles to move towards the regions occupied by particles of the same colour. This recolouring step enables us to achieve separation of the fluids. It is carried out by the maximisation problem as given in Equation 3.62, see [34]. Because the particles are forced to stay with the particles having the same colour, the Colour Gradient model is only able to simulate completely immiscible fluids.

$$W(f_i^{r''}, f_i^{b''}) = \max_{f_i^{r''}, f_i^{b''}} [\sum_i (f''_{r_i} - f''_{b_i})e_i]. \tag{3.62}$$

Subject to constraints:

$$\rho_r'' = \sum_i f_i^{r''} = \rho_r, \tag{3.63}$$

$$f_i^{r''} + f_i^{b''} = f_i. \tag{3.64}$$

A general colour gradient lattice Boltzmann model can be summarised as:

- Single phase collision using MRT scheme.
- Add a surface tension perturbation to f_i' obtaining f_i''.
- Recolouring.
- Streaming.

In both the Free Energy model and the Colour Gradient model, only the value of distribution functions of the nearest neighbour nodes are needed, which keeps the inherent parallelism of the lattice Boltzmann method.

3.10.4. Optimised Colour Gradient lattice Boltzmann Model

An optimised Colour-Gradient approach was proposed by Ahrenholz *et al.* [66] for multicomponent flow calculations. This model, which is based on the Rothman-Keller/Gunstensen model [34], improves the numerical stability and permits higher viscosity ratio and lower capillary number calculation [66]. It uses only one full sized distribution function for the pressure, velocity field. Two extra LB equations of which the distribution function values do not need to be stored are used to simulate the evolution of interface movements along with the velocity. Additional terms to the equilibrium moments in the MRT collision step are given to generate surface tension while a recolouring algorithm was performed to limit the diffusion near the interface.

We use ρ_r, ρ_b to indicate the densities of two components respectively. An order parameter ϕ is defined as:

$$\phi = \frac{\rho_r - \rho_b}{\rho_r + \rho_b}. \tag{3.65}$$

The color gradient vector $\mathbf{C}$ of the phase field can be calculated using the following equation [66]:

$$\mathbf{C}(t, x) = \frac{3}{c^2 \Delta t} \sum_i w_i e_i \phi(t, x + e_i \Delta t), \tag{3.66}$$

where the weights w_i are:

$$w_i = \begin{cases} 1/3 & i = 0 \\ 1/8 & i = 1..6 \\ 1/36 & i = 7..18 \end{cases} . \tag{3.67}$$

The orientation of the interface can be obtained by using a normalised gradient:

$$n_\alpha = \frac{C_\alpha}{|C|}. \tag{3.68}$$

Two separate LB equations are used to compute the advection of density fields of ρ_r, ρ_b, we give the equation of the red component as an example, the LB equation for blue component can be obtained similarly [66].

$$g_i(t + \Delta t, x + e_i \Delta t) = g_i^{eq}(\rho_r(t, x), u(t, x)). \tag{3.69}$$

The equilibrium distribution function g_i^{eq} is:

$$g_i^{eq} = w_i \rho_r (1 + \frac{3}{c^2} e_i \cdot u) \tag{3.70}$$

According to Equation (3.69), we know that in the collision step, only the equilibrium distribution function, which can be computed locally, is needed. As a result, it's not necessary to store all the values of the distribution functions, only the summation $\rho_r = \sum_{i=0}^{18} g_i$ needs to be stored. This new approach, which is a simplified LB equation with $\frac{\Delta t}{\tau} = 1$, decreases the memory requirement and consequently improves the computing efficiency.

A recolouring scheme used in the original Colour gradient model [36] is employed in this approach between collision and streaming step in order to minimise the diffusion near the interface.

The contribution of surface tension to the moments can be computed with the gradient of the phase field. The additional terms related to surface

tension generation are given as [66]:

$$m_1 = -\sigma|C|(n_x^2 + n_y^2 + n_z^2) = \sigma|C|, \tag{3.71}$$

$$m_9 = \tfrac{1}{2}\sigma|C|(2n_x^2 - n_y^2 - n_z^2), \tag{3.72}$$

$$m_{11} = \tfrac{1}{2}\sigma|C|(n_y^2 - n_z^2), \tag{3.73}$$

$$m_{13} = \tfrac{1}{2}\sigma|C|(n_x n_y), \tag{3.74}$$

$$m_{14} = \tfrac{1}{2}\sigma|C|(n_y n_z), \tag{3.75}$$

$$m_{15} = \tfrac{1}{2}\sigma|C|(n_x n_z). \tag{3.76}$$

The method for adjusting the contact angle between fluid and solid is the same as the original colour gradient LB model and can be found in [67]. The parameter selection of relaxation matrix, kinematic and bulk viscosity calculation are performed as normal MRT-LBM [33].

3.11. LB method for Solute/Heat transfer

In this chapter, we introduce a solute/thermal LB scheme proposed by Sukop *et al.* [6] and Yoshino *et al.* [68] to study thermal and solute transport in porous media. A passive scalar approach was used to solve the convection-diffusion equation. A second distribution function g with a simpler equilibrium distribution was introduced to describe the evolution of the solute concentration:

$$g_i^{eq} = \omega_i \rho_s (1 + 3e_i \cdot u) \tag{3.77}$$

where the component ρ_s indicates the local concentration of the solute:

$$\rho_s = \sum_{i=0}^{a} g_i \tag{3.78}$$

The following LB equation is used to compute the evolution of the distribution of solute:

$$g_i(r + c_i\Delta t, t + \Delta t) - g_i(r,t) = -\frac{1}{\tau_g}[g_i(r,t) - g_i^{eq}(r,t)] \tag{3.79}$$

The diffusion coefficient D_s is given by

$$D_s = \frac{1}{3}(\tau_s - 0.5) \tag{3.80}$$

As shown in [69] and [70] the governing equation at the macro scale can be recovered by applying a Chapman-Enskog expansion to Equations 3.79 and 3.78:

$$\frac{\partial \rho_s}{\partial t} = -\nu \nabla \rho_s + D_s \Delta \rho_s \tag{3.81}$$

3.12. Macroscopic governing equation

In this chapter, the relationship between the lattice Boltzmann equation and Navier-Stokes equation is given. A series of partial differential equations in terms of $\rho, \boldsymbol{V}, p$, which describe the lattice Boltzmann equation in the limits of $\Delta x, \Delta t \to 0$, are derived to recover the mass and momentum conservation equations. Chapman-Enskog expansion has been widely used to derive the macroscopic equation of the lattice Boltzmann method [71, 72, 67, 58], Skordos introduced an approach based mainly on Taylor series analysis and Chapman-Enskog expansion was only used in the derivation of the momentum conservation equation [47]. The following macroscopic equation derivation for the D2Q9 lattice Boltzmann method is based on Skordos's approach.

First, we review the standard lattice Boltzmann equation using the D2Q9 model. To simplify the derivation, we assume that $\Delta x = \Delta t = 1$.

$$f_i(x + e_i \Delta t, t + \Delta t) = f_i(x, t) + \frac{f_i^{eq} - f_i}{\tau} \tag{3.82}$$

where f^{eq} is the equilibrium distribution function and can be computed by:

$$f_i^{eq}(x) = \omega_i \rho(x)[1 + 3\boldsymbol{e_i} \cdot \boldsymbol{V} + \frac{9(\boldsymbol{e_i} \cdot \boldsymbol{V})^2}{2} - \frac{3\boldsymbol{V}^2}{2}] \tag{3.83}$$

where the weights $\omega_i = 4/9$ for $i = 0$, $1/9$ for $i = 1, 2, 3, 4$ and $1/36$ for $i = 5, 6, 7, 8$. The equilibrium distribution conserves the mass and momentum

locally:

$$\sum_{i=0}^{8} f_i = \sum_{i=0}^{8} f_i^{eq} \tag{3.84}$$

$$\sum_{i=0}^{8} f_i \boldsymbol{V} = \sum_{i=0}^{8} f_i^{eq} \boldsymbol{V} \tag{3.85}$$

$$\tag{3.86}$$

The macroscopic quantities density ρ and velocity $\boldsymbol{V}$ can be computed as:

$$\sum_i f_i = \rho, \tag{3.87}$$

$$\sum_i \boldsymbol{e_i} f_i = \rho \boldsymbol{V} \quad . \tag{3.88}$$

Then, Equation 3.82 can be expanded into a Taylor series around the point (x, t), which has the form:

$$\Delta t(\frac{\partial}{\partial t} + \boldsymbol{e_i} \cdot \nabla) f_i + \frac{\Delta t^2}{2}(\frac{\partial}{\partial t} + \boldsymbol{e_i} \cdot \nabla)^2 f_i + \cdots = \frac{f_i - f_i^{eq}}{-\tau} \tag{3.89}$$

The right-hand side of Equation 3.89 vanishes due to the mass and momentum conservation of equilibrium distribution functions (Equation 3.87, 3.88). Expanding the distribution functions and the time and space derivatives in terms of the Kundsen number [73], ϵ:

$$f_i = f_i^{(0)} + \epsilon f_i^{(1)} + \epsilon^2 f_i^{(2)} + \cdots , \tag{3.90}$$

$$\partial_t = \epsilon \partial_{1t} + \epsilon^2 \partial_{2t} + \cdots , \tag{3.91}$$

Substitution of Equation 3.90,3.91 to Equation 3.89 and considering the terms $O(\epsilon)$ and $O(\epsilon^2)$ we can perform a Chapman-Enskog expansion to obtain the following equations:

$$\frac{\partial(\sum_i f_i)}{\partial t} + \nabla \cdot (\sum_i \boldsymbol{e_i} f_i) + \cdots = 0 \tag{3.92}$$

$$\frac{\partial(\sum_i \boldsymbol{e_i} f_i)}{\partial t} + \nabla \cdot (\sum_i \boldsymbol{e_i e_i} f_i) + \cdots = 0 \tag{3.93}$$

For Equation 3.92, we can easily convert it into an equation in terms of ρ and $\boldsymbol{V}$ by using Equations 3.87, 3.88. The result is:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \boldsymbol{V}) = 0 \tag{3.94}$$

which is the mass continuity equation. For Equation 3.93, a complication arises due to the pressure tensor term $(\sum_{i=0}^{8} \boldsymbol{e_i e_i} f_i)$. To transfer the pressure tensor in terms of ρ and V, an approximation of f_i is made with Chapman-Enskog expansion. The Chapman-Enskog expansion tells us that the distribution function f_i can be approximated with the equilibrium distribution function f_i^{eq} to zero order [47]:

$$f_i(x,t) = f_i^{eq}(x,t) + f_i^1(x,t) \tag{3.95}$$

where f_i^1 is the correction term to first order and so on. By substituting f_i by f_i^{eq}, which is its zero order estimation, in Equation 3.89, the correction term f_i^1 can be expressed as:

$$f_i^1 = -\tau \Delta t [\frac{\partial f_i^{eq}}{\partial t} + \boldsymbol{e_i} \cdot \nabla f_i^{eq}] \tag{3.96}$$

The f_i in Equation 3.93 is further replaced by $f_i^{eq}(x,t) + f_i^1(x,t)$ with Equation 3.96 and combined with the formula of the equilibrium distribution function f_i^{eq}, which has been given in Equation 3.83. Then we can produce an equation with ρ and $\boldsymbol{V}$ only, and take the x-direction as an example to derive the momentum equation:

$$\frac{(\rho \boldsymbol{V_x})}{\partial t} + \frac{(\rho \boldsymbol{V_x V_x})}{\partial x} + \frac{(\rho \boldsymbol{V_x V_y})}{\partial y} = \frac{\partial p}{\partial x} - \nu \nabla^2 (\rho \boldsymbol{V_x}) + \mu \frac{\partial(\nabla \cdot (\rho \boldsymbol{V}))}{\partial x} \tag{3.97}$$

where $\nu = \frac{1}{3}(\tau - \frac{1}{2})$, $p = \frac{1}{3}\rho$ and $\mu = \frac{4}{3}\nu$. We find that Equation 3.97 is the momentum equation of the Navier-Stokes equation if the fluid is

incompressible:

$$\rho(\frac{\partial \boldsymbol{V}}{\partial t} + \boldsymbol{V} \cdot \nabla \boldsymbol{V}) = -\nabla p + \nu \nabla^2 \boldsymbol{V} + (\frac{1}{3}\nu + \mu)\nabla(\nabla \cdot \boldsymbol{V}) \tag{3.98}$$

According to the derivation, we know that the kinetic viscosity ν and pressure p can be computed by $\nu = \frac{1}{3}(\tau - \frac{1}{2})$ and $p = \frac{1}{3}\rho$ respectively. Finally, we give a brief error analysis to illustrate that Equation 3.97 is of second order to the Navier-Stokes equations. In Equation 3.93, f_i has been replaced by f_i^{eq}, which is a zero order approximation. As only the derivative of f_i is needed, the substitution keeps the calculation of $f_i^{eq} + f_i$ first order while keeping the Taylor series in second order.

4. Single phase flow in porous media

The lattice Boltzmann method is able to handle extremely complex geometries and is a promising method for flow calculations on porous media. In this chapter, validation of single phase LB and flow calculation of reservoir rocks are shown. The permeability of the reservoir rocks is predicted using the LB code and compared to available experimental measurement data.

4.1. Introduction

For many scientific and industrial applications, it is important to know the permeability of a particular porous medium, such as sandstone. One important application is in the oilfield industry, where the production of hydrocarbons is often limited by the permeability of the reservoir rock. Recently, the lattice Boltzmann (LB) simulations have become available as a method to calculate the permeability of complex three-dimensional geometries. These simulations also give access to the full flow field at the pore scale. For single phase flow, the LB flow field generated can then be used to calculate hydrodynamic dispersion and displacement propagators in the porous medium available from NMR experiments [7]. Heijs and Lowe were among the first to use LB simulations to study flow through Fontainebleau sandstone [74]. Auzerais *et al.* [75] used LB simulations to calculate the permeability of Fontainebleau sandstone. They used X-ray microtomography (XMT) imaging and provided a direct comparison with laboratory measurements. Kang *et al.* [76] developed a unified LB method for flow in multiscale porous media. This model not only can simulate flow in porous systems of various length scales but also can simulate flow in porous systems where multiple length scales coexist. Simulations performed on a fractured porous system show that this method gives very good overall permeability values

for the whole fractured system. Although the list above is far from exhaustive, the papers mentioned provide a good introduction to the subject. In this study, we report on the calculation of the flow in a realistic complex three-dimensional (3D) models of four rock samples. In section 4.2 and 4.3 we introduce several numerical verifications on problems for which the analytical solution is available and the boundary condition is also studied. The porous media and the quantities computed are described in section 4.4. Results for the dependence of the permeability measurement on the system size of the computational sub-sample are reported in section 4.5. There we also investigate the effect of boundary conditions. We observe that for different boundary conditions, the permeabilities of the larger sub-samples are less dependent on the particular boundary conditions used than those of the smaller sub-samples.

4.2. Verifications for the single-phase lattice Boltzmann method

4.2.1. Poiseuille flow simulation and flux calculation in a narrow channel

First, Poiseuille flow in a channel with two parallel solid surfaces was studied. This model can be regarded as a simplified pore structure in porous media. Periodic boundaries are used in the flow direction. The analytical velocity profile in a slit of width $2a$ is parabolic and is given by the Poiseuille equation [77]:

$$u(x) = \frac{G^*}{2\nu}(a^2 - x^2) \tag{4.1}$$

where G^* is a pressure gradient or a body force applied on the fluid,x is the coordinate, u, ν is velocity and viscosity respectively. The flux across the outlet can be computed by $\int u(x)dx$. We compute the flux across the outlet of slits with different width by using the half-way and standard bounce-back boundary conditions for the walls, while periodic boundary conditions are used for the inlet and the outlet boundaries. The details of standard and half-way bounce back boundary conditions will be explained in section 4.3. The width varies from 3 to 21 lattice points. The simulated and

analytical velocity profiles are shown in Figure 4.1. Excellent agreement is obtained in the simulations for all widths. Even when the slit has only one cell spacing for the fluid, the LBM code still gave excellent results for the velocity estimation. This excellent agreement is due to the use of the half-way bounce-back boundary condition for the walls along the flow direction and the use of the Multiple-Relaxation-Time scheme (MRT). The accuracy of the half-way bounce-back boundary condition is second order [6], and the Poiseuille flow is a second order problem (analytical solution is shown in Equation 4.1). Therefore LBM can predict the velocity field perfectly. The MRT scheme eliminates the viscosity dependence of the velocity calculations. A standard lattice Boltzmann method simulation without the MRT scheme introduced an error in the velocity ranging from $8\% - 17.8\%$ depending on the viscosity of the fluid [44]. In our results in Figure 4.1, accurate velocity fields can be obtained with arbitrary viscosity values.

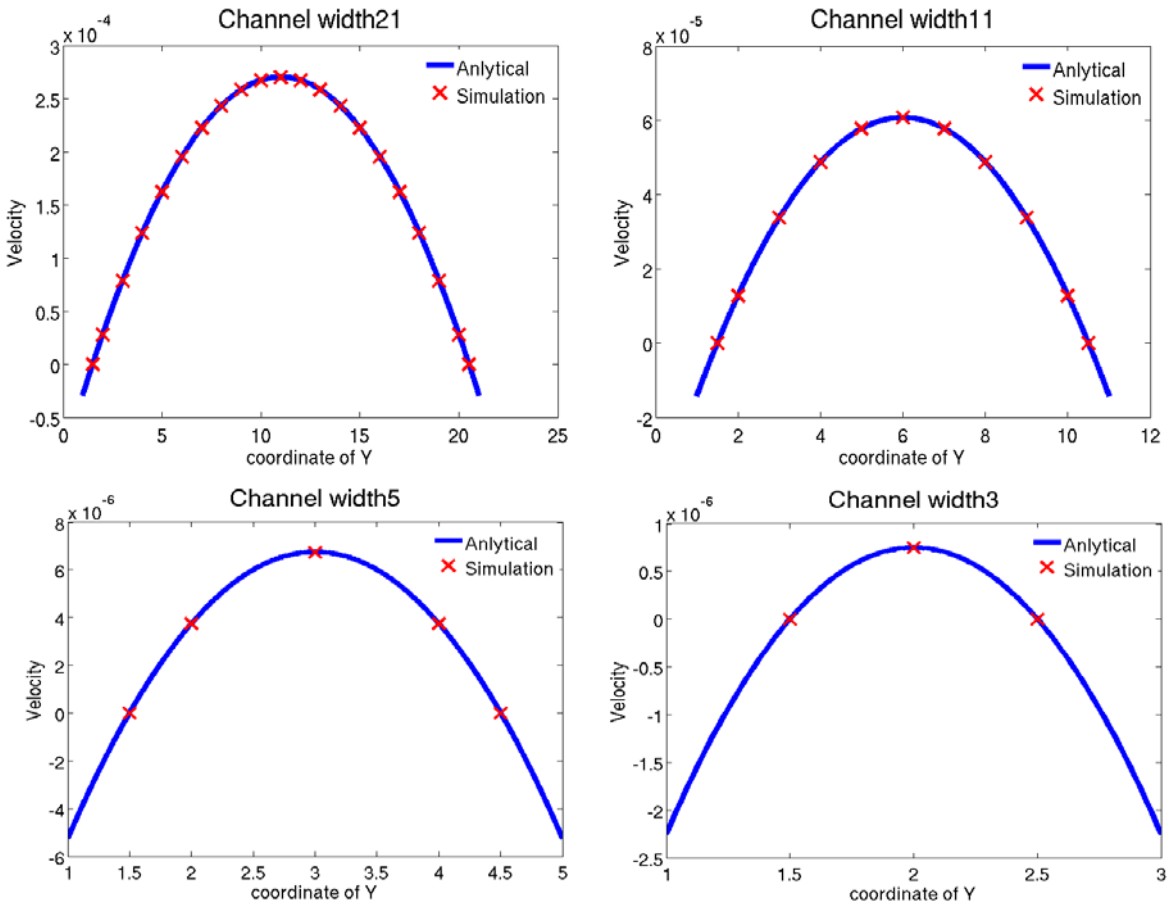


Figure 4.1.: Simulated velocity compared with analytical solution for different channel widths of 21, 11, 5, 3 in lattice unit

The flux is calculated by integrating the velocity along y. The simulated flux is calculated by summing up the velocity along the direction is perpendicular to the flow direction. Since we know from our previous simulation that the velocity profile is accurate, the calculation of the flux becomes a procedure of numerical integration. A midpoint rule is used to calculate the flux. Therefore, the more integration points we have, the higher accuracy we can obtain. This analysis matches our numerical experiment (Fig.4.2). When the channel spacing is only one cell, although we can predict the velocity accurately, the error in the flux calculation is 50%. However, when the channel is 3 lattice sites wide, the error in the flux calculation decreases

dramatically to 5%, which is an acceptable accuracy. This is an important observation for permeability calculations in larger systems. The error will continue to decrease as the width of channel increases. We can obtain very good results with a width bigger than 30.

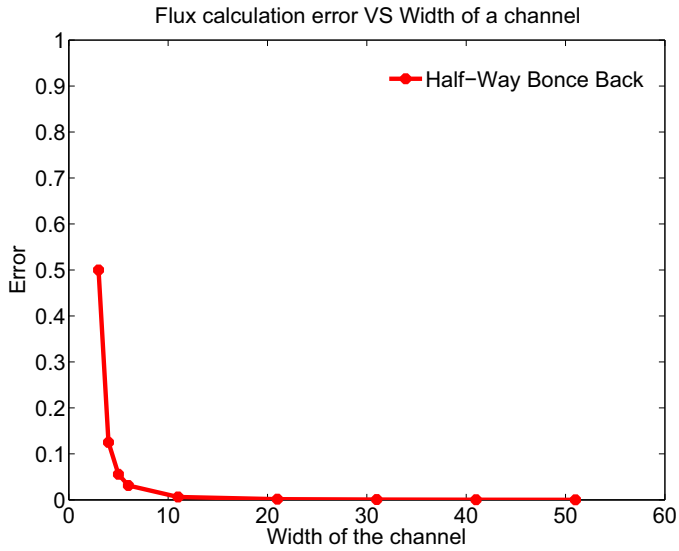


Figure 4.2.: Error of simulated flux of a slit

4.2.2. Flow through A Pipe

A simple 3D cylindrical tube model with a grid of 60x60x60 is studied with our code. Figure 4.3 shows the geometry of the pipe model. The porosity is dependent on the radius. The fluid velocity has a parabolic profile similar to Poiseuille flow. The analytical solution for this laminar Poiseuille flow is [2]:

$$Q = \frac{\pi R^4}{8\eta} \frac{\partial P}{\partial x} \tag{4.2}$$

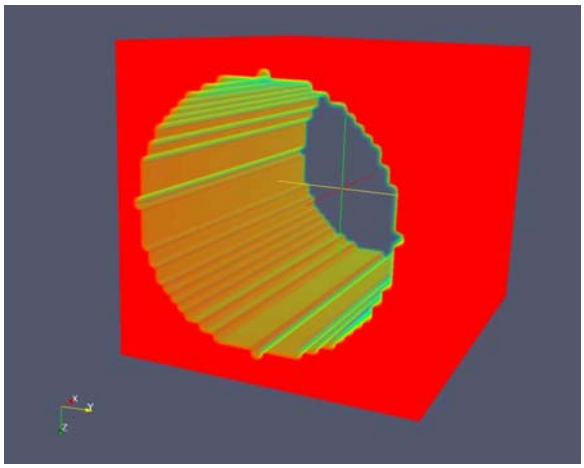


Figure 4.3.: The pipe model

where η is the dynamic viscosity, R is the radius of the tube and $\frac{\partial P}{\partial x}$ is the pressure gradient. The permeability κ is computed by applying Darcy's law:

$$\kappa = \frac{\pi R^4}{8A} \tag{4.3}$$

where A is the cross-sectional area. The calculated permeability from our code and Keehm's paper (using single relaxation time LB)[2] are shown in Figure 4.4. Good agreement with the theoretical prediction is obtained by both our code and Keehm's results. Relative errors in the percentage of the analytical solution with increasing radius of the tube are shown in Figure 4.5. The results show the trend of error decreasing from 40% to around 5% with tube radius increasing. This is because a big radius gives a finer mesh on the boundaries. Our code also gives a lower error of $2\% - 5\%$ compared with Keehm's method. These better results are probably due to the use of the MRT scheme.

Another interesting result can be observed in Figure 4.5: the error observed in a simulation with an integer radius is bigger than that for a non-integer radius. This reduction of error is due to the boundary condition. The input file of the boundary is a binary matrix with 0 and 1 that represents pore nodes or solid respectively. If the radius is one half larger than an

integer value, the actual boundary will be located between the pore nodes and solid nodes. This introduces an effective half-way bounce-back boundary condition [78]. Otherwise, a standard bounce-back boundary condition (SBB) is used. We have already seen that half-way bounce-back boundary conditions have a second order accuracy which is one order higher than the standard bounce-back boundary condition. This numerical experiment suggests that the use of half-way bounce-back boundary conditions allows more accurate calculation of the permeability. We can also learn from these simulation results that boundary conditions affect the permeability calculation significantly[78].

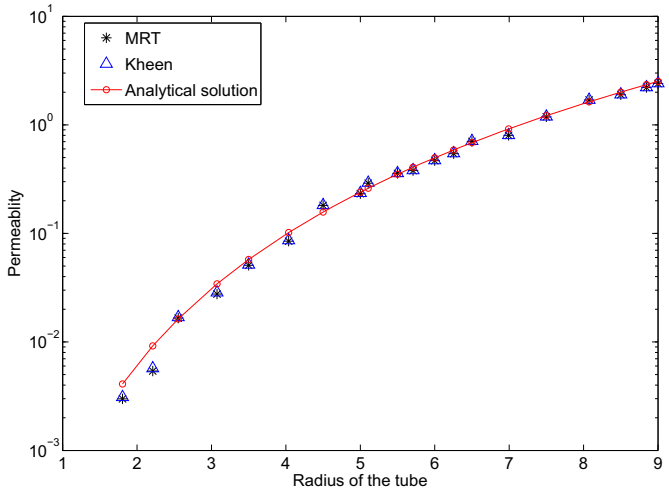


Figure 4.4.: Simulated permeability from our code, Keehm's paper[2] and the theoretical prediction as a function of tube radius, both are in lattice units

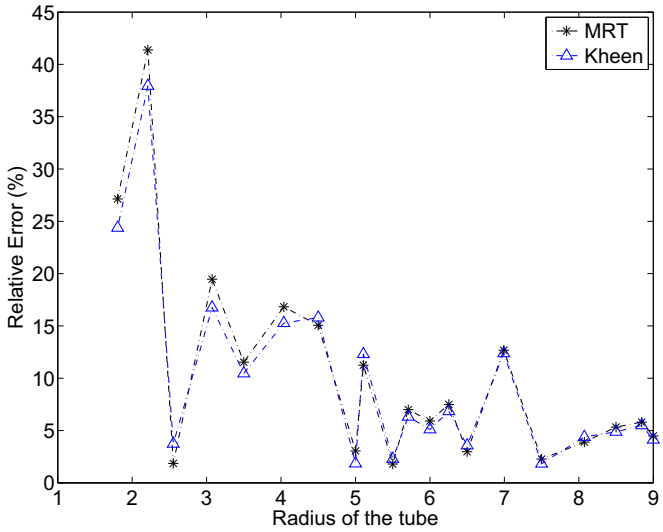


Figure 4.5.: Relative error in percentage of the analytical solution with the radius of the tube. My results are shown as black star points, Keehm's results[2] are shown as blue triangle points

4.2.3. Simulation of flow in fibrous porous media

A porous media with rectangular periodic arrays of cylinders with elliptical cross section [79] is investigated to evaluate the performance of our code. The square packing configuration for rows of elliptical fibres is shown in Figure 4.6. The LBM simulation domain is shown in Figure 4.7. Periodic boundary conditions are applied in x, y and z directions. The porosity is determined by the parameters Lx, Ly, a, b. An analytical formulation of the permeability K is derived by Phelan *et al.* [80]:

$$K = \frac{2L_y}{L_x} \frac{1}{\int_{-L_y/2}^{L_y/2} \frac{h^3}{3}} \tag{4.4}$$

where h is a function of the y-coordinate, as shown in Figure 4.7. It indicates the distance between the cylinder and the computational boundary. To evaluate our simulated permeability with this theoretical prediction, a dimensionless permeability K^* is defined as:

$$K^* = \frac{4K}{ab} \tag{4.5}$$

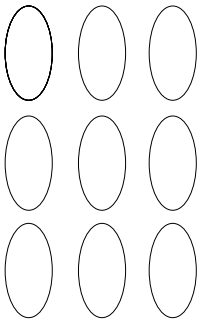


Figure 4.6.: Geometry of fibrous porous media

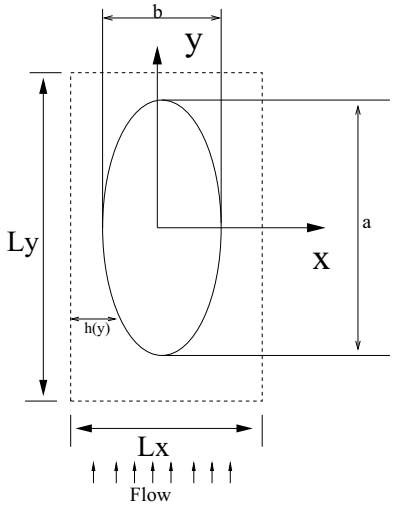


Figure 4.7.: LBM simulation domain

The permeability for a single cylinder with periodic boundary conditions is calculated using a 150x150 2D mesh in the simulation. The results are compared with Keehm's LB results [2] and the analytical solution. It can be seen from Figure 4.8 that good agreement with the analytical solution is obtained. For porous media with a low porosity (less than 23%), the prediction of the permeability from both Keehm's and our results showed a

big error (up to 47% for my results, 80% for Keehm's results). This is due to the lack of fine mesh between adjacent cylinders.

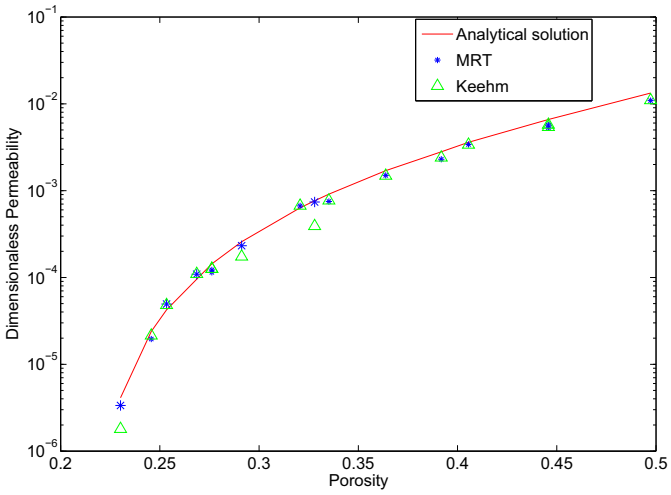


Figure 4.8.: Simulated dimensionless permeability of arrays of cylinders with elliptical cross sections

4.3. Boundary conditions at the solid nodes: a note of warning

To study flow in porous media at the pore scale using LB simulations, it is very important to use appropriate boundary conditions for fluid-rock interactions. Wall boundary conditions in the LB method were originally taken from the lattice gas (LG) method. For example, the bounce-back scheme is used at the walls to obtain no-slip velocity conditions. In the bounce-back scheme, when a particle distribution streams to a wall node, it scatters back to the node it came from. Because for a node near a boundary, some of its neighbouring nodes may be located outside the flow domain, the distribution functions from these non-fluid nodes are therefore unknown after each streaming process. The bounce-back scheme is a simple way to fix these unknown distributions on the wall nodes. It was found that the bounce-back condition is only first order in numerical accuracy at the boundaries [81, 82, 83]. However, if the location of the wall is placed halfway

between the wall node and the first adjacent fluid node, it can be shown that the method is second order. These results were proven both by numerical experiments and by the analytical solution of the LBGK equation for a Poiseuille flow in 2D [84, 85]. However, even with the half-way bounce-back boundary condition, the Poiseuille profile shows deviations from the analytical solution, i.e., it is still a parabola but with a shift at the boundary. The bounce-back boundary condition mimics boundaries that move with a speed that depends on the relaxation time τ. In the general case, the precise location of the wall at the zero slip velocity lies somewhere between the wall node and the neighbouring fluid node and can be determined by the eigenvalues of the linearised collision operator [83]. This unfavourable dependence of relaxation time τ can be removed using the MRT scheme. The effective wall is located at exactly halfway between the wall node and the neighbouring fluid node [44].

4.3.1. A brief review of some second order boundary conditions

A solution to these numerical errors are the so-called second order boundaries, for which the desired location of the non-slip boundaries is determined by extrapolating the distribution function from the last fluid points.

Inamuro *et al.* [86] proposed a method in which the unknown distribution functions are assumed to be an equilibrium distribution function with a counter slip velocity, which is determined so that the fluid velocity at the wall is equal to the wall velocity. The limitation of this method is that it can handle only flat boundaries and has problems with nodes at the corners.

Noble *et al.* [87] introduced a method to calculate the particle distribution at the boundaries from the velocity boundary conditions and the particle distributions of the neighbouring fluid nodes (utilising conservation of mass and momentum). They seek an answer to the question: knowing the velocity of the fluid at the boundaries, can an appropriate particle distribution be found such that imposing this distribution provides the prescribed velocity condition? This hydrodynamic approach seeks to maintain a specified velocity profile on the boundaries. Moreover, the density at the

boundary becomes a computed quantity. Since the density is related to the pressure through the isothermal equation of state, the algorithm they propose supplies the pressure boundary condition. However, this method also has the drawback that it can be applied only to simple geometries, such as flat boundaries.

Chen *et al.*[88] enforced the condition of correct flux of momentum near the wall as an extra condition. In this approach, the lattice Boltzmann scheme is regarded as a special finite difference of the kinetic equation for f_i in the Boltzmann equation for the discrete velocity distribution. These authors proposed boundary conditions as an extrapolation scheme, to allow the boundary nodes to evolve following the lattice Boltzmann relaxation dynamics. This is achieved by assuming that there is an additional layer of nodes inside the wall beyond the boundaries. The distribution functions of these additional nodes are calculated before the streaming operation using a second order extrapolation based on the distributions on the boundary layer and the neighbouring layer inside the fluid. After this extrapolation, the streaming step is carried out for all nodes. The collision step for the boundary (wall) nodes is done by enforcing the velocity (or pressure) boundary conditions for the equilibrium distribution functions.

Verberg and Ladd [89] proposed a continuous bounce-back boundary condition to incorporate solid-fluid boundary conditions on length scales smaller than the grid spacing. They assumed that the population densities are uniformly distributed throughout the volume of the Wigner-Seitz cell surrounding each node. For each node, a real numbered parameter is then introduced, which represents the fluid volume associated with that node. The distributions that stream to a solid node are reflected with rules that depend on this fluid fraction parameter.

Martys *et al.* [90] adopted a boundary condition in which the distribution of particles going out from the wall are set equal to their complementary incoming velocity distributions. This sets the normal velocity to zero. The remaining directions (in the plane of the wall) then have their distribution functions set to the average of the incoming directions, thus setting the tangential velocity to zero. During the collision step, the collision operator is

applied to the boundary nodes as well as to the fluid nodes, and the resulting distribution functions are then propagated normally in the next cycle [82]. This scheme might be a relatively simple solution to implement for lattice Boltzmann simulations of flow in porous media.

For practical simulations, however, the bounce-back condition is very attractive, because it is a simple and computationally efficient method for imposing no-flow conditions on irregularly shaped walls, such as in porous media. However, since bounce-back conditions do not give an accurate velocity, errors can be introduced in the results, such as a permeability that depends on the fluid viscosity [91].

Another source of error in the numerical quality of LB simulations, arises from the finite-size effects due to an insufficient number of lattice points compared to the mean free path of the fluid particles. These Knudsen-like effects depend on both the relaxation parameter (which controls the mean free path) and on the lattice resolution. The accuracy of the results increases with increasing grid resolution, and decreasing relaxation parameter (viscosity).

A linear dependence of the permeability on the viscosity was found in simulations of flow in porous media that implement the SRT bounce-back boundary conditions [92, 93, 94]. This viscosity dependence, however, becomes weak with increasing grid resolution. For MRT-LBM calculations, the permeability was found independent of the viscosity [44]. In this study, a MRT collision term is used and the viscosity is chosen as $\nu = 1/6$.

4.4. Materials and methods

4.4.1. Bentheimer sandstone

The data sets used in this study were extracted from the image of a cylindrical core of beadpack, LV60 sandpack, Bentheimer sandstone and Portland carbonate obtained by X-ray microtomography (XMT) at the European Synchrotron Research facility in Grenoble and Imperial College London. The scanning were done by reserchers in these two institutions. The XMT

image consists of voxels arranged in a three-dimensional lattice. The sizes, resolutions and petrophysical parameters of the sample is given in Table 8.1. A 3D volume rendering of the Bentheimer sandstone is shown in Figure 4.9.

Table 4.1.: Petrophysical parameters

	Sample size (in pixels)	Resolution	porosity	permeability
Beadpack	512x256x256	$5.0\mu m$	0.357	2624.53 mD
Sandpack LV60	512x256x256	$7.249\mu m$	0.323	62000 mD
Bentheimer sandstone	512x256x256	$4.9\mu m$	0.23427	4755 mD
Portland carbonate	512x256x256	$9.0\mu m$	0.0917	355mD

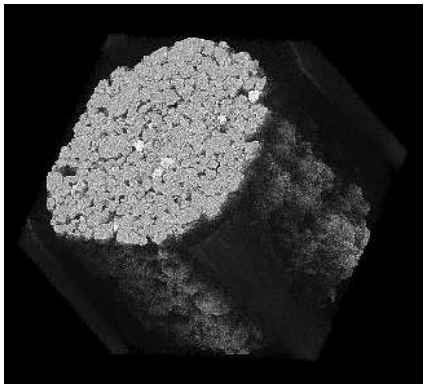
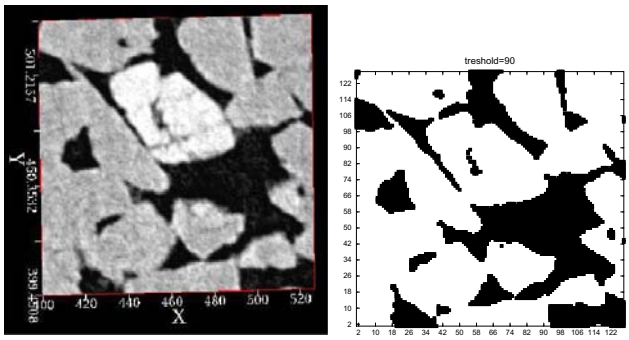


Figure 4.9.: 3D volume rendering of the XMT image of the Bentheimer sandstone used in this study.

In the image data, each voxel corresponds to a byte-value, which stores a greyscale value [0-255] that represents the attenuation of the rock at that point in space. This value is proportional to the density of the material: the higher the value, the denser is the system. Typically, the pore space has low attenuation values. The permeability and porosity of these particular samples of rock have not been measured experimentally. However, a permeability of 2 Darcy and a porosity of 22% are typical values for Bentheimer sandstones [95]. Using this estimated rock porosity, it is possible to determine a threshold in the greyscale, to discriminate between pores and rock, and byte-rate the XMT image to generate a binary (0 pore site, 1 rock site) representation of the rock and the pore space, as illustrated in Figure 4.10. This representation can then be directly loaded into the lattice Boltzmann

code, and used as the rock matrix (solid boundary) for the simulations. We also define the void fraction ϕ in a rock sample as the number of "fluid" lattice points (*i.e.* the lattice points which are not on a solid obstacle) divided by the total number of lattice points. The void fraction is related to the porosity of the rock, which can be characterised experimentally (e.g. by mercury porosimetry).



(a) greyscale (b) bitmap

Figure 4.10.: 2D slice of a subsample (128^3) of the Bentheimer sandstone used in this study. Figure (a) is the greyscale image from XMT data, and figure (b) is its bit-map representation using the value 90 for the threshold.

4.4.2. Conversion from lattice units to physical units

Length and time can be converted from physical units into lattice units in the following way. The unit of length is given by the lattice resolution δx. The unit of time, δt, can be derived from the kinematic viscosity. We define μ to be the kinematic viscosity in lattice units and μ_{mKs} the kinematic viscosity in mKs (SI) units. Considering that the kinematic viscosity has the dimensions of length squared over time, we have

$$\mu_{\mathrm{mKs}} = \mu\, \delta x^2 / \delta t. \tag{4.6}$$

The kinematic viscosity in lattice units can be calculated using Equation 3.23.

For example, if we take the XMT resolution for the Bentheimer data as the lattice spacing, *i.e.* δx=4.9μm, and the kinematic viscosity of water,

μ_{mKs}=10^{-6} $\mathrm{m^2s^{-1}}$, using equation (4.6) with $\mu = 1/6$ ($s_\mu = 1$), we have δt=4 μs.

4.4.3. Calculation of rock permeability

The permeability of a porous medium can be calculated from the empirical Darcy's law. This well known relation states that the flow rate, J, is proportional to the force driving the fluid, the coefficient of proportionality being the permeability of the medium, K, divided by the dynamic viscosity of the fluid μ. Darcy's law can be written as

$$J = -\frac{K}{\mu}(\nabla P - \rho g), \tag{4.7}$$

where J is the flow rate per unit area of cross section (flux), K is the permeability, ∇P is the pressure drop between inlet and outlet, ρ is the fluid density, g is body force (for example gravity), and μ is the dynamic viscosity of the fluid (related to the kinematic viscosity by $\mu = \nu\rho$). By measuring (or calculating) the flux for different pressure drops (or body force values), and using equation (4.7), the permeability K can be derived. The permeability has dimensions of an area, and it is measured in units of Darcy. Converted to SI, one Darcy becomes about $9.869 \cdot 10^{-13}$ m^2, and the calculated permeability can be converted from lattice units [l.u.] to Darcys using the following expression

$$K[\text{Darcy}] = K[\text{l.u.}]\frac{\delta x^2}{0.9869} \tag{4.8}$$

where δx is the lattice spacing in microns.

4.4.4. Computational details

To estimate the single phase permeability of the sample, we impose a flow in the positive z direction of the rock with periodic boundary condition applied in the flow direction. The flow is driven only by a body force g (i.e. no pressure drop is explicitly present). This choice is due to the fact that it is difficult to explicitly introduce a pressure difference in lattice Boltzmann simulations. A pressure difference could in principle be simulated by introducing a density gradient in the fluid, but having density differences in an

incompressible fluid is not consistent. However, we would like to point out that this approach is used by several authors, and that it gives results in good agreement to the ones obtained by using a body force. A correspondence between the body force, g, and the pressure drop, ∇P, can be defined using the following equation,

$$\nabla P = \frac{(P_i - P_o)}{L} = g\rho \tag{4.9}$$

where P_i and P_o are the pressures at the inlet and outlet respectively, L is the distance between inlet and outlet, and ρ is the fluid density. Equation (4.9) can be used to compare the simulations with experiments, in which usually a pressure drop is used to drive the fluid flow. Using the unit of length and time, and the value of the kinematic viscosity, as described in subsection 4.4.2, we have

$$\nabla P = g\frac{\delta x}{\delta t^2}\,\rho \text{ [Pa/m]} \tag{4.10}$$

where ρ can be taken as the density of water at room temperature (10^3 Kg/m^3), and g is in lattice units.

Each simulation is run until the steady state is achieved (*i.e.* until the average flux is constant in time). Then the flux across each slice in the xy-plane (perpendicular to the direction of the flow) is calculated according to

$$J(z) = \frac{1}{n_x\, n_y} \sum_{x,y=1}^{n_x,n_y} v(x,y,z), \qquad z = 1, ..., n_z \tag{4.11}$$

where x, y and z denote the coordinates of a lattice site, and n_x, n_y and n_z are the system sizes (in lattice points).

For spatially periodic media, the pressure can be decomposed into a spatially periodic component $P_{per}(x)$ and a linear component ∇P_x, where ∇P is the average pressure gradient and x is the position vector. Please note that a medium which is laterally sealed is spatially periodic along the two transversal axes. For more details, see [96].

4.5. Results

4.5.1. Validation of Darcy's law

Because of mass conservation and fluid incompressibility, the flux in each longitudinal slice parallel to the main flow direction (z-axis) should be constant. However, small compressibility effects are observed in our lattice Boltzmann simulations, resulting in a variation of the values of $J(z)$ of Bentheimer sandstone with z, as shown in Figure 4.11. The compressibility

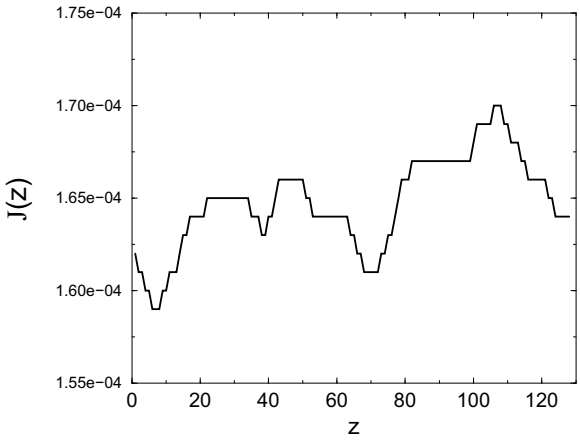


Figure 4.11.: Flux, $J(z)$ of Bentheimer sandstone, across each cross-sectional area perpendicular to the forcing—and main flow—direction (z-axis) with body force g=0.0001. The flux is expressed in lattice units.

effects depend on the fluid velocity, and hence these differences increase with increasing flow driving force. To take into account these fluctuations, we calculate the average flux, $\bar{J}$, as

$$\bar{J} = \frac{1}{n_z} \sum_{z=1}^{n_z} J(z). \tag{4.12}$$

In Figure 4.12 we plot the average flux, $\bar{J}$, as a function of applied force, g, in a 128^3 lattice sites sample of the Bentheimer.

Note that the linear behaviour (with zero intercept) predicted by Darcy's law is verified for low forcing, but deviations are observed if the forcing is too high. For high forcing, the mean fluid velocity starts to approach the

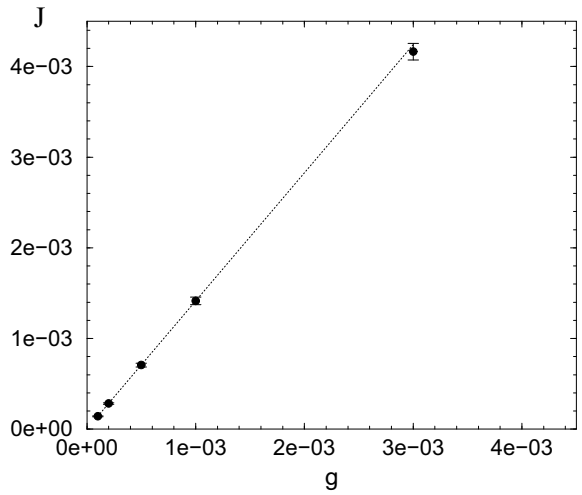


Figure 4.12.: Average flux, $\bar{J}$ of Bentheimer sandstone, as a function of applied force, g, to verify the linear behaviour predicted by Darcy's law (Eq. 4.7). The dotted line is the linear fit restricted to the data which is in the linear regime. Both flux and force are expressed in lattice units. The error bars are calculated as standard deviation over the number of slices along the force direction.

speed of sound, $c_s = \frac{1}{\sqrt{3}}$, and compressibility effects become important. The permeability can be calculated from the slope of the plotted line in the region where linear behaviour is observed.

In Figure 4.13, we show the standard deviation of the average flux $\bar{J}$, as a function of the force applied, g. Please note that the standard deviation of the flux increases almost linearly with the force applied, for forces smaller than 0.004. Above this value, the dependence is much stronger. This is probably due to the following two issues: the first is the compressibility of the fluid; the second is the use of simple bounce-back boundary conditions, see Section 4.3.

We find good agreement between the calculated values of the flux as a function of applied force and the linear behaviour predicted by Darcy's law. This is an example of how microscopic quantities, such as the velocity field at the pore scale computed using the LB method, can be related to macroscopic, phenomenological laws.

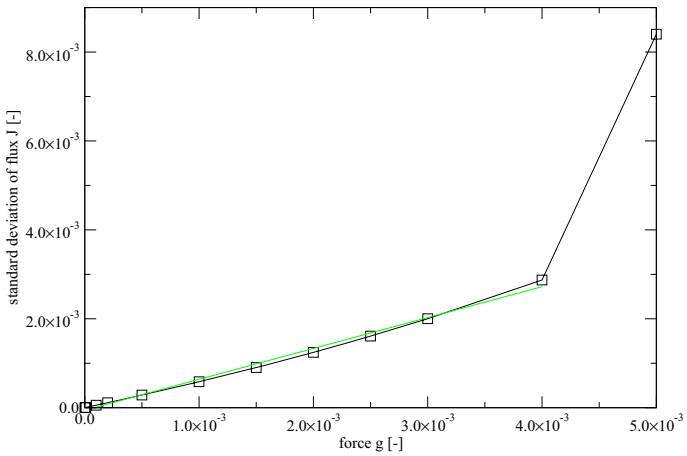


Figure 4.13.: Standard deviation of the average flux, $\bar{J}$ of Bentheimer sandstone, as a function of applied force, g.

Once the right forcing regime is found, it is sufficient to compute the flux for one forcing level and the permeability can be directly calculated using expression (4.7). For all the simulations presented in this study we have chosen a value of the force g=0.0001.

The minimum number of LB steps required to reach steady state is, in general, system size dependent. Using g=0.0001 as the driving force, and τ=1 as the BGK relaxation parameter, we have verified that 5000 steps are sufficient for equilibration, even in the largest sample (512^3) that we have considered in this work. For this number of steps, average quantities, such as the flux, have reached equilibrium. Note, however, that the local densities and velocities fields are not yet fully equilibrated after 5000 steps, and it takes about 8000 steps for the microscopic quantities to reach steady state.

4.5.2. Effect of system size

In this section, we show how the permeability of a realistic rock sample depends on the size of the sample, *i.e.* on the portion of rock used in the

simulations to calculate the flux. Please note that the influence of the sample size was discussed by Thovert *et al.* [97]. The influence of the boundary conditions was discussed in this paper and also by Gerbaux *et al.* [98].

In this study, the issue is addressed by considering samples of different sizes, and by studying the distribution of permeability as the sample size increases. The largest size we have considered is a 512^3 cube, centred in the middle of the full data set, then a 256^3 cube, also centred in the middle of the full data set. The 256^3 sample has then been divided into 8 and 64 cubes of size 128^3 and 64^3, respectively. To give an idea of the physical size of these samples, consider that a 64^3 cube has a linear size of 313 μm. The permeability has been computed in all these samples. The values of permeability, K, as a function of void fraction ϕ are reported in Figure 4.14. Observe how the data for the smaller size we have considered (64^3)

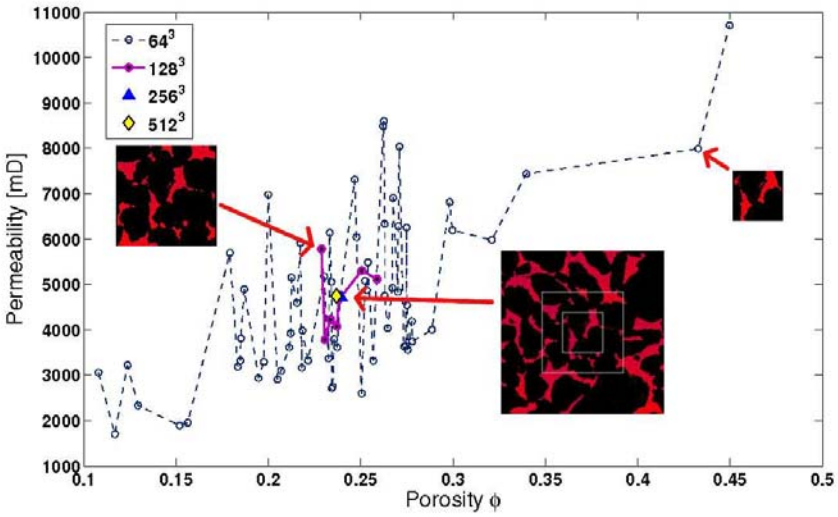


Figure 4.14.: Permeability, K (in Darcys) as a function of void fraction, ϕ, for rock samples of different size. The dotted line connecting the points for the systems of size 64^3 is only a guide to the eye.

is more spread-out, both in void fraction (ϕ) and in permeability, while, as the system size increases, the data tends to concentrate into a narrower region. On average, we found that the permeability increases almost linearly with increasing void fraction, although for the smallest samples we observed

fluctuations. It is worth noting that, the averaged permeability of all 64^3, 128^3 256^3 and 512^3 samples are 4717.66 mD, 4717.63mD, 4713.8mD and 4747.5mD respectively. This data shows that the averaged permeability of small size samples (64^3, 128^3 samples) are very close to the permeability of the large size sample (256^3, 512^3), which might mean that the permeability of large size samples can be predicted indirectly by averaging the permeability of its small sub-samples. It is also worth mentioning that compared to the Single-Relaxation-Time scheme (SRT) results [95], the degree of fluctuation of MRT results is much smaller, while the degree of linearity is higher.

To study the permeability dependency of different rocks on sample size, we calculate the permeability of different sizes of beadpack, LV60 sandpack and Portland carbonates with a similar resolution of $5\mu m/pixel$. The results are shown together in Figure 4.15. As we can see, the permeability increases with fluctuations as the porosity rises. All the permeability and porosity results converge to the value of large size samples that can be regarded as Representative Element Volume (REV)[99]. As the porosity and permeability of these rocks differs significantly from each other, we normalise the permeability and void fraction by $\kappa' = (\kappa - \kappa_{ave})/\kappa_{ave}$ and $\phi' = (\phi - \phi_{ave})/\phi_{ave}$ for the ease of comparison. The rescaled permeability as function of porosity are shown in Figure 4.16. It can be observed from this figure that from beadpack, sandpack, Bentheimer sandstone to Portland carbonates, the magnitude of fluctuation in terms of porosity and permeability increase dramatically which means the convergence rate of permeability decreases from beadpack to carbonate. We also find that the variation of porosity and corresponding permeability increases from beadpack to sandpack, Bentheimer sandstone, to Portland carbonate. These differences between different rocks indicate that the the degree of heterogeneity increases from beadpack to carbonate which needs a large sample to be a REV.

4.5.3. Effect of inlet and outlet boundary conditions

The aim of this section is to investigate the influence of boundary conditions used in LB simulations on the calculated flow field, and on the permeability in particular. Note that here by boundary conditions we mean the con-

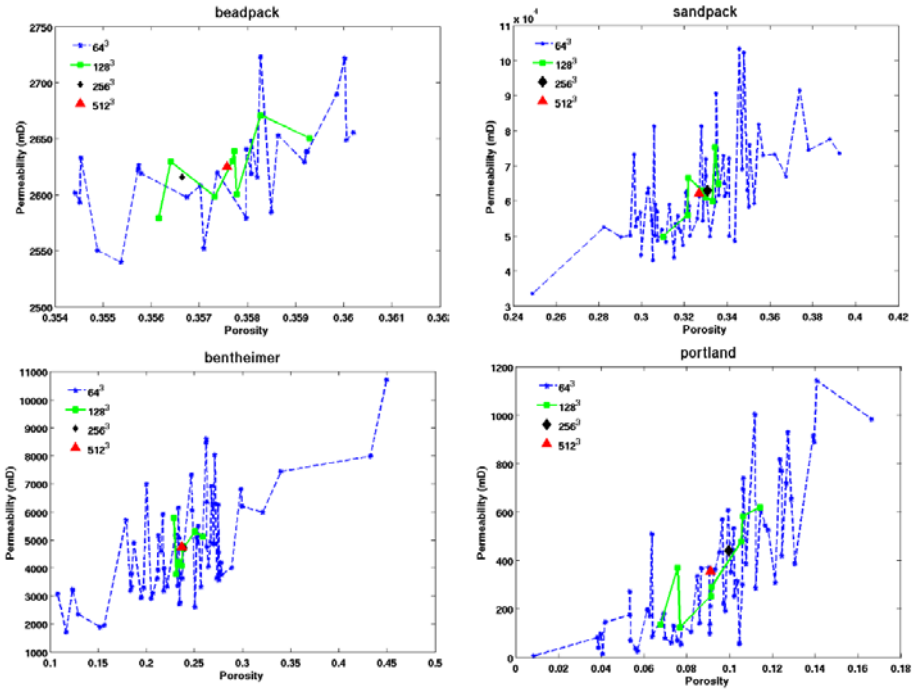


Figure 4.15.: Permeability, K (in Darcys) of different rock samples (beadpack, sandpack, Bentheimer sandstone, Portland carbonate) as a function of void fraction, ϕ, for rock samples of different size.

ditions imposed on the surface of the rock sample, and not the boundary conditions at the wall nodes, as in section 4.3. We consider, for different sample sizes, the following cases:

- Simple periodic boundary conditions (PBC) in the three spatial directions.
- Periodic boundary conditions with a fluid buffer 4 lattice sites wide in the direction of the flow (direction of the forcing, z-axis).
- Mirror boundary conditions: The system is mirrored in the direction of the flow (z-axis), and then periodic boundary conditions are used. For example, if the system size is 64^3 lattice sites, then the dimensions of the mirrored system are 64x64x128 lattice sites.
- To study the effect of the surrounding environment and rock geometry on the flow field, we also consider the case of a given rock sample when

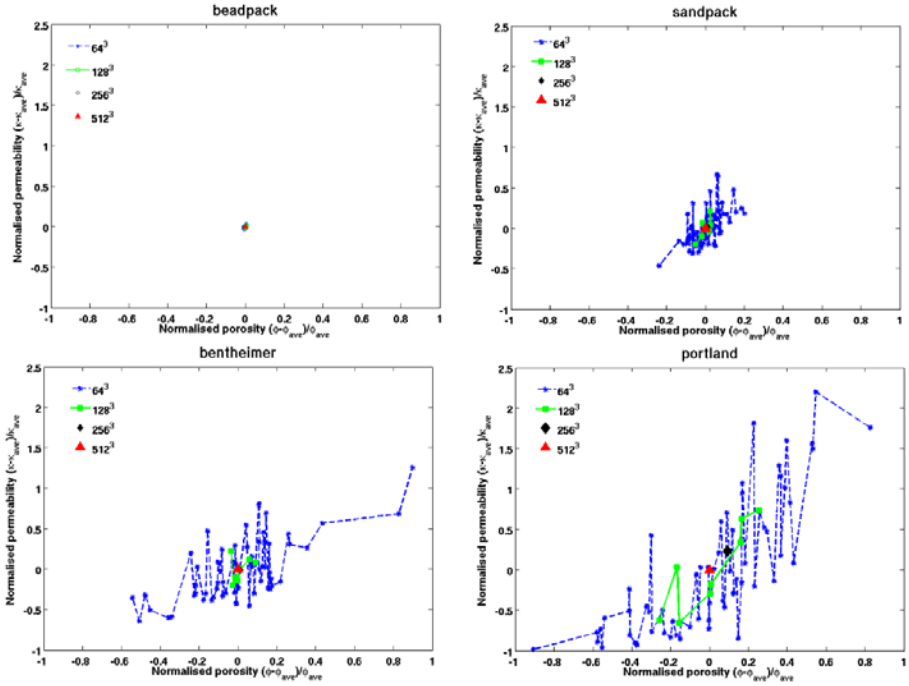
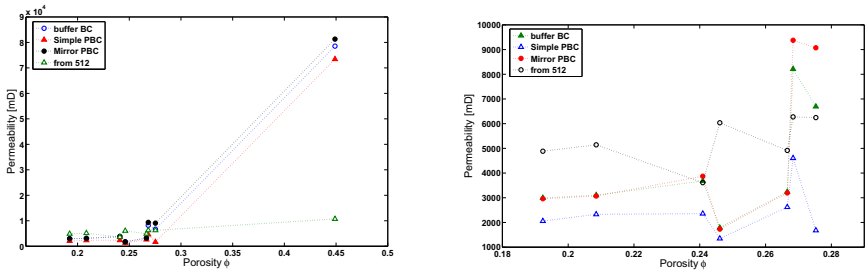


Figure 4.16.: Normalised Permeability, $\kappa' = (\kappa - \kappa_{ave})/\kappa_{ave}$ as a function of normalised void fraction, $\phi' = (\phi - \phi_{ave})/\phi_{ave}$ for beadpack (top left), sandpack (top right), Bentheimer sandstone (bottom left) and Portland carbonate (bottom right)

taken as a sub-sample of a larger system. This means that we do a single flow simulation for the larger system and calculate the fluxes for the sub-samples within the larger one.

Figure 4.17 shows the permeability of Bentheimer sandstone for eight 64^3 cubes for different boundary conditions. As a first observation, we note that the largest changes in permeability value with different boundary conditions are observed for the cube with the largest void fraction and highest permeability (last points on the right in Figure 4.17(a)). This is the cube where one large channel is present. If the cube is part of a larger portion of rock, the fluid flows through different paths and channels across the rock, and does not all flow through the large channel. On the other hand, if this cube is taken by itself, all of the fluid flows into the large channel, and the permeability of the rock is higher. From Figure 4.17(b), where we exclude the results for the cube discussed above, we observe that simple PBC give lower



(a) (b)

Figure 4.17.: Permeability, K (in Darcys), as a function of void fraction, ϕ, for 8 rock samples of size 64^3 and different boundary conditions (a). Figure (b) represents the same data as figure (a), but excluding data for the sample with largest void fraction.

permeability values with respect to the other boundary conditions. This is because, by simply replicating the data set in the direction of the flow, some channels might be artificially closed by the replica image of the rock. The addition of a buffer reopens these channels, hence resulting in a more permeable medium. Also, two sets of parallel curves in K versus ϕ can be identified in Figure 4.17(b), one corresponding to the case when the cubes are isolated, with both simple PBC and mirror PBC (open and full circles), and one for the case when the cubes are taken as sub-samples of larger samples (open triangles from a 256^3 sample and full triangles from a 512^3 sample). This indicates, not surprisingly, that the flow patterns and fluid average velocity are very much dependent on the rock geometry. Moreover, it is also worth noticing that for all boundary conditions considered, the behaviour of the permeability as a function of void fraction is not strictly monotonous. This indicates that, besides the void fraction, the connectivity between the pore space is also important in determining the flow (and hence the rock permeability).

Comparing the permeability of the 64^3 cubes (Figure 4.17(a)) with the permeability of the 128^3 cubes (Figure 4.18), for different boundary conditions, it can be noted that in the case of the larger sample the values of permeability are less dependent on boundary conditions, and on whether the sample is isolated or part of a larger portion of rock. This gives an indication of the length scales over which the rock can be considered ho-

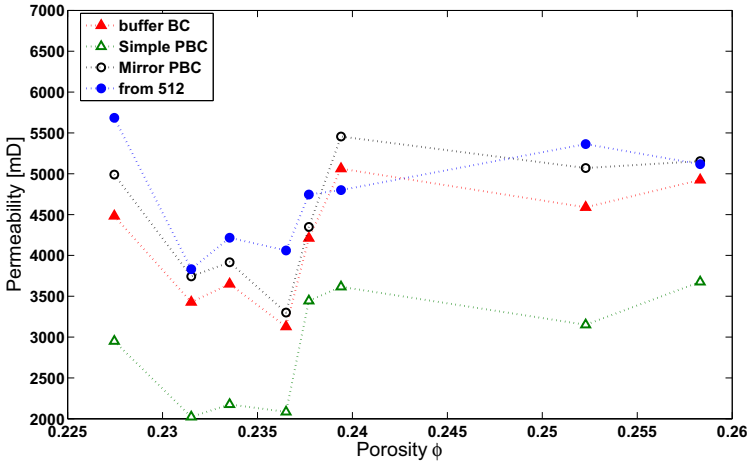
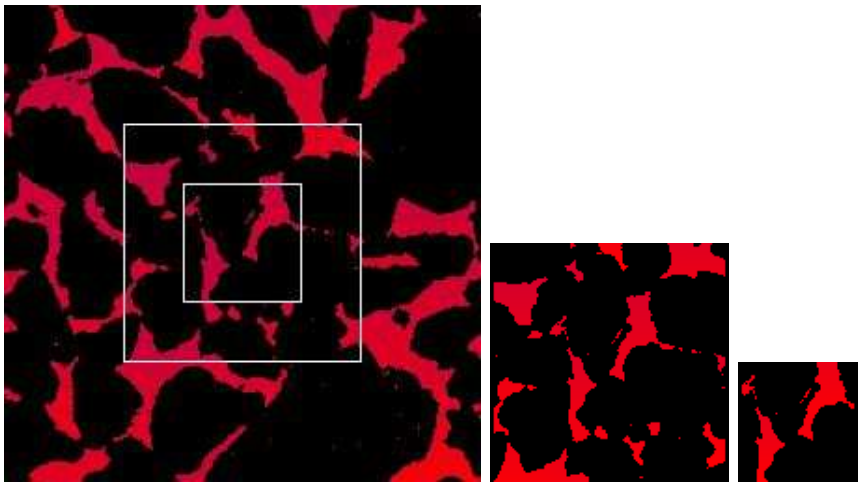


Figure 4.18.: Permeability, K (in Darcys), as a function of void fraction, ϕ, for 8 rock samples of size 128^3 and different boundary conditions.

mogeneous. Note also the narrower range of variation in void fractions and permeability in the case of the 128^3 systems, as already observed in Figure 4.14.

4.5.4. Influence of boundary conditions as a function of system size

For this study we have considered three sub-samples of increasing sizes of Bentheimer sandstone, namely: 64^3, 128^3 and 256^3 voxels, respectively, all centred at the centre of the full sample (see Figure 4.19). The permeability as a function of system size for different boundary conditions is plotted in Figure 4.20. A systematic trend in the value of the permeability can be observed; for each size, and independently of the environment, the permeability increases in going from simple PBC, to buffer, to mirror. However, as the sample size increases, the value of the permeability becomes less dependent on the type of boundary conditions used. The reason for the observed trend is that, for larger samples, surface effects on the fluid flow behaviour are smaller. For reference purposes, we also plot the permeability for the 512^3 sample with simple periodic boundary conditions (star symbol). It should also be pointed out that the data points for the smallest system



(a) 256^3 (b) 128^3 (c) 64^3

Figure 4.19.: Two-dimensional slices (perpendicular to the y-axis) of samples of Bentheimer sandstone. The slices are taken in the centre of each sample. Note that the larger samples contain the smaller one/ones. The pore space is depicted in red and the rock in black.

(64^3) show a marked dependence on the particular system chosen, since this sample size is not large enough to capture the length scales of the rock inhomogeneities, as also shown by the large variations in the permeability values of Figure 4.14.

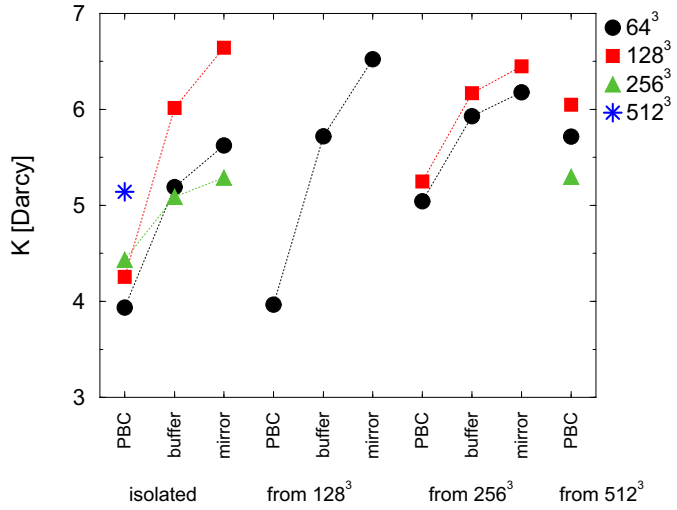


Figure 4.20.: Permeability, K, in Darcys, as a function of void fraction, ϕ, for rock samples of size 64^3, 128^3, 256^3 and 512^3, for different boundary conditions.

5. Comparison study of Multi-Component Lattice Boltzmann models [1]

We compare three different multi-component LB models for two problems: Poiseuille flow for binary fluids with viscosity contrast and capillary fingering. The numerical results are presented and compared with theoretical predictions. The thickness of the interface, limitations of the models and numerical stability are also discussed. A summary is given to show the capability and limitation of every model.

5.1. Poiseuille flow simulation for a binary immiscible fluid system with viscosity contrast

We carry out simulations of two fluids with a kinematic viscosity contrast in a channel and compare with theoretical predictions in Figure 5.3. The domain is periodic in y direction and bounded with non-slip walls. The system is initialised with substance 0 in the middle and substance 1 on both sides near the walls. An initial density value of 1 is applied to both substance 0 and 1. The force acceleration applied to each substance is 10^{-6} $l.u.$.

Figure 5.1 shows the velocity and density profile for the Shan-Chen model with viscosity ratio 4. The interactions were calculated using an eighth order isotropy scheme and the Exact Difference Method was employed to incorporate the interaction forces between components. The simulation results are generally in good agreement with the analytical solution [77], away from

[1] This chapter has been published in Computers & Mathematics with Applications 65.6 (2013): 882-890.

the interfaces. The density profile shows that the two components are immiscible except for some noise near the interfaces. It is worth mentioning that a very strong interaction is required to achieve an immiscible interface; we observed very high spurious velocities up to 10^{-1}(in lattice units) near the interfaces. Figure 5.2 shows the velocity profile for the Shan-Chen model with viscosity ratio 10. We observe that the simulation result does not match the analytical solution and some noise is found near the interface. This phenomenon has also been found by Sukop *et al.* [6]. To explore the reason for the poor agreement with the theoretical prediction, the density profile of the two substances is shown as well. In the density profile, we observe significant diffusion between the two substances. The concentration of substance 0 decreases in the middle and increases in the domains near the walls. To separate the two components, a stronger interaction is needed. However, in that case the simulation becomes unstable due to high spurious velocities caused by the strong interactions. A previous study showed good agreement with theoretical predictions for fluids having a viscosity ratio less than 6 [77]. It turns out to be difficult to calculate the flow of binary fluids with a viscosity ratio larger than 6 as numerical instabilities emerge. We observe that the thickness of the interface is significant (around 10 lattice sites). The deviation from the analytical solution is probably due to the low resolution of the channel as some very good results using fine resolution have been reported [59, 41]. However, this will increase the computational cost of the problem. In summary, we believe that the use of the Shan-Chen model, with regard to simulation of flow in real porous media geometries, is limited for the following reasons:

- The wide liquid-liquid interface is unfavourable for the simulation of flow in porous media, particularly reservoir rocks, where the average pore radius is typically around 10-30 [l.u.].
- It is difficult to increase the resolution of pore space images obtained from x-ray microtomography (XMT) due to experimental limitations.
- Even if it is possible to increase the resolution, this would substantially increase the computational burden

For comparison, we have carried out the same calculations using the Shan-Chen model with Guo's force term [58]. We observe that agreement with

the analytical solution is not as good as for the Exact Difference Method (EDM). Also, the spurious velocities observed near the interfaces are more significant for the Guo model compared to EDM.

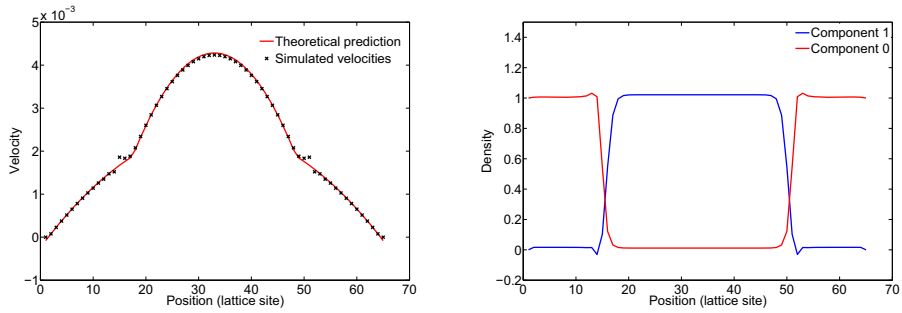


Figure 5.1.: Left: Simulated (black) and analytical (red) velocity profiles for Poiseuille flow of binary fluids with viscosity contrast for the Shan-Chen model; Right: Simulated density of the Shan-Chen model. The viscosity ratio between the two substances is 4. The initial densities are both set as 1, surface tension as 0.01 (l.u.)

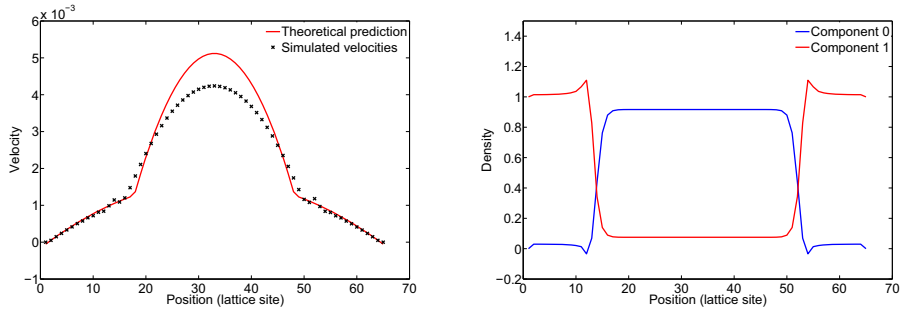


Figure 5.2.: Simulated velocity (left) and density (right) profiles for the Shan-Chen model. The viscosity ratio between the two substances is 10. The initial densities are both set as 1

The simulation results of both the Free Energy Model and the Colour Gradient Model are shown in Fig. 5.3 and give excellent agreement with the analytical solution. Two immiscible fluids are found constrained in their respective areas and no unexpected diffusion is discovered. We note that the interface thickness of the Free Energy Model is around 6 lattice units (l.u.) and the interface location is found at $x = 38$ instead of their initial position of $x = 35$, so that an interfacial movement of 3 lattice units is discovered

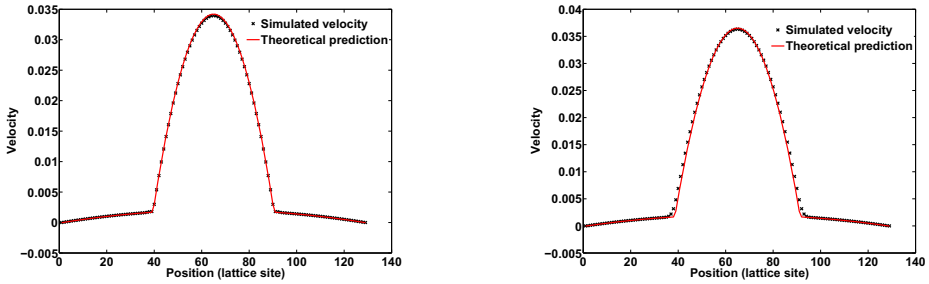


Figure 5.3.: Simulated velocity profiles for the Colour Gradient (left) and Free Energy Model (right) with viscosity ratio 100, surface tension 0.01 (l.u.)

from the result. The reason for this interface shifting is not clear and needs further investigation. The interface thickness of the Colour Gradient Model is around 4 l.u., which is smaller than that of the Free Energy Model. An interfacial shift is observed in the colour gradient model of around 1.5 l.u., which is significantly smaller than the free energy model. The recolouring algorithm separates the two immiscible fluids very well and gives nearly 0 diffusivity as expected.

The same trend regarding the variation of interfacial thickness for different multi-component models is observed for different viscosity ratios and interfacial tensions. As an example, Poiseuille flow simulations were performed with a viscosity ratio of 10 and a surface tension of 0.005. The Shan-Chen model generates a thickness of 6 l.u., while the Free Energy model and the Colour Gradient model give 4 and 2-3 lattice units respectively.

5.2. Capillary fingering simulation

Capillary fingering is a well known hydrodynamic instability which occurs in various displacements during oil/gas production. Here we consider one fluid which is displaced by a second one, having a different viscosity, along a channel with non-slip walls. A growing finger of the driving fluid will be produced if the capillary number Ca is big enough.

$$Ca = \frac{u_t \nu_2}{\sigma}, \tag{5.1}$$

where u_t is the velocity of the tip of the finger and ν_2 is the viscosity of driving fluid. σ is the surface tension.

Chin *et al.* [77] studied viscous fingering using the Shan-Chen model. Although fingering structures were observed in their simulations, the development of a finger is clearer in simulations at higher surface tensions, which is counterintuitive. We normally expect that increased surface tension is not favourable to the generation of fingers. It is not clear whether these structures were produced due to viscous fingering or due to other effects [77]. Numerical instability and high diffusivity were also found in the simulation when the viscosity ratio is larger than 7.

Here we use the Colour Gradient Model and the Free Energy Model to study viscous fingering. A domain with grid size 512x32 is used in the simulation. The first half of the domain contains substance 0 and the rest is occupied by phase 1. Periodic boundary conditions are used in x direction and bounded with non-slip boundaries. A pressure gradient is imposed by applying a body force in the x direction. Figure 5.4 shows the evolution of fingers for binary fluids with a viscosity ratio of 20 and a tip velocity of 0.01, simulated by the Free Energy Model. From top to bottom, these are finger evolutions for surface tension values of 0.06780, 0.03890, 0.01985 and 0.00992 (l.u.) respectively. No finger will be produced if the surface tension is high. When the surface tension decreases, fingers are observed. However, with decreasing surface tension, less stable interfaces are produced. Figure 5.5 shows the evolution of the fingers of the same binary fluids with a larger tip velocity of 0.05, calculated by the Colour Gradient Model. In all cases, stable fingers are observed. Halpern and Gaver [100] studied the fingering phenomenon in a channel, by measuring the width of the fingers as a function of the capillary number Ca. We show our results obtained from the Free Energy and Colour Gradient Models in comparison with the results from Halpern and Gaver [100] in Figure 5.6.

Good agreement is achieved, although some small discrepancies are found. These discrepancies might have arisen from the boundary conditions. The Free Energy Model gives better agreement for low capillary number simulations and the Colour Gradient Models did better in high capillary number

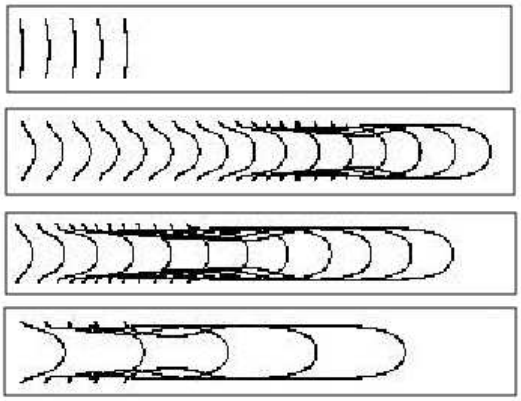


Figure 5.4.: From top to bottom: Fingering evolution for the Free Energy Model with surface tensions of 0.06780, 0.03890, 0.01985, 0.00992 [l.u.] at a time interval of 1000 time steps. Viscosity ratio is 10; the tip velocity is 0.01. The number of snapshots is different due to the different time required for the evolution of fingers. For large surface tension, the fingers develop very slowly. For low surface tension, on the other hand, we find very long fingers (developing very quickly).

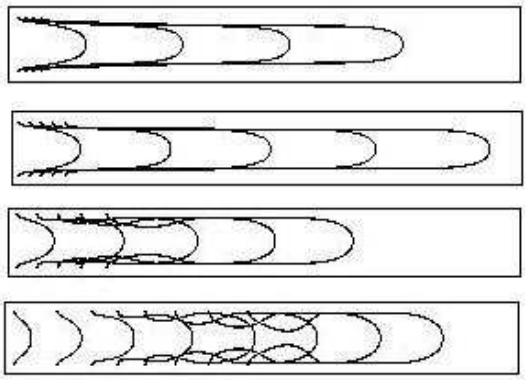


Figure 5.5.: From top to bottom: Fingering evolution for the Colour Gradient Model with surface tensions of 0.03890, 0.01985, 0.00992, 0.00496 at a time interval of 1000 time steps. Viscosity ratio is 20; the tip velocity is 0.05

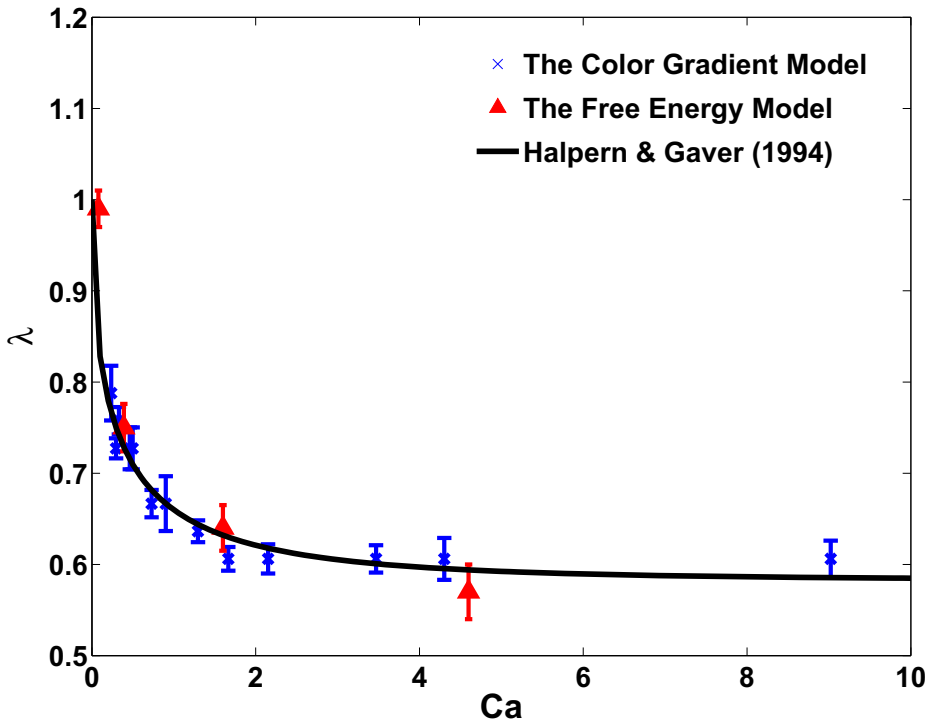


Figure 5.6.: Finger width as a function of Capillary number. Our simulation results from the Free Energy and Colour Gradient Model model are shown as triangles and stars respectively, in comparison with the results from Halpern shown as a solid black line

simulations. These numerical examples show that fingering phenomena can be captured properly by both the Free Energy Model and the Colour Gradient Model. It is worth mentioning that the numerical stability for both the Free Energy Model and the Colour Gradient Model is similar. The maximum viscosity ratio for dynamic interface simulations is around 20. To achieve a higher viscosity ratio, an optimised perturbation term based on the MRT collision scheme needs to be used. In this case, the viscosity ratio can be increased to 1000 with reasonable numerical stability [65].

5.3. Summary

The advantages and limitations of the Shan-Chen model, the Colour Gradient Model and the Free Energy Model have been investigated and directly compared with regard to seven different aspects:

1. The maximum density ratio of binary fluids that can be achieved
2. The maximum viscosity ratio of binary fluids that can be achieved
3. Interfacial thickness in Poiseuille flow
4. Interface shifting distance in Poiseuille flow
5. Ability to describe capillary fingering

It has been observed that the Shan-Chen model is capable of simulating high density ratio fluids [59, 101]. As such, it is a promising tool for liquid-gas systems, but gives relatively low numerical stability and wide interfaces for multi-component immiscible systems. For this reason, this model may not be the optimal solution for simulation of immiscible flows [102]. Both the colour gradient and the free energy model are capable of simulating fluids with significant viscosity contrast, and recovering the analytical solutions of Poiseuille flow and fingering simulations. In our study, we found that the maximum viscosity ratio of these two models depends on the type of problem. If the interface is static, fluids with a viscosity ratio up to 120 can be simulated. However, for dynamic interface simulations, this value decreases to 20. We conclude that the free energy and colour gradient models seem appropriate to simulate the flow of binary fluids with high viscosity contrast and high numerical stability. This is of great importance for the study of immiscible flow in porous media, in particular for CO2 storage and Enhanced Oil Recovery (EOR) operations.

Table 5.1.: Quantitative comparison of three multi-component LB models

	The Shan-Chen Pseudo Potential Model	The Free Energy Model	The Colour Gradient Model
Maximum density ratio	up to 10^9 [41]	1	1
Maximum viscosity ratio	5	120 (static interfaces) 20 (dynamic interfaces)	120 (static interfaces) 20 (dynamic interfaces)
Interface thickness (Poiseuille flow simulation)	10 lu	5 lu	2-3 lu
Interface Shifting distance (Poiseuille flow)	10 lu	5 lu	2-3 lu
Ability of simulating capillary fingering	No	Yes	Yes
Inherent Parallelism	Low	High	High
Diffusion	High diffusivity interface	Diffusivity can be controlled by a single parameter	Complete immiscible, no diffusion on the surface

6. Multi-Component flow in porous media

6.1. Introduction

Multi-component flows are of great importance in many engineering applications, including petroleum, biochemical and chemical engineering. Several conventional CFD techniques, including the volume of fluid (VOF) and level-set methods, have been used to study multi-component flow. Interfacial dynamics at large scales can be captured by these techniques, but information of small scale interfaces is often missing [103]. The lattice Boltzmann method (LBM) is an alternative solution for simulations of complex flow due to its statistical physics background, easy implementation, strength of dealing with complex geometries and inherent parallelism [104]. In this chapter, we will consider lattice Boltzmann simulations to study the flow of water, supercritical CO_2 and oil in porous media, for CO_2 storage and Enhanced Oil Recovery (EOR) operations.

6.2. Verifications for the multi-component lattice Boltzmann method

To validate the LB multi-component model, we calculate five test problems, for which the analytical solution can be calculated. Flow in a capillary tube with different cross sections, and the relative permeability of an idealised geometry are investigated and compared with analytical solutions.

6.2.1. Capillary Pressure

A tube with a rectangular cross section shown in Figure (6.1) is used to measure the capillary pressure of a strongly non-wetting bubble of different

sizes. The setup of the simulation and the equilibrium state of the bubble are shown in Figure (6.1). The analytical solution of the equilibrium capillary pressure relation was given by Mason and Morrow [105]:

$$P_c = \frac{\sigma(1 + 2\sqrt{\pi G})cos\theta}{r} F(\theta, G) \tag{6.1}$$

where r is the inscribed radius, $G = A/O^2$, A is the cross-sectional area, O is the perimeter length. $F(\theta, G)$ is close to one for small contact angles. The tube size is set as 60x30x30, the contact angle of the wetting phase is 0 degrees. A surface tension $\sigma = 0.01$ ([lattice units]) was employed. We use a fixed geometry with different initial non-wetting phase volumes to achieve a different inscribed radius in Equation (6.1). The capillary pressure as a function of inverse of the inscribed radius is shown in Figure (6.2). As can be seen, the LB computed capillary pressure is in good agreement with the theoretical predictions.

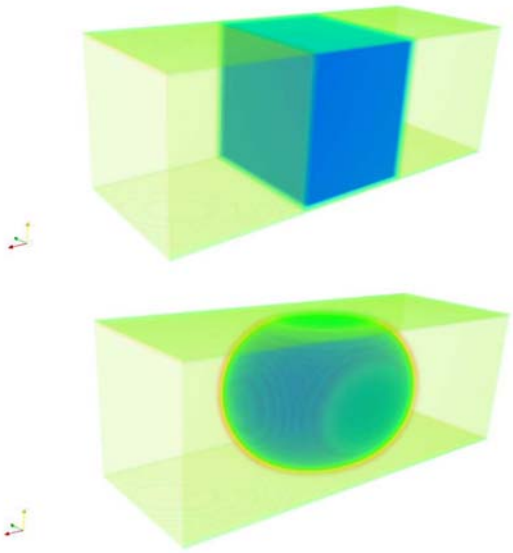


Figure 6.1.: Capillary tube with rectangular cross section: initial setup for non-wetting phase coloured blue (top); equilibrium state (bottom)

To verify the algorithm for simulating the dynamic entry of non-wetting phase in the tube, two capillary tubes are set up. Figure (6.3) shows the geometry of the capillary tubes. We study the pore space having a rectan-

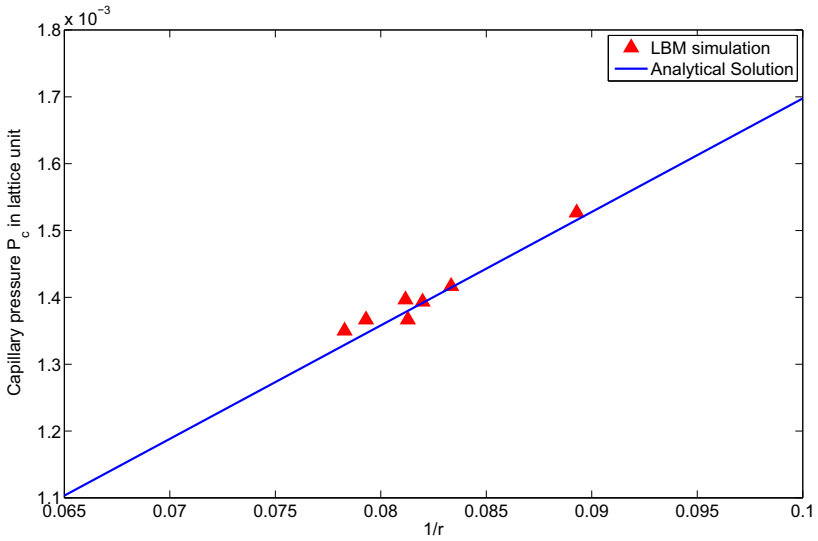


Figure 6.2.: Capillary pressure VS inscribed radius, the analytical solution is plotted using a solid line

gular cross section (left in Figure 6.3) with two resolutions: 12x6 and 24x12. A non-wetting phase reservoir is located near the inlet of the tube in the flow direction where a porous plate prevents the non-wetting phase flowing out through the outlet [45](the geometry of the porous plate is shown in Figure 6.3). This configuration enables us to measure the capillary pressure of small pores. Without the porous plate, the high pressure cannot be held as the non-wetting phase will flow out through the outlet freely.

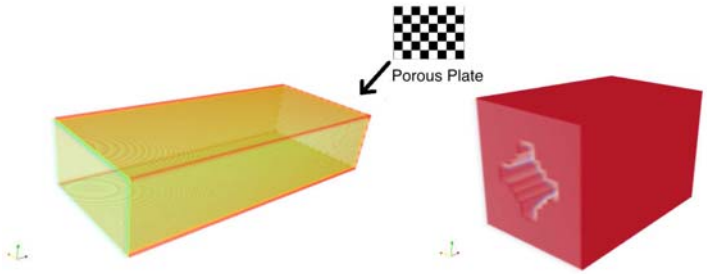


Figure 6.3.: A capillary tube with rectangular and uniform grain cross section

To estimate the capillary pressure as a function of wetting phase satura-

tion in the primary drainage, we started from zero capillary pressure, which was achived by setting the inlet and outlet with the same density/pressure. Then, the density/pressure of the outlet was incrementally decreased to achieve the desired capillary pressure. When the saturation reached the steady state, which implies that Equation (6.2) is satisfied for at least 500 time steps, the wetting saturation was recorded. The surface tension was set as 0.01 [l.u.], and an equal viscosity for wetting and non-wetting phase of 0.02 was used in the calculation. The contact angle for the wetting phase was 0 degrees. The mean curvature for the case of zero contact angle is given by Bear [99] as $R = 1/(2/L_1 + 2/L_2)$, in which L_1 and L_2 are the length and width of the rectangular cross section respectively.

$$|\frac{S_w(t) - S_w(t-50)}{S_w(t-50)}| <= 10^{-6} \tag{6.2}$$

The second type of capillary tube has four grains of uniform radius and is shown at the right in Figure (6.3). The inscribed radius of the pore space is known as $R = (\sqrt{2} - 1)R_{grain}$. The tube size is 50x20x20 while the grain radius is 12. The third type of test geometry is an idealised porous medium which packs eight spheres with equal radius (Figure 6.4). The equation for the inscribed radius is the same for the second type of capillary tube. The size of the sphere pack is 50x20x20 with a sphere radius of 9.

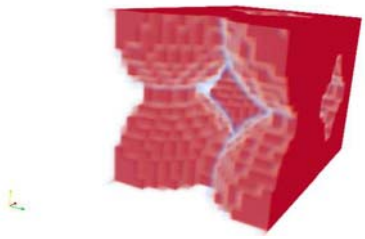


Figure 6.4.: Idealised porous medium

The entry pressure can be computed by using the Young-Laplace equation. The curve of non-dimensional pressure P_cR/γ versus wetting phase saturation for primary drainage of the three test geometries, obtained from LB simulations, are shown in Figures (6.5, 6.6, 6.7).

As can be seen, the LB simulation results for entry pressure are in excellent agreement with analytical solutions. The simulated saturation increases

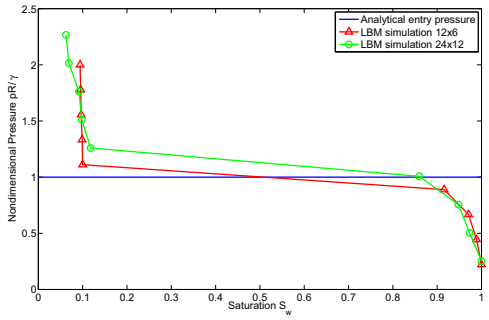


Figure 6.5.: Capillary pressure curve for primary drainage process in a capillary tube with rectangular cross section, considering two levels of resolution: 12x6 and 24x12.

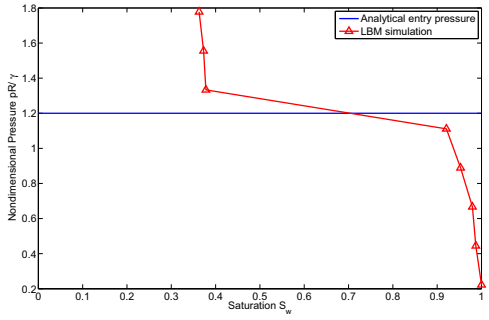


Figure 6.6.: Capillary pressure curve for primary drainage process in a capillary tube with four grains of equal radius.

slowly with the increase of pressure. When the pressure is increased beyond the entry pressure, the saturation increases dramatically to a high value and the non-wetting phase was found to enter the pore/throat. This entry value was found to match the analytical prediction by the Young-Laplace law (blue line in figures). According to the results, the discretization does not affect the simulation results significantly, all the cases predict the entry pressure accurately. Snapshots of the primary drainage in an idealised porous medium are shown in Figure (6.8).

6.2.2. Relative Permeability

In multi-phase flow in porous media, Darcy's law (see equation 4.7) can be extended to describe the ratio of permeability of each phase to the absolute

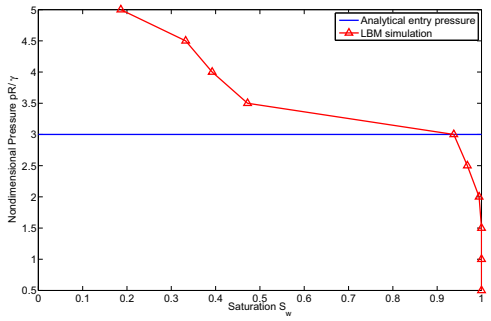


Figure 6.7.: Capillary pressure curve for primary drainage process in an idealised porous medium, which packs eight spheres of equal radius.

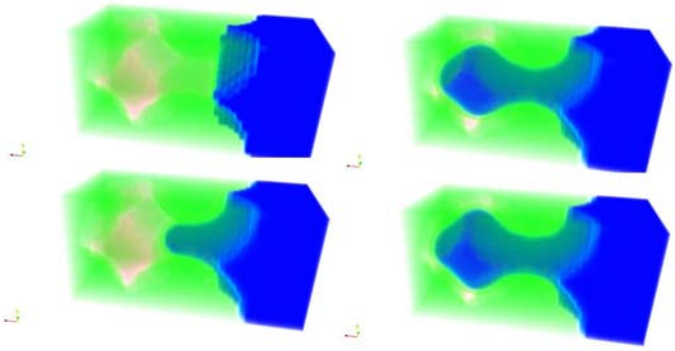


Figure 6.8.: Snapshots of primary drainage in idealised porous medium, from top to bottom, left to right.

permeability which can be used as a dimensionless measure of the effective permeability of each phase. The extension of Darcy's law can be written as:

$$\kappa_\alpha(S_\alpha) = -\frac{\mu_\alpha \mathbf{v}_\alpha}{\Delta P_\alpha \kappa} \tag{6.3}$$

where the subscript α refers to the fluid phase α, S_α is the saturation, κ is the absolute permeability, and the terms κ_α, $\mathbf{v}_\alpha$ and ΔP are the permeability of phase α, volumetric averaged fluid velocity and pressure gradient respectively. In the simulation, all the terms are in lattice units.

In multi-phase flow in porous media, the dimensionless capillary number

Ca is used to measure the relative effect of the viscous and capillary forces, and is defined as:

$$Ca = \frac{\mu U}{\sigma} \tag{6.4}$$

where μ refers to the viscosity, U is the Darcy velocity[1], and σ is the surface tension. Several fundamental studies on the relative permeability have been carried out by Marle [106], Avraan and Payatakes [107]. They found that, if the capillary force dominates the flow, the fluids are hydrodynamically decoupled. In that case, the extension of Darcy's law (Equation 6.3) is valid and the relative permeability only depends on the geometry of the porous media, wettability and phase saturation. If the viscous effect increases, the fluids are coupled and this leads to the mobilisation of trapped fluids [108]. The relative permeability eventually shifts towards a linear dependency with saturation in the near miscible region [109].

As model systems, immiscible two-phase flow through two channels (2D) and a cylindrical channel (3D) were studied using the LB multi-phase model described in Chapter 3.10.4. The predicted relative permeability is compared with analytical solutions. The geometry of the 2D channel and the setup of the simulation is shown in Figure 6.9. Periodic boundary conditions are applied in the flow direction whereas a half-way bounce back boundary condition is used on the upper and lower walls. A body force is applied to drive both phases and to avoid the pressure gradient which normally leads to capillary end effects. Viscosity ratios of the non-wetting and wetting phase $M = \frac{\mu_{nw}}{\mu_w}$ of 1, 0.1 and 0.05 are studied. A mesh of 60x50 is used for all simulations. The analytical solution can be derived from the Poisseuille flow velocity profile and is given as [110]:

$$\kappa_w = 0.5 S_w^2 (3 - S_w) \tag{6.5}$$

$$\kappa_{nw} = S_{nw}[1.5M + S_{nw}^2(1 - 1.5M)] \tag{6.6}$$

The simulation results for different viscosity ratios together with the analytical solution are shown in Figure 6.10, 6.11 and 6.12. As can be seen,

[1]Darcy velocity $U = Q/A$, where Q is the flux over a cross-section of area A.

the predicted relative permeability from LB agrees well with analytical solutions for all viscosity ratios. This is because the flow in a channel with two parallel walls is a second order problem, the LB method is a second order method and the half-way bounce back boundary condition is of second order accuracy, therefore the LB simulation can accurately predict the relative permeability for this problem.

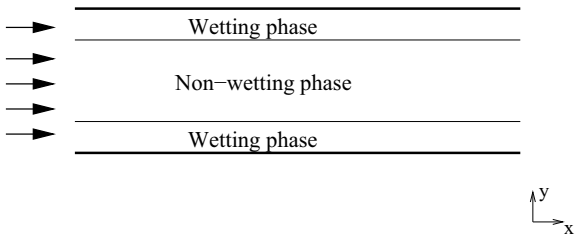


Figure 6.9.: Geometry and setup of immiscible two-phase flow in a 2D channel

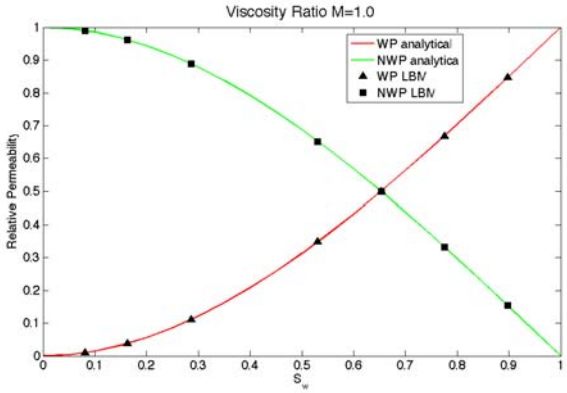


Figure 6.10.: Relative permeability of a binary fluid in a 2D channel, LBM predictions and analytical solution, viscositiy ratio $M = 1.0$

Having validated 2D model systems, two-phase flow through a 3D cylindrical channel was calculated. The geometry and setup are shown in Figure 6.13. The mesh resolution is 100x50x50. The relative permeability for viscosity ratios $M = 1$ and $M = 0.2$ is calculated and compared with the analytical solution given by Goldsmith and Mason (Equation 6.8)[111] as shown in Figure 6.14

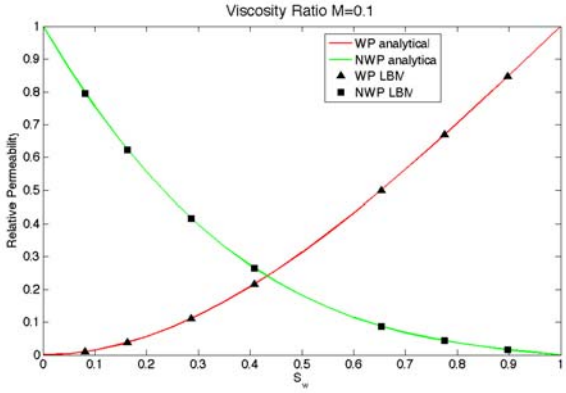


Figure 6.11.: Relative permeability of binary fluid in a 2D channel, LBM predictions and analytical solution, viscosity ratio $M = 0.1$

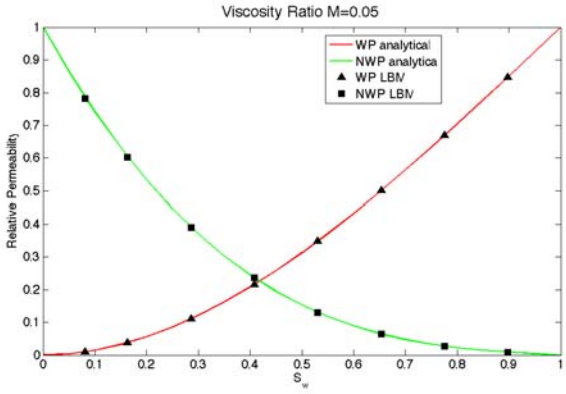


Figure 6.12.: Relative permeability of binary fluid in a 2D channel, LBM predictions and analytical solution, viscosity ratio $M = 0.05$

$$\kappa_w = S_w^2 \tag{6.7}$$

$$\kappa_{nw} = 2S_{nw}M + S_{nw}^2(1 - 2M) \tag{6.8}$$

As can be seen in Figure 6.14, the predicted relative permeability confirms the validity of the LB implementation for binary fluids with equal viscosity. However, for $M = 0.2$, the relative permeability of the non-wetting phase for low saturation is slightly lower than theoretical predictions. This may be due to the finite size resolution of the cylindrical channel walls, which is not observed in the 2D flat channel test (Figure 6.10, 6.11 and 6.12)

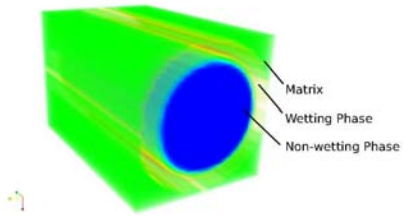


Figure 6.13.: Geometry and setup of immiscible two-phase flow in a 3D cylindrical channel

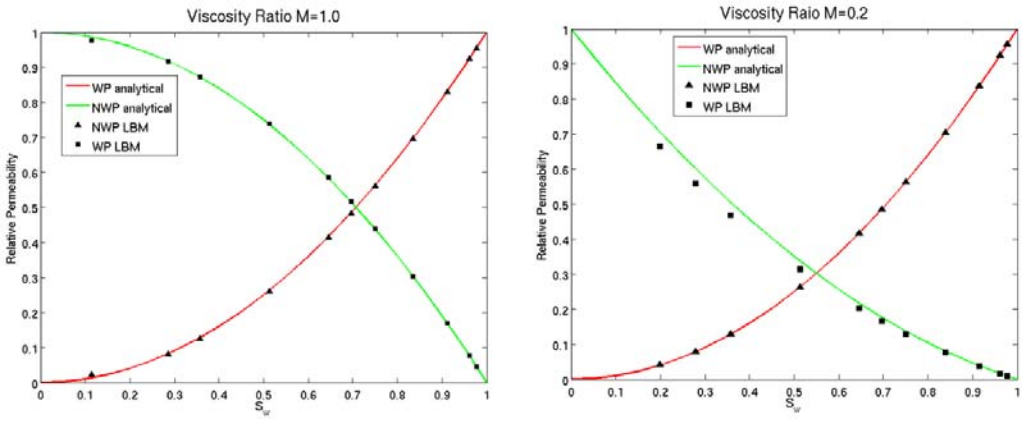


Figure 6.14.: Relative permeability as a function of wetting saturation S_w with different viscosity ratios (Left: equal viscosity M=1.0, Right: viscosity ratio 5, M=0.2)

6.3. Experimental and LBM study of multi-phase flow in micro-models

In this chapter, fluid displacement in single pore junction micro-models with equal and unequal arms is studied. We also investigate a design based on an actual rock section: Berea sandstone. In the Pore Network Models (PNMs), which are widely used to predict the displacement of binary fluid in porous media, the drainage and imbibition is calculated using the Young-Laplace Law [112, 113]:

$$P_c = 2\gamma cos\theta(\frac{1}{h} + \frac{1}{w}) \tag{6.9}$$

where, P_c is the capillary pressure, γ is the interfacial tension, θ the contact angle and h and w the height and width of the channel. In PNM, a

capillary filling rule was used to study the drainage and imbibition [112]. For primary drainage, where the non-wetting phase is injected into a medium completely saturated with wetting phase, the non-wetting phase should select the channel with the lowest capillary entry pressure first. For imbibition, where the wetting phase is injected into a porous medium initially filled with non-wetting phase, the wetting phase will enter the narrowest channel with the highest capillary pressure.

This work aims to find a comprehensive solution to the question of whether the capillary pressure filling rules used by network modelling are actually observed in single junction and imbibition experiments and direct simulations. Experiments and LBM simulations were performed to study the spontaneous imbibition of decane in air-saturated micro-models. We found that, for primary drainage, the capillary filling rule holds; however, for the case of imbibition, the capillary filling rules may not be valid. The local geometry of the network model junction plays an important role. Moreover, the experimental data was compared to computer simulations. The simulations results were in excellent agreement with experimental data in both imbibition and primary drainage, which confirmed the ability of the lattice Boltzmann method to study complex flow at the pore scales.

6.3.1. Methodology

Three micro-models of 2.5 dimensionality (they have a finite depth of $75\mu m$) were designed to study fluid flow in single junctions, with increasing complexity, ranging from a simple square pore with equal, unequal arms, to a pore structure representing actual Berea sandstone. The patterns with typical scales are shown in Figure 6.15. The oil-wetting micro-models were fabricated in PMMA by Epigem Ltd. A Zeiss inverted microscope (AXIP Observer A1.M) was used to capture still images and the displacement process was captured by a high speed video microscope (FastCam MC2.1, Photron) with a maximum frame rate of 10,000 frames per second (fps) and a resolution of 512x512 up to a recording setting of 2,000fps. (The experiments in this chapter were carried out by Emily Chapman from Department of Chemical Engineering, Imperial College London)

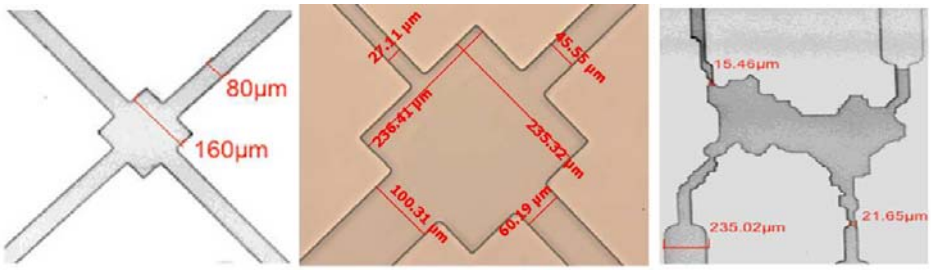


Figure 6.15.: Single Junction Designs, images were taken by Zeiss microscope

The colour gradient multi-component lattice Boltzmann method described in Chapter 3.10.3 was used to calculate the flow in micro-models. For the boundary conditions, we used the simple half-way bounce back scheme which offers second order accuracy and conserves the mass of all components. To simulate the spontaneous imbibitions, a modification on the geometry was performed. We took the micro-model with a squared pore in the centre as an example to show the initial configuration of the simulation. This configuration is shown in Figure 6.16, the green represents the solid, the colours blue and red represent air and decane respectively. A big reservoir which connects the top three channels was built for both decane and air at the bottom; periodic boundary conditions were applied in X,Y and Z direction.

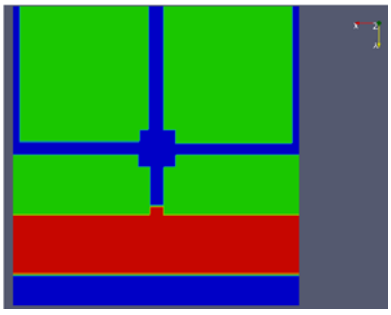


Figure 6.16.: The initial configuration for multi-component simulation

For the main displacement process of two-component flow in micro-models, the Reynolds number is very small (< 0.01), so that the inertial effects are

negligible. The density of decane and air in the simulation was set as equal. Another important dimensionless parameter is the capillary number which represents the relative effect of viscous force versus surface tension on the interfaces:

$$Ca = \frac{u_w \mu_w}{\sigma} \tag{6.10}$$

where u_w and μ_w are the Darcy velocity and dynamic viscosities of the wetting phase. σ is the value of surface tension and was set as 10^{-2} in all simulations. We use a low viscosity ratio of 10 in the simulation rather than the real decane/air system viscosity ratio of 50. This is justified because the typical capillary number for the flow in micro-models is of order of 10^{-5} which means the capillary force dominates the displacement process. Therefore the ratio of 10 or even lower is sufficient to reproduce the main physical process [66]. The contact angle is set to 30°. The implementation of contact angle can be found in [36].

The initial distribution of decane and air was shown in Figure 6.16. Because the micro-model is oil-wet, the decane will spontaneously imbibe into the model due to the capillary pressure. A video of the process can be captured by the high speed camera for the comparison with the LB simulations. It is worth noting that, all the simulations were performed in 3D in order to recover the experiments.

6.3.2. Simulation results and comparisons with experimental data

Primary Drainage

Firstly, the primary drainage process was studied. The decane was forced to enter the single square pore model with unequal arms from the top-right channel. A body force of 10^{-5}, which is equivalent to pressure gradient, was imposed along the entry channel direction to mimic the pressure in the experiments [73]. As can be seen in Figure 6.17, in both experimental and simulation results, although the entry fluid reached the top-left channel having the smallest width, the filling fluid did not enter due to the high capillary pressure. The filling fluid chose the furthest channel, which has

the largest width, to fill first. The capillary filling law holds in this case: the channel with lowest capillary pressure will be filled first. The LB simulation predicted the primary drainage in the micro-models accurately, quite good agreements were achieved. According to Figure 6.17, we can see that, not only the filling order was recovered properly, but also that the curvature of interface and details of fluid distribution matched the experimental results well. Two bubbles caused by snap-off in the left-bottom and right bottom corner of pore were captured by the LB simulation. However, the volume of bubbles in the simulation is slightly bigger than that in experiments; this deviation might be caused by the neglect of density contrast and compressibility of the air/decane system. Generally, we think the agreements is quite satisfactory.

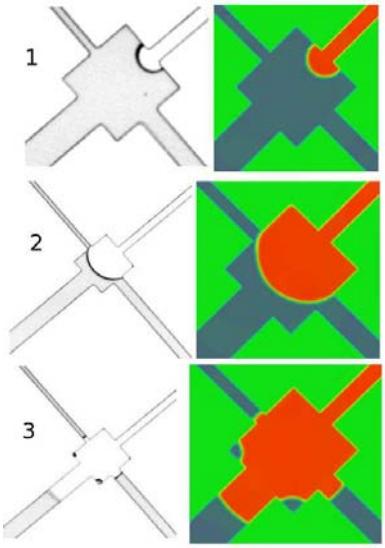


Figure 6.17.: Snapshots of primary drainage of decane in a single junction micro-model, experiments results and the lattice Boltzmann simulations. The experimental data and simulation results are shown together for easier comparison. The left black-white snapshots are experimental data, the colour snapshots were obtained from the LB simulation.

Spontaneous Imbibition

In both micro-models, the wetting phase (decane) has spontaneously imbibed into the model from the top right corner (the single junction micro-model) and bottom right corner (the Berea micro-model). According to the capillary entry pressure rules used by network modelling, the fluid should enter the smallest channel which has the highest capillary pressure first; however, in our study, this is not the case. The results for the single junction micro-model and the Berea micro-model obtained by experiments and the parallel lattice Boltzmann simulation are shown in Figure 6.18 and Figure 6.20 respectively. The snapshots are taken from the top of the micro-models (Z direction). The black-white snapshots are experimental data, the light grey and dark grey represent decane and air respectively, the interface is shown in dark black. The lattice Boltzmann simulations are shown with colours: the red, blue and green represent decane, air and PMMA base respectively.

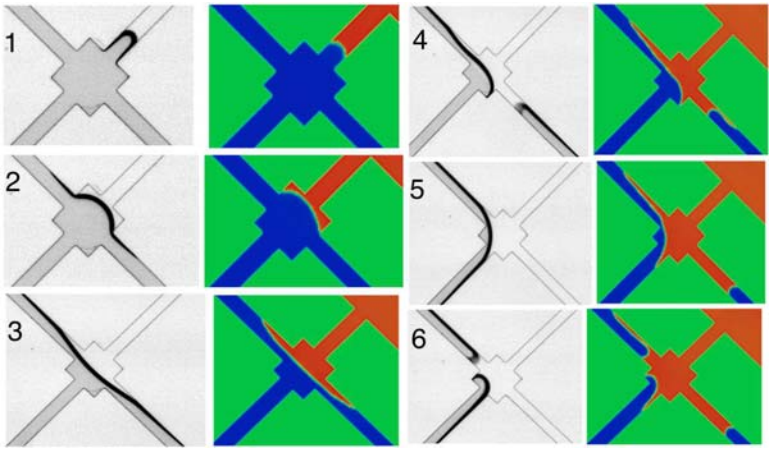


Figure 6.18.: Snapshots of spontaneous imbibition of decane in a single junction micro-model with equal arms. The left black-white snapshots are experimental data, the colour snapshots are obtained from the LB simulation.

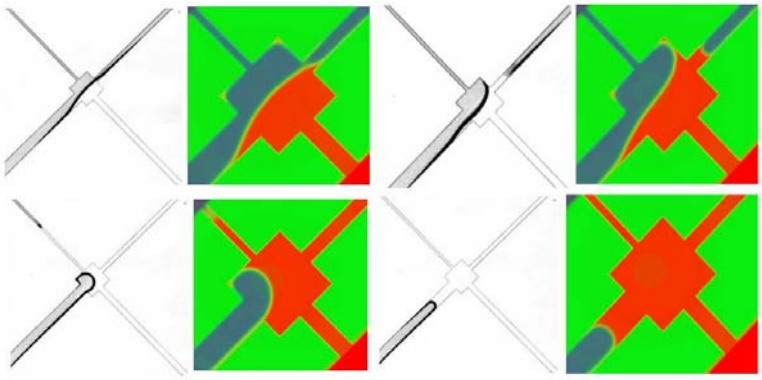


Figure 6.19.: Sequential snapshots of spontaneous imbibition of decane in a single junction micro-model with unequal arms.

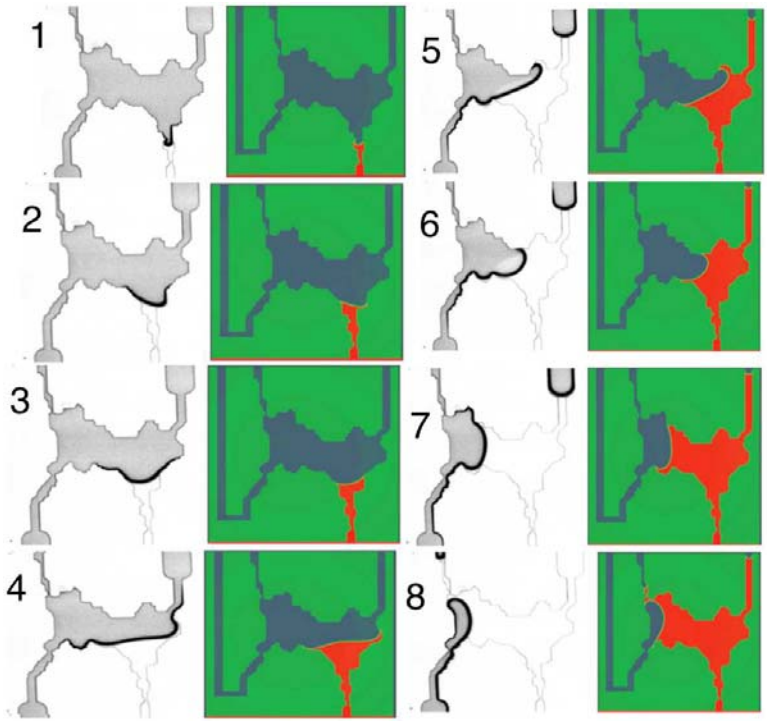


Figure 6.20.: Snapshots of spontaneous imbibition of decane in a Berea sandstone micro-model

In Figure (6.18), the decane was imbibed into the bottom right corner

first, then top left and bottom left corner which is not consistent with the filling rule used in network modelling (Figure 6.18.3, 6.18.4). In network modelling, the fluid should imbibe into all the other channels simultaneously. We think this inconsistency was caused by the asymmetry of the experimental micro-model. As a result, we slightly revised the geometry for the LB simulation to break the symmetry. The width of the bottom right channel is one lattice smaller than all the other channels, and the bottom left corner of the squared pore was one lattice smaller than the other corners. It should be noted that after the modification on the geometry, the width of the top left and bottom left channel is the same. According to the filling rules of network modelling the decane should enter two channels simultaneously; however, both the experiment and simulation showed that the decane imbibes into the top left channel first as a result of interface contact with the top left corner in advance (Figure 6.18.5,6.18.6). According to Figure 6.19, the capillary filling law broke again. The decane fills first not the narrowest channel, which has the highest capillary pressure, but the nearest channel that the filling fluid reached first. The results of the LB simulation and the experiments show very good agreements, almost all the main process and interface movements were captured by the LB simulation.

For the imbibition in the Berea sandstone micro-model (Figure 6.20), the results from the experiment and the LB simulation showed excellent agreement. It can be observed that the decane imbibes from the bottom right corner and enters the nearest top right channel first. After that it entered the top left channel which has the highest capillary pressure. This result supported again our hypothesis, for the case of imbibition, the local geometry of the network model junction determines the filling sequence, rather than the capillary pressure of the channels.

Although the agreement of displacement of interfaces between the simulations and the experiments are quite satisfying, the time scales did not match well. This is due to the neglect of density ratio between decane and air. The approximation of density ratio equal to 1 is based on the assumption of a low Reynolds number. However, the Reynolds number increases dramatically when the interface of decane touches the solid wall. As a result, the inertia plays an important role at that moment, and the density

ratio is presumably no longer negligible.

6.3.3. Summary and outlook

In this chapter, the primary drainage and spontaneous imbibition of decane/air systems in two micro-models with increasing complexity were investigated. Experiments and the LB simulations were performed to study the flow in these micro-models. The LB simulations successfully simulate 3D displacement of decane/air systems, as the observed displacement process was well matched. The results demonstrate that the capillary filling rules used in the network modelling may not be valid for the case of imbibition, instead the local geometry plays a major role. The comparison illustrates that the LB simulation is promising and quite adequate for multi-component fluid flow simulation in porous media.

6.4. Relative permeability of reservoir rocks

6.4.1. Discussion of phase separation and wettability

In this section we present results of immiscible fluid flow using the optimised colour gradient model [66] discussed in section 3.10.3. We first study phase separation and wettability effects. Then we describe calculations of relative permeabilities in the Bentheimer rock sample.

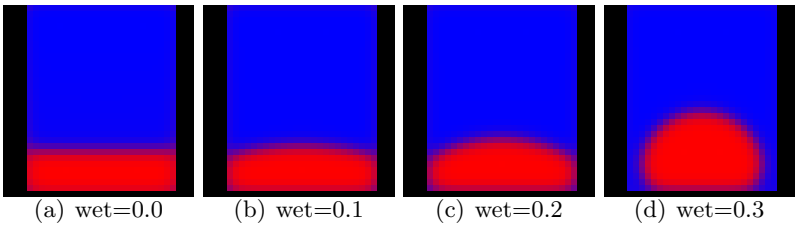
The phase separation of a binary immiscible mixture in LB simulations depends on a number of parameters, particularly the value of surface tension σ. LB simulations of critical spinodal decomposition in binary immiscible fluids [114] were also carried out to determine the properties of a binary mixture as a function of different parameter sets. Different wettabilities are implemented in our model by assigning a particular order parameter ϕ_w to the solid sites (the surface). This order parameter is not evolving with the LB equation, but exerts a phase colour gradient on the neighbouring fluid, with the same surface tension of σ as the fluid-fluid interaction. For example, for water wettability, if the order parameter of the water phase is $\phi = 1.0$, then a value between 0.0 (neutral wetting) and 1.0 (maximal wettability) can be assigned to the solid sites. The contact angle θ of the wettability is given by $\theta = arccos(\phi)$ [36].

To verify that our implementation of two-phase flow simulations with the LB method is able to model a wide range of wettability situations, we used a simple model of a box containing two immiscible fluids above a surface with variable wettability. The non-wetting fluid is initially placed in a cubic configuration at the centre of the box and in contact with the surface, then a LB simulation is performed until the non-wetting phase has reached its equilibrium shape. The results are shown in Figure 6.21 for oil concentration of 25% (the oil phase is coloured in red). When the surface is non-wetting, the non-wetting (oil) phase spreads over the entire surface. As the water wettability of the surface increases, the non-wetting phase starts to detach from the surface, and forms a contact angle with it. This contact angle can be defined as the angle between the wet surface and a plane tangent to the non-wetting phase droplet where the droplet starts to detach from the surface, as illustrated in Figure 6.22. This contact angle increases with increasing wettability. At high wettability ($\phi_{solid} = 0.8$) the non-wetting phase is completely detached from the surface and forms a spherical bubble.

6.4.2. Capillary pressure

The capillary pressure for Bentheimer sandstone (Fig. 4.9, Table 8.1) is measured numerically using LBM. A drainage process is simulated to mimic the experimental configuration. A buffer layer of 10 lattice sites wide is added at the inlet of the sample for injecting non-wetting phase. A porous plate shown in Fig. 6.3 is added at the outlet of the sample to prevent the non-wetting phase flowing out. This configuration is consistent with the experimental setup. Periodic boundary conditions are used for flow calculation, and a constant colour boundary condition, which converts all entry fluid into non-wetting, is applied for the inlet. A certain pressure gradient ΔP is applied to both phases until the system reaches steady state (saturation becomes constant). Then, one point on the capillary pressure curve is obtained. More points on the curve are obtained by applying different pressure gradients ΔP.

The calculated capillary pressure is converted into the dimensionless Leverett J-function:



(a) wet=0.0 (b) wet=0.1 (c) wet=0.2 (d) wet=0.3

(e) wet=0.4 (f) wet=0.5 (g) wet=0.6 (h) wet=0.8

Figure 6.21.: Equilibrium configurations (as 2D slices taken at the centre of the 3D system) of a binary immiscible fluid mixture as a function of increasing water wettability of the bottom surface. Wetting phase (water) is depicted in blue and non-wetting phase (oil) in red. Note the finite size effects in panel (c), where the detaching droplet of oil touches the lateral boundaries of the box. In this case, it is not possible to determine the contact angle. However, there are no finite size effects for all the other values of wettability.

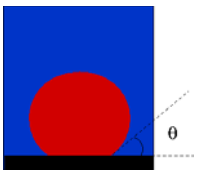


Figure 6.22.: Schematic illustration of how the contact angle θ is defined.

$$J(s) = \frac{P_c}{\sigma \cos\theta}\sqrt{\frac{\kappa}{\phi}}, \tag{6.11}$$

where P_c is capillary pressure, κ absolute permeability, ϕ the porosity, σ surface tension, θ contact angle. The comparison of computed and measured capillary pressure curve is shown in Figure 6.23. The agreement between the experimental measurement and simulation is generally good. The predicted irreducible wetting phase saturation ($S_{wi} = 7\%$) is similar to the experimental measurement. However, the calculated capillary pressure for this saturation is significantly lower than the experimental data. This may be due to resolution effects, some very small pores or throats may not be captured by XMT imaging. Some very small pores or throats in the geometry are also difficult for LB simulation. These pore sizes or throat sizes may be smaller than the width of the interface of LB simulation, therefore very high capillary pressure may lead to low numerical stability. These difficulties can be overcome using high resolution, but it will significantly increase the computing expense.

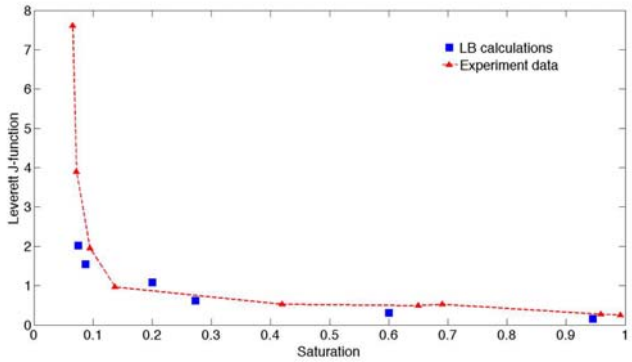


Figure 6.23.: Computed versus experimental capillary pressure curves for Bentheimer sandstone.

6.4.3. Relative permeabilities

There are two main methods to measure the relative permeability in the laboratory: unsteady and steady method. The steady state method [115, 116], which is most widely used in the lab, injects a fixed ratio of two phases simultaneously at constant flowrate through the porous media until the

saturation and pressure drop become steady. The hysteresis effect is reproduced by injecting fluid into the sample according to the direction of the saturation change. In the unsteady measurement [116], which is the quickest laboratory method, the sample is initially saturated with one phase only, and the other phase is injected into the sample to displace the in-situ fluid. The changes in pressure and fluid produced are continuously measured. The relative permeability of the in-situ and driving fluid are calculated using an equation originally developed by Buckley and Leverett [117, 118]. The saturation is not required to achieve equilibrium,which makes this method fast and of low cost compared to the steady state method. It is worth noting that this method is based on the assumption that the flow velocity is high enough, which means that capillary end effects are negligible [108]. An example of the unsteady-state method is the Johnson-Bossler-Naumann (JBN) method [119].

To the author's knowledge, only a few papers have studied the direct calculation of relative permeability of reservoir rocks [108, 3, 44]. The calculation requires a high level of accuracy, reliability and efficiency of the algorithm. It is also very difficult to recover the lab experiment configurations in simulations. The available papers of direct calculation of relative permeability on reservoir rocks randomly distribute the non-wetting phase according to the desired saturation [46, 3, 44]. This initial configuration is easy to set up but has a number of disadvantages:

- It does not recover the lab experimental measurement procedures.
- Some small or dead pore/channels might be occupied by the non-wetting phase, which is not the case in the experiments due to high capillary pressure.
- Is not able to calculate the imbibition relative permeability curve due to the lack of drainage-imbibition hysteresis

In this study, we are going to examine the random distribution relative permeability calculation method, and propose another new method for direct calculation of relative permeability which considers drainage-imbibition hysteresis and can produce both a drainage and an imbibition relative per-

meability curve.

Firstly, we calculated the relative permeabilities using the random distribution method for both oil and water of Bentheimer sandstone of size 320x320x320 with a resolution of $4.9\mu m$. A driving force was applied on both phases. To initialise the oil and water phases, each fluid node is randomly set as oil or water according to the desired saturation. A surface tension $\sigma = 0.01$ in lattice units and $20N/m$ in physical units was used in the simulations. The contact angle was set as 35° uniformly across the whole sample, this value is consistent with oil-water contact angles in water-wet rocks [3]. In all cases, we run the simulations until a steady state is achieved. As the initial distribution of the non-wetting phase is randomly set and followed with a phase separation process, the non-wetting phase might be found in some pores that should not be occupied by the non-wetting phase due to the high capillary pressure or local geometry. To take into account these inaccessible pores, the experimental data is rescaled using the Equation [120]:

$$S_w = \frac{S_w^* - S_{wi}}{1 - S_{wi}}, \tag{6.12}$$

where S^* is the total saturation of the wetting fluid and S_{wi} is the "irreducible" wetting saturation. In this case, $S_{wi} = 0.05$ according to the primary drainage experiment [3]. The non-wetting phase distribution of different wetting phase saturations equal to 0.2, 0.4, 0.6, 0.8 is shown in Figure 6.24. As we can see from these figures, for low wetting phase saturations, the non-wetting phase occupies most of the pores, and most of the non-wetting phase is connected. Due to the high connectivity of non-wetting phase for low saturations, the relative permeability of the non-wetting phase is much higher than that of the wetting phase. For high saturations, most of the pores are occupied by the wetting phase, whereas the non-wetting phase turns into a lot of isolated droplets due to the capillary forces. These droplets can hardly move due to high capillary forces, and jam the small throats for which the local capillary force is high. These hardly moving droplets not only reduce the relative permeability of the non-wetting phase, but also significantly reduce the relative permeability of the wetting phase due to the blocking of small pores and throats. The simulation results

and primary drainage experimental data by Ramstad *et al.* [3] are shown in Figure 6.25. In all cases, the rock is considered to be water-wet and the capillary number Ca is controlled by body force and set as 10^{-5}. The simulated non-wetting phase relative permeability is in fair agreement with the experimental data. The wetting phase relative permeability agrees well with the experimental data at medium and high saturation, but slightly over-predicts at low saturation. This could be caused by the finite resolution of the mesh. For low wetting phase saturations, most of the wetting phase is close to the rock as a thin layer of which the thickness should realistically be much smaller than the radius of the channel or pores. Due to the finite resolution of the mesh, the thickness of these layers may be over-predicted and therefore yield a slightly higher calculated relative permeability.

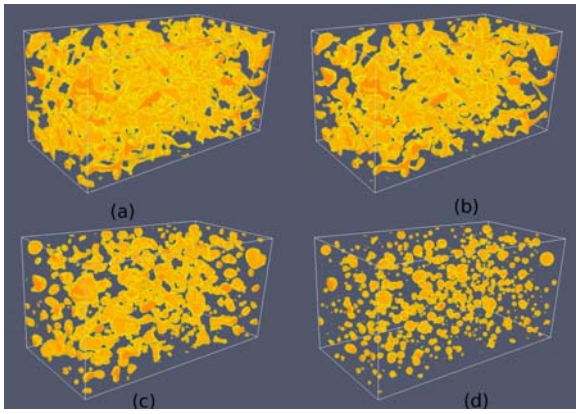


Figure 6.24.: Oil (Non-wetting phase) distribution of different saturation in Bentheimer sandstone: (a) $S_w = 0.2$, (b) $S_w = 0.4$, (c) $S_w = 0.6$, (d) $S_w = 0.8$

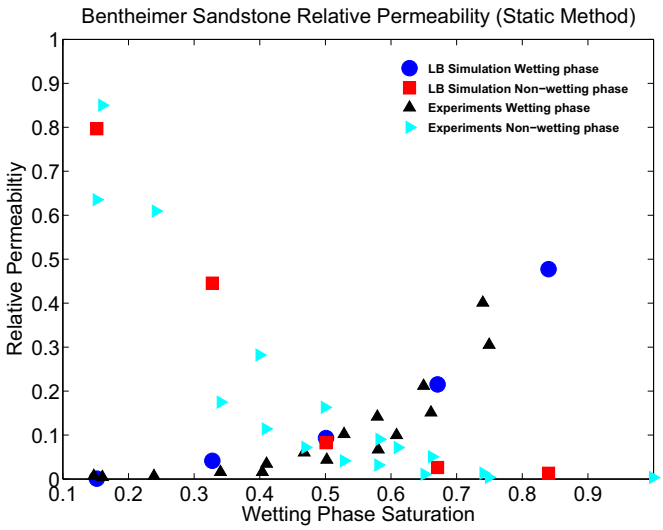


Figure 6.25.: Relative permeabilities for wetting phase (water, blue round points) and non-wetting phase (oil, red squared points) in a sample of Bentheimer sandstone of 320^3 voxels.Primary drainage experimental data (by Ramstad *et al.* [3]) are shown as triangular points.

To consider the drainage-imbibition hysteresis, a forced drainage, followed by a forced imbibition was carried out to generate the distribution of oil/water and then used as the initial distribution for the relative permeability calculations. The rock sample is fully saturated with water, except for a buffer layer of thickness 10 lattice sites, which is set near the inlet and outlet of the sample. The buffer layer at the outlet is saturated with water while the buffer layer at the inlet is saturated with oil and forcibly injected into the sample until the saturations converge which completes the drainage calculation. The distribution of oil/water at different saturation values is saved for the drainage relative permeability calculations. To carry out the imbibition calculation, the fluid in the buffer layer is then changed into wetting phase and the forced injection continues until the saturations converge again. The drainage-imbibition calculation snapshots are shown in Figure 6.26 and Figure 6.27. The distributions of oil/water are saved for the imbibition relative permeability calculations. These distributions need post-processing to be used as the initial distribution for the relative permeability calculations. The buffer layers are removed from the geometry and

the geometry is mirrored in order to use periodic boundary conditions in the relative permeability calculation. The post-processing flowchart is shown in Figure 6.28.

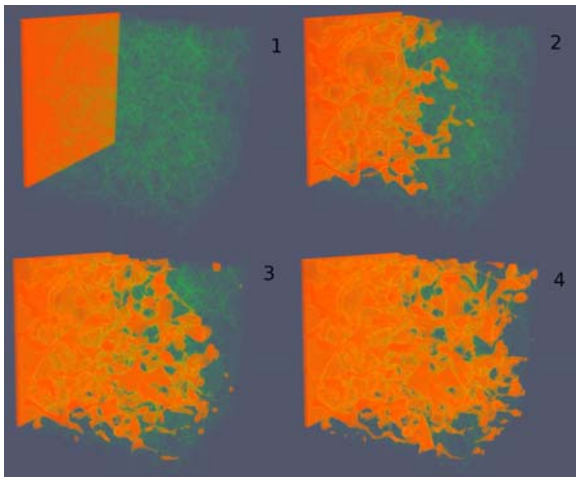


Figure 6.26.: Primary drainage simulation of Bentheimer sandstone.

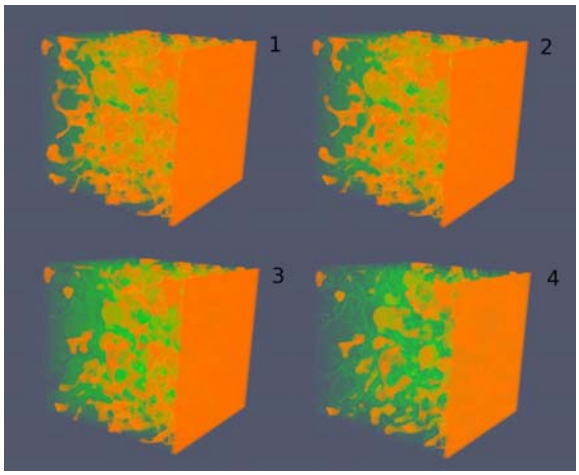


Figure 6.27.: Forced imbibition simulation of Bentheimer sandstone.

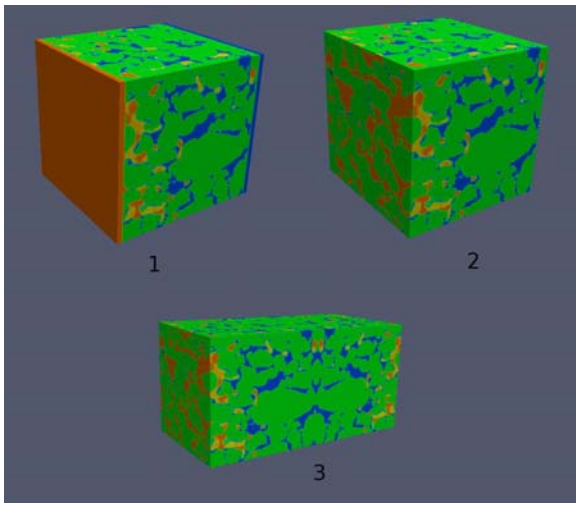
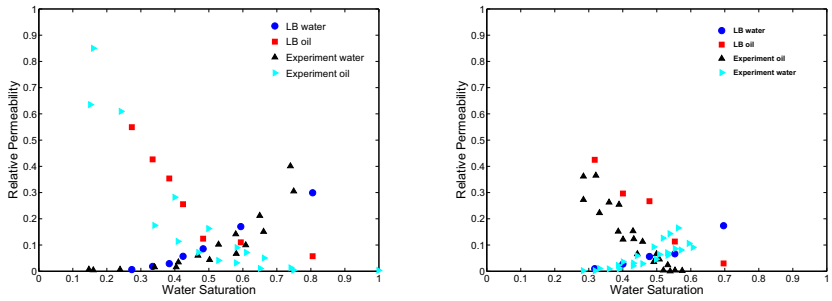


Figure 6.28.: Post-processing of drainage-imbibition results for relative permeability calculation: (1) drainage/imbibition results at a desired saturation (2) remove the buffer layers at the inlet and outlet (3) mirror the geometry along with wetting/non-wetting phase distributions for relative permeability.

With the distributions of oil/water from the drainage-imbibition calculation, a body force was applied on both phases. The magnitude of the body force was determined by the desired capillary number. In this study, the capillary number was controlled to be near 10^{-5} which is similar to that used in the experiments. After the permeability of both phases converged, the relative permeability for a desired saturation was obtained. With these data, we can produce a drainage relative permeability curve and an imbibition relative permeability curves for both drainage and imbibition as in Figure 6.29.

The figures show that the calculation results using our new method are generally in good agreement with experimental data. For the drainage relative permeability calculation, the new algorithm is of similar accuracy as the random distribution algorithm. However, the non-wetting phase (oil) relative permeability is slightly over-predicted for high saturation imbibition calculations. The relative permeability end point for imbibition is also over-predicted by simulation. These discrepancies may be caused by the



(a) drainage cycle (b) imbibition cycle

Figure 6.29.: Relative permeability simulation and experimental data for Bentheimer sandstone.

finite resolution of the mesh in small pores and throats. As a uniform mesh is used in the simulation, the number of mesh mesh points is limited, part of the thin wetting layer (with a thickness of which is smaller than the imaging resolution, in this study $4.9\mu m$) near the rock surfaces and small scale snap-off oil droplets may not always be captured effectively [121]. The thin wetting layers near the rock contribute little to the general wetting phase relative permeability. Therefore the lack of wetting layers does not affect the wetting phase relative permeability results significantly. In contrast, the lack of small non-wetting (oil) droplets which have high capillary pressure and possibly block small channels, may increase the relative permeability of the non-wetting phase because fewer channels are blocked. A potential solution could be to use a finer mesh. However, this will decrease the computational efficiency.

Generally speaking, this new direct relative permeability method can predict the relative permeability with reasonable accuracy for both drainage and imbibition. This method is recommended for relative permeability calculations for following reasons:

- The new method is closer to the lab operations
- The saturation does not need to be normalised to consider the "irreducible" pores
- It is closer to the reservoir condition.

- Drainage and imbibition hysteresis is considered in the method and both drainage and imbibition relative permeability can be predicted.

However, it does have several disadvantages including:

- The drainage and imbibition simulation take significantly more time than the random distribution method.
- Extra post-processing of drainage and imbibition results are needed.
- The non-wetting phase relative permeability might be over-predicted due to the limit of the mesh resolution.

6.5. Calculation of cluster size distributions for residual oil in sandstones

6.5.1. Introduction

The displacement of non-wetting phase (oil/super-critical(SC)-CO_2) in porous media is of great importance in many engineering applications such as enhanced oil recovery (EOR) or Carbon Capture and Storage (CCS). A residual phase of non-wetting phase remains in the porous medium due to capillary trapping. This may be advantageous in carbon geo-sequestration, where residual trapping of displaced CO_2 ensures long-term storage [4, 122, 123]. On the other hand, it may be a problem in oil recovery, where this portion of oil is trapped in the pores and cannot be produced by normal operations such as waterflooding. Therefore a better understanding of the characteristics of the capillary trapping of the residual phase has become a central issue in both oil and gas EOR and CCS improvement design.

Lorenz and Ziff [124] studied capillary trapping assuming that it is a percolation process, where the wetting phase fills the small throats which have high capillary pressures, while keeping the non-wetting phase in larger pores. They found that the number of residual clusters $N(s)$ of size s, scales with an exponent τ. A value of 2.189 was found for normal percolation [124, 125]. Blunt *et al.* [126] simulated the drainage and imbibition using a cubic lattice three-dimensional network model. A power-law non-wetting

cluster size distribution was observed with a lower exponent $\tau = 2.12$.

To study the characteristics of the residual phase, including residual saturation and size distribution, the non-wetting phase was solidified in order to measure the non-wetting phase directly in early studies [127]. X-ray microtomography (XMT) methods have recently been used to image both the pore space and fluids. The grains, wetting and non-wetting phase can be observed directly with a resolution of a few microns [125, 128, 4, 129, 130]. From XMT studies in Doddington and Clashach sandstones, residual cluster size distributions of oil/water and sc-CO_2 were measured and approximate power-law distributions were observed [125, 128]. The effect of wettability on residual cluster size distribution in beadpacks was investigated and more small clusters were observed in oil-wet conditions [131, 132]. Iglauer *et al.* [4] measured the residual oil cluster size distribution in Doddington and Clashach sandstone of different wettability conditions. A higher power-law exponent $\tau = 2.12$ for the oil-wet core was found compared to the water-wet core samples. In this study, a 3D lattice Boltzmann multi-component code is used to simulate two-phase flow in two reservoir sandstone samples: Bentheimer and Clashach. Residual cluster size distributions are calculated and compared with available experimental data [128, 4].

6.5.2. LB method for two-phase displacement in reservoir rocks

The samples were imaged using x-ray microtomography (XMT) with a grid resolution of 4.9 microns/pixel for Bentheimer sandstone and 8.96 microns/pixel for Clashach sandstone. The images were filtered and segmented into a binary file to provide information of solid and pore structure. The reconstructed 3D geometries are shown in Figure 6.30 and Figure 6.31. A 3D single phase LB solver is used to calculate the absolute permeability and a colour gradient multi-component LB solver [121] is used to simulate the multi-phase flow in the samples. The grid and calculated petrophysical properties, and fluid parameters used in simulation are summarised in Table 6.1 and 6.2.

To simulate the drainage and imbibition processes, two buffer layers of

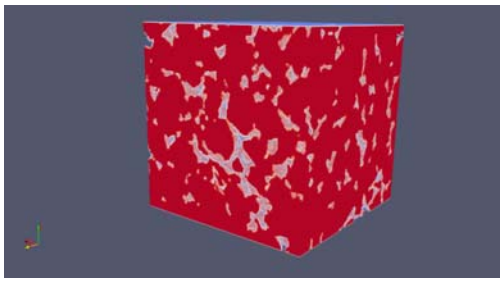


Figure 6.30.: A 3D image of Clashach sandstone with a resolution of 8.96 microns/pixel. The red represents grains while the light grey represents the pore space.

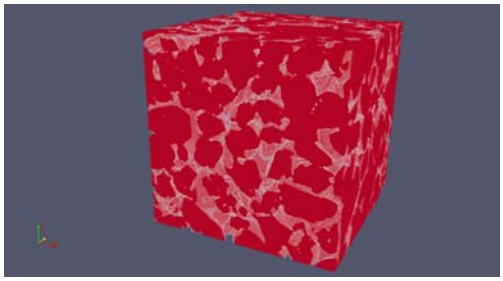


Figure 6.31.: A 3D image of Bentheimer sandstone with a resolution of 4.9 microns/pixel.

10 lattice sites each are added at the inlet and outlet of the sample to allow fluid injection, and flow out. At the beginning, the sample is fully saturated with water (wetting phase) except for the buffer inlet layer, which is saturated with oil (non-wetting phase) (Figure 6.32). A uniform body force is applied in the x-direction to drive the non-wetting fluid into the sample. Periodic boundary conditions are applied, so that the fluid flowing out will enter the sample on the opposite side. The out-flowing fluid will also re-colour into non-wetting phase before entering the inlet. This mimics a pure non-wetting phase injection drainage process. The simulation is stopped

Sample	Size	Resolution	Porosity	Calculated Permeability
Bentheimer	300 x 300 x 300	$4.9\mu m$	0.22	4970 mD
Clashach	300 x 300 x 300	$8.96\mu m$	0.132	129 mD

Table 6.1.: Sample size and petrophysical properties

Quantity	simulation in lattice unit	simulation in physical unit	Experimental value[4]
Density wetting phase (Brine)	1.0	$1000kg/m^3$	$1030kg/m^3$
Density non-wetting phase (n-Decane)	1.0	$1000kg/m^3$	$730kg/m^3$
Viscosity wetting phase	0.05	$1.25mm^2/s$	$1.039mm^2/s$
Viscosity non-wetting phase	0.05	$1.25mm^2/s$	$1.26mm^2/s$
Surface tension	0.01	12.5mN/m	23.8mN/m
Contact angle	35°	35°	35°

Table 6.2.: Fluid properties used in simulations

when the average flux rate and wetting phase converge to a constant value. This means that no non-wetting phase can be injected in the sample due to the high capillary pressure. Snapshots of the drainage process of Bentheimer and Clashach sandstones are shown in Figure 6.33 and Figure 6.34. From the figures, we can see that the big pores and channels are filled with non-wetting phase first, due to low capillary pressure. Some small pores and throats are not filled with non-wetting phase due to the very high capillary pressure. The fluid in these pores becomes residual wetting phase that cannot be drained out, unless a further increase of injection pressure is applied.

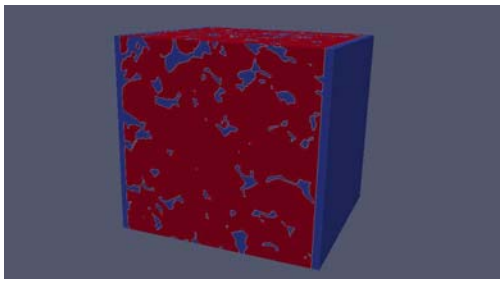


Figure 6.32.: Initial setup for drainage/imbibition simulation. The pore space, inlet and outlet buffer layers are shown in blue.

The inlet buffer layer is then replaced with wetting phase and the fluid flowing out is recoloured as wetting to conduct the imbibition simulation. The terminal condition of the simulation is similar to drainage: the simulation stops when the wetting phase saturations converge to a constant value. The pressure applied ensures a low corresponding capillary number

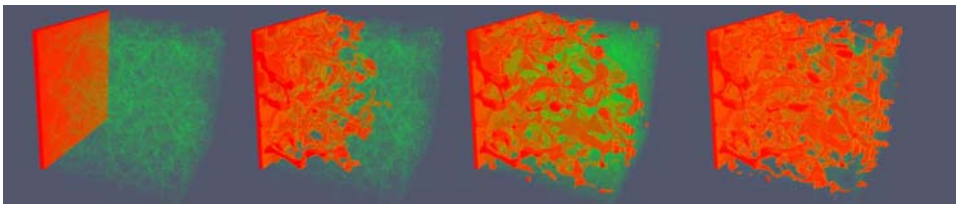


Figure 6.33.: Snapshots of drainage process simulation of Bentheimer sandstone. The non-wetting phase (oil) is shown in red and the rock is shown in transparent green. Saturation increases from left to right.

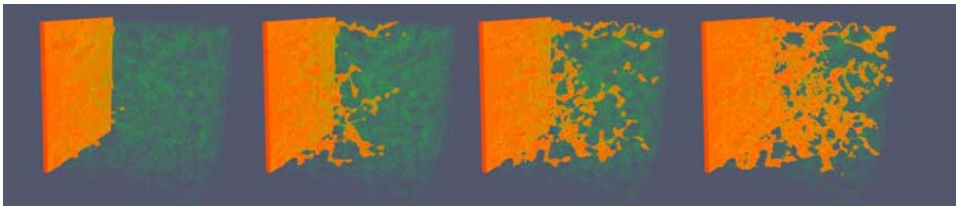


Figure 6.34.: Snapshots of drainage process simulation of Clashach sandstone.

$N_{cap} = 3.56 \times 10^{-4}$ which is consistent with experimental data.[2]

Snapshots of imbibition and a cross section of residual oil distribution are shown in Figure 6.35, 6.36 and 6.37. We can see that due to the wettability, the wetting phase contacts the sample surfaces and fills the narrow regions first, whereas the non-wetting phase preferentially stays in the centre of the pores, which leads to snap-off of non-wetting phase. The strength of snap-off depends on the capillary number and wettability. When the contact angle increases, the piston-like advance dominates the displacement of flows and suppresses the snap-off, which leads to little or no trapping [133]. The simulation results are consistent with experimental data. The observed residual saturation in the imbibition simulation 21%. This is slightly higher than the experimental measurements 18.8%[4]. To directly compare the residual cluster sizes and shape, we plot the largest three residual oil clusters (12000 $\sim$ 15000 voxels), three medium size clusters (4000 $\sim$ 6000 voxels) and

[2]The capillary number is a dimensionless defined as $N_{cap} = \frac{V\mu}{\sigma}$, where V is the characteristic velocity, μ is the viscosity and σ is the surface tension. It represents the relative effect of viscous forces versus surface tension acting across an interface between a binary immiscible fluid system.

three small clusters (400 $\sim$ 700 voxels) in Figure 6.38. The large residual oil clusters are found to span several pores. As the cluster size decreases, the cluster spans fewer pores and becomes spherical in shape. The small clusters are all found to be spherical in shape and occupy only one pore. This observation is consistent with experimental data in the literature [125, 128, 4].

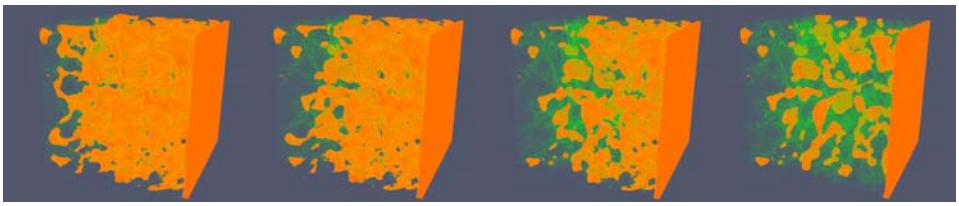


Figure 6.35.: Snapshots of an imbibition process simulation in the Bentheimer sandstone. Saturation decreases from left to right.

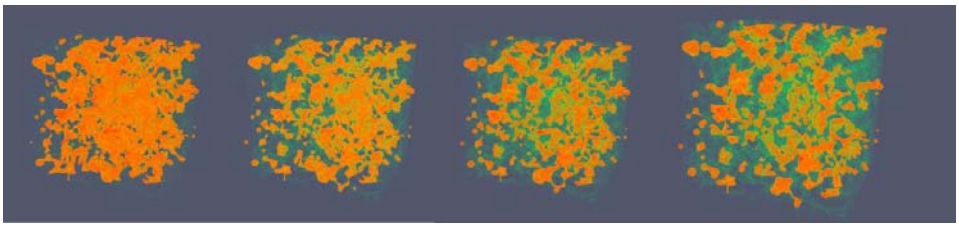


Figure 6.36.: Snapshots of an imbibition process simulation in the Clashach sandstone.

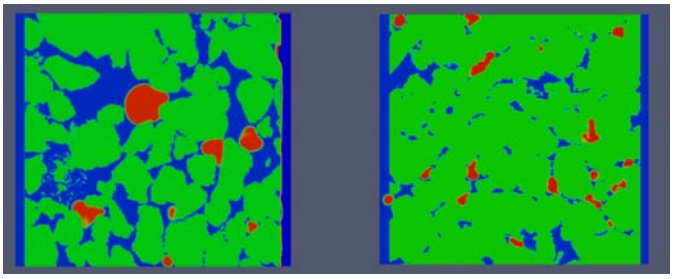


Figure 6.37.: Cross section snapshot of residual non-wetting phase distributions after waterflooding. Left: Bentheimer sandstone; Right: Clashach sandstone

To quantitatively compare the cluster size distributions, the cumulative cluster size distribution $S(s) = \sum_{i=s}^{\infty} in(i)$ [128, 4] is computed, where

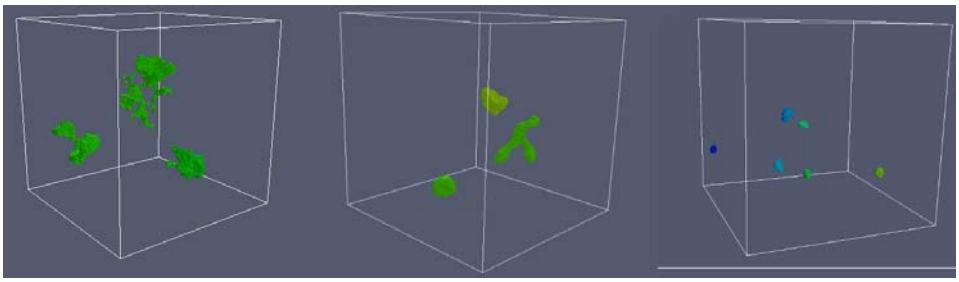


Figure 6.38.: Left: three largest residual non-wetting clusters (12000 $\sim$ 15000 voxels); Middle: three medium size residual non-wetting clusters (4000 $\sim$ 6000 voxels); Right: Six small size residual clusters (400 $\sim$ 700 voxels)

$n(i) = N(i)/N_v$, $N(i)$ is the number of residual clusters of size i and N_v is the total number of pore-space voxels. $S(s)$ is shown as a function of s on logarithmic axes for both Bentheimer and Clashach sandstone are plot in Figure 6.39 and Figure 6.40. The experimental measurements of Clashach sandstone is also shown in Figure 6.40 for direct comparison. Both figures show a similar tendency: we observe the power-law dependence of $N(s)$, and obtain $\tau = 2.10$ and $\tau = 2.13$ for Bentheimer and Clashach respectively. The simulation exponents are slightly lower than the percolation theory prediction $\tau = 2.189$ [133] but higher than the experimental measurements $\tau = 2.05$ (Clashach sandstone) [4]. This difference in τ indicates that the LB simulation results produce fewer residual oil clusters than observed in the experiments. This may be caused by the finite mesh resolution of the simulation. A uniform mesh of resolution 5 or 9 microns/pixel is used in the simulation; as a consequence, some small residual clusters may not be captured and this finite resolution may also affect the coalescence of clusters. It is worth noting that the cumulative cluster size distribution $S(s)$ increases dramatically at large size values $s > 10^4$, and levels out soon for $s < 10^3$. This indicates that, although the number of large clusters ($>$ 10000 voxels) is relatively small compared to the total number of residual clusters (less than 1%), it forms more than 80% of the residual saturation. The number of small residual clusters (less than 1000 voxels) is very big (90% of total residual cluster number), but only contributes a very small portion to the residual saturation (less than 1.5%).

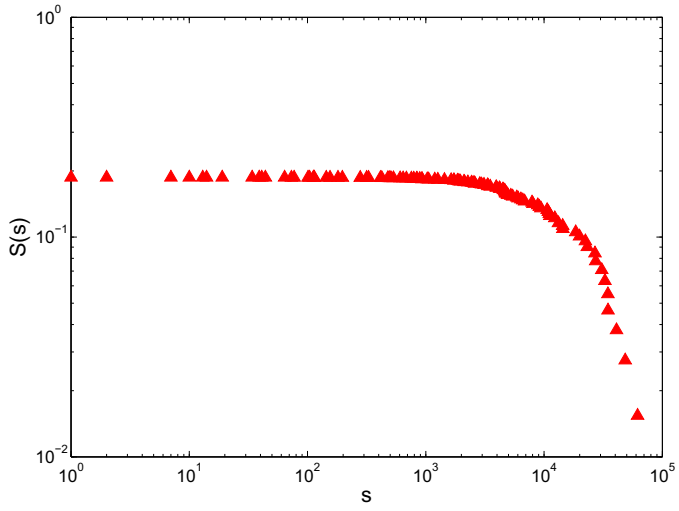


Figure 6.39.: Residual oil cluster distributions in Bentheimer sandstone. $S(s)$ is the cumulative cluster size distribution function, while s is the cluster size

6.5.3. Summary

In this study, the lattice Boltzmann method is used for the first time to directly calculate the residual cluster size distribution of Bentheimer and Clashach sandstone. The residual cluster distribution is generated following a full drainage and imbibition process. The shape characteristics of residual clusters and cluster size distributions are investigated and compared to available percolation theoretical study and XMT experimental measurements. We confirm conclusions from theoretical and experimental studies that the distribution of residual cluster size is a power law. Exponents are obtained close to the experimental values. This implies that the lattice Boltzmann method can handle the complex capillary-controlled displacement of multi-component systems in real reservoir rocks. This new technique can be of benefit to both Enhanced Oil Recovery (EOR) and Carbon Capture and Storage (CCS) design.

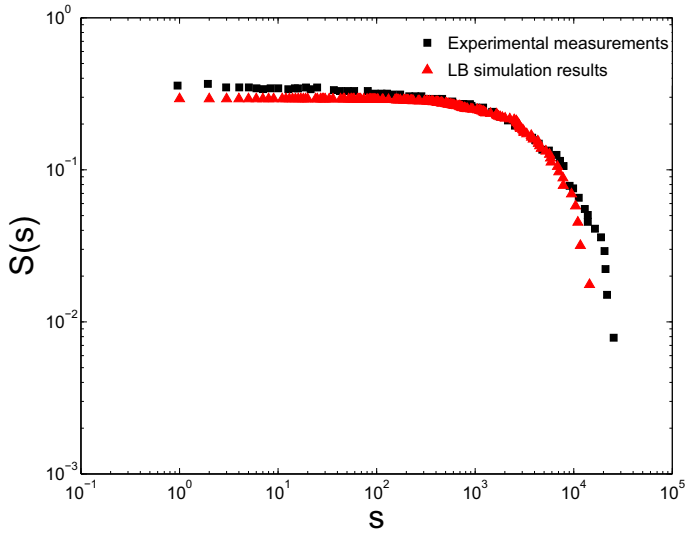


Figure 6.40.: Residual oil cluster distributions of experimental measurements[4] (black squared dots) and LB simulation results (red triangular dots) in Clashach sandstone.

7. Solute/Heat transfer simulation using the lattice Boltzmann method

7.1. Introduction

Solute/heat dispersion in porous media is of great importance in many scientific and engineering problems [134]. Although it is conceptually a simple process, the macroscopic behaviour is complex and a rigorous theoretical description remains an outstanding scientific challenge [135]. Various numerical methods have been proposed to investigate transport in porous media. Network modelling has been widely used to simulate the dispersion in porous media at the pore scale [136, 137, 138, 139]. However, for heterogeneous porous media, such as carbonate rocks, it is very difficult to reliably extract a network [140]. For this reason, direct calculation on three-dimensional pore space images was proposed to avoid network extraction [46]. Coelho *et al.* used a finite difference method to solve for the flow and dispersion in unconsolidated beadpacks and sandstones and found good agreement with experimental results [141]. By contrast, Adler and Thovert reported that the agreement with experimental data for Vosges sandstone, using similar methods, is not satisfactory [142]. Maier *et al.* used the lattice Boltzmann method to solve for the flow combined with a random-walk particle-tracking (PT) method to simulate dispersion in a pack of spheres. Comparison with nuclear magnetic resonance (NMR) spectroscopy experiments show agreement for transient, as well as asymptotic, dispersion rates [143]. Recently, Scheven *et al.*[7] reported molecular displacement (or propagator) distributions for different rock samples obtained from Pulsed Field Gradient - Nuclear Magnetic Resonance (PFG-NMR) experiments [7, 144]. They showed that the signature of the propagator distribution strongly depends on the

heterogeneity of the porous medium. Bijeljic *et al.* [145] simulated solute transport through pore space images of different rocks using a Stokes solver for the flow field and a streamline-based algorithm for solute dispersion. They observed qualitative agreement with the experimental NMR propagator results as reported in [7]. Here we calculate, for the first time, both the flow field and hydrodynamic dispersion from LB calculations. We consider rock samples of increasing heterogeneity: a beadpack, a Bentheimer sandstone and a Portland carbonate. The LB method has already been shown to be a particularly efficient method for the calculation of flow fields in complex geometries, such as porous media [104]. Here we extend the LB method and develop a new algorithm to calculate molecular displacement distributions directly from LB simulations. In this study, we obtain quantitative agreement with the experimental distributions reported and explain some outstanding questions in the experimental results [7].

7.2. Verification: Natural convection in a cavity

The natural convection within an enclosed cavity is studied using a thermal LB method described in Chapter 3.11. The setup for the calculation is shown in Figure 7.1. The top and bottom boundaries are adiabatic surfaces, while the left and right surfaces are set to $30°C$ and $10°C$ respectively. The initial temperature of the fluid inside was set as $20°C$ which is the averaged temperature between the hot and cold surfaces. The initial velocity was set to 0 for everywhere. The Mach number[1] and Prandtl number[2] of the fluid are 0.1 and 0.71 respectively. The simulations were tested with a range of Rayleigh numbers: $Ra = 10^3$, 10^4, 10^5, 10^6, 10^7. The Rayleigh number is a dimensionless number associated with buoyancy driven flow and measures the strength of conduction and convection. When the Rayleigh number is low, the heat transfer is in the form of conduction; when it is high, heat transfer is mainly in the form of convection. It is defined by:

$$Ra = g\beta\Delta T H^3/\alpha\nu \tag{7.1}$$

[1]The Mach number is a dimensionless quantity representing the ratio of speed of an object moving through a fluid and the local speed of sound [146].

[2]The Prandtl number Pr is a dimensionless number; the ratio of momentum diffusivity (kinematic viscosity) to thermal diffusivity [147].

where ΔT is the temperature difference between the hot and cold surfaces, H is the cavity dimensions, α is the thermal diffusivity and ν is the kinematic viscosity. The mean and maximum local Nusselt number, which is the ratio of heat transferred by convection to by conduction at the isothermal, vertical boundaries, are calculated and compared to the benchmark solutions [148]. The following equation is used to compute the mean Nusselt number and local Nusselt number:

$$Nu(x) = \int_0^n (PrRe \cdot u_1 - \frac{\partial T}{\partial x})dy \tag{7.2}$$

$$Nu_{local} = PrRe \cdot u_1 - \frac{\partial T}{\partial x} \tag{7.3}$$

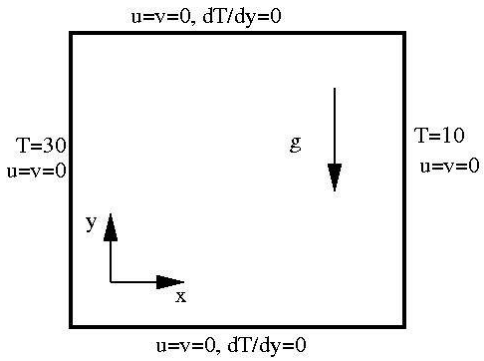


Figure 7.1.: Geometry of the cavity and initial setup of the simulation

7.2.1. Streamlines and Isotherms

Streamlines and isotherms are shown in Figure 7.2, 7.3, 7.4, 7.5, 7.6. These images were found to be in good agreement with those by Hortmann and Peric, de Vahl Davis [148, 149]. As can be seen in Figure 7.2, at $Ra = 10^3$, the the temperature contours are nearly parallel to each other, which means the dominant heat transfer mechanism is conduction. As the Ralyleigh number increases, the temperature field becomes flatter and stretches to the centre of the cavity (Figure 7.2, 7.4, 7.5, 7.6). This is due to the increasing buoyancy exerted on the hot fluid, and the heat transfer is becoming convective dominant.

For the streamlines, a single vortex is found in the low Rayleigh number simulation (Figure 7.2). As the Rayleigh number increases to 10^4, the vortex

becomes elliptical and stretches towards the vertical boundaries (Figure 7.3). Two vortices can be found for Rayleigh number 10^5. These rotate in a clockwise direction due to a small temperature gradient in the centre of the cavity (Figure 7.4). As the Rayleigh number increases to 10^6, the flow becomes more turbulent and a third vortex is formed in the centre. The two vortices at either side are stretched and move to the boundary (Figure 7.5). If the Rayleigh number increases further to 10^7, the two vortices on either side of the centre split into a large cortex and a smaller one. These are located closer to the walls (Figure 7.6). All the results agree reasonably well with the benchmark solutions [148, 149] .

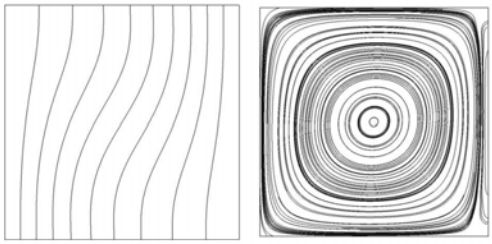


Figure 7.2.: Left: Isotherm, Right: streamline; Ra=10^3

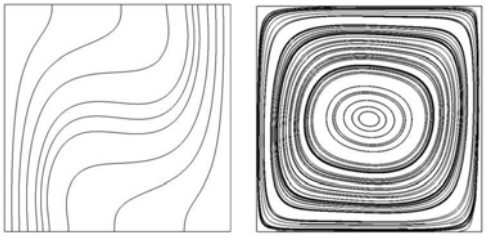


Figure 7.3.: Left: Isotherm, Right: streamline; Ra=10^4

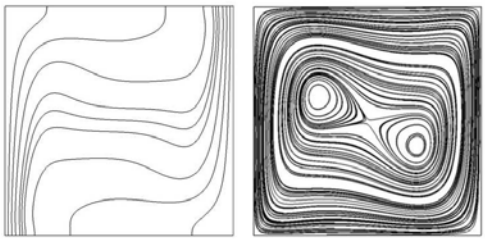


Figure 7.4.: Left: Isotherm, Right: streamline; Ra=10^5

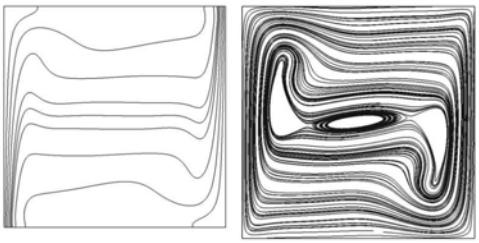


Figure 7.5.: Left: Isotherm, Right: streamline; Ra=10^6

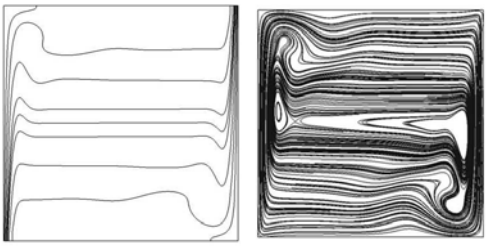


Figure 7.6.: Left: Isotherm, Right: streamline; Ra=10^7

7.2.2. The Nusselt Number

The calculated mean Nusselt numbers and maximum local Nusselt numbers are shown with benchmark solutions [148, 149] in Table 7.1.

Table 7.1.: Calculated mean and maximum local Nusselt numbers and comparison to benchmark solutions.

Ra	Nu_{mean} 128x128	Nu_{mean} 256x256	Benchmark Solutions [148, 149]	Nu_{max} 128x128	Nu_{max} 256x256	Benchmark Solutions [148, 149]
10^3	1.21121	1.11740	1.118	1.520	1.51552	
10^4	2.25326	2.24259	2.24475	3.56168	3.53555	3.53087
10^5	4.53345	4.52653	4.52164	7.82336	7.77109	7.772013
10^6	8.73804	8.82444	8.82513	17.5789	17.7613	17.5360
10^7	15.7321			36.7223		

The accuracy of the fine mesh (256x256) is found to be greater than 99.9% for all cases whereas the accuracy of the coarse mesh (128x128) varies from 99.7% to 91.7%. The increased accuracy with the fine mesh is due to the better resolution of the boundary layer. The accuracy can be further improved by using a finer mesh near the boundary and a coarse mesh at the rear part away from the boundary. This non-uniform mesh can guarantee

a high accuracy without increasing the computational time and memory usage [148].

The correlation for the mean Nusselt number and results obtained by Markatos and Pericleous [5] are plotted in Figure 7.7. The correlation proposed by Markatos and Pericleous can be summarised as:

For laminar flow ($10^3 <= Ra <= 10^6$) :

$$Nu = 0.143Ra^{0.299} \tag{7.4}$$

For Turbulent flow ($10^6 < Ra <= 10^12$):

$$Nu = 0.082Ra^{0.329} \tag{7.5}$$

The present Nusselt number results suggest a correlation for laminar region as:

$$Nu = 0.146Ra^{0.297} \tag{7.6}$$

This matches the benchmark solution well with a variation less than 0.6%. As can be seen in Figure 7.7, the present work matches the benchmark solution very well with a small variation between 0.3% to 1.4%.

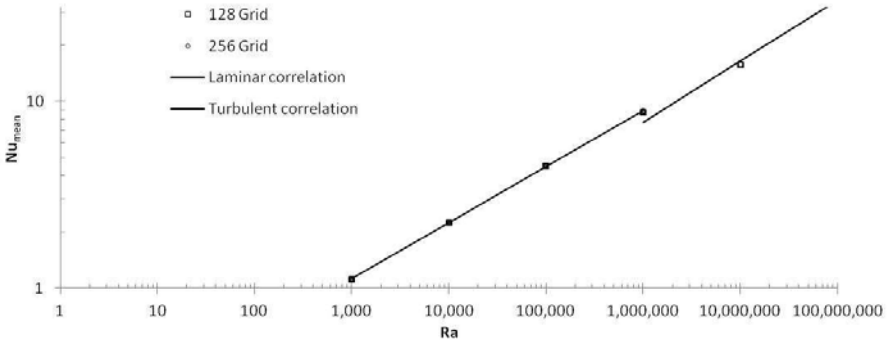


Figure 7.7.: Nusselt number as a function of the Rayleigh number, simulation results and benchmark solution given by Markatos and Pericleous [5]

7.3. Validation: Taylor Dispersion

The Taylor-Aris dispersion [150, 151] in a straight-walled channel under Poisseuille flow is calculated using the thermal LB code. The geometry and simulation setup is shown in Figure 7.8. The dispersion coefficient can be calculated analytically as [152]:

$$D = D_m + \frac{(2a)^2 v^2}{210 D_m} \tag{7.7}$$

where $2a$ is the width of the channel, v is the averaged velocity of the flow and D_m is the molecular diffusion coefficient. The dispersion coefficient of the flow can be calculated numerically by the equation [145]:

$$D = \frac{1}{2}\frac{d\sigma^2}{dt} \tag{7.8}$$

where σ is the variance of the particle displacement. The calculated dispersion coefficient as a function of channel width is plotted in Figure 7.9 along with the analytical solution. The simulation results match the analytical solution well.

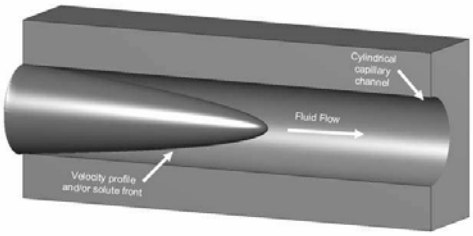


Figure 7.8.: Taylor dispersion (Figure from Sukop [6])

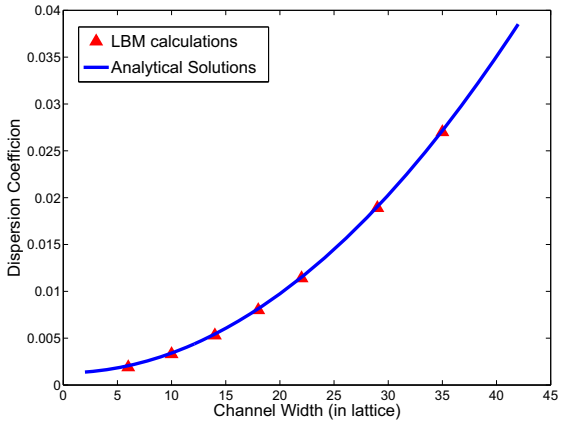


Figure 7.9.: Simulated Taylor-Aris dispersion coefficient as a function of channel width versus analytical solution. The red points are simulation results, the solid line is analytical solution

8. Lattice Boltzmann simulation of solute transport in reservoir rocks [1]

8.1. Introduction

Solute dispersion in porous media is of great importance in many scientific and engineering problems [134]. Although it is conceptually a simple process, the macroscopic behaviour is complex and a rigorous theoretical description remains an outstanding scientific challenge [135]. Various numerical methods have been proposed to investigate transport in porous media. Network modelling has been widely used to simulate the dispersion in porous media at the pore scale [136, 137, 138, 139]. However, for heterogeneous porous media, such as carbonate rocks, it is very difficult to reliably extract a network [140]. For this reason, direct calculation on three dimensional pore space images was proposed to avoid network extraction [46]. Coelho *et al.* used a finite difference method to solve for the flow and dispersion in unconsolidated bead packs and sandstones and found good agreement with experimental results [141]. By contrast, Adler and Thovert reported that the agreement with experimental data for Vosges sandstone, using similar methods, is not satisfactory [142]. Maier *et al.* used the lattice Boltzmann method to solve for the flow combined with a random-walk particle-tracking (PT) method to simulate dispersion in a pack of spheres. Comparison with nuclear magnetic resonance (NMR) spectroscopy experiments show agreement for transient, as well as asymptotic, dispersion rates [143]. Recently, Scheven *et al.* reported molecular displacement (or propagator) distributions for different rock samples obtained from Pulsed Field Gradient - Nuclear Magnetic Resonance (PFG-NMR)

[1] This chapter has been submitted to Water Resource Research for possible publication

experiments [7, 144]. They showed that the signature of the propagator distribution strongly depends on the heterogeneity of the porous medium. Bijeljic *et al.* [145] simulated solute transport through pore space images of different rocks using a Stokes solver for the flow field and a streamline based algorithm for solute dispersion. They observed qualitative agreement with the experimental NMR propagator results as reported in [7]. Here we calculate, for the first time, both the flow field and hydrodynamic dispersion from LB calculations. We consider rock samples of increasing heterogeneity: a bead pack, a Bentheimer sandstone and a Portland carbonate. The LB method has already been shown to be a particularly efficient method for the calculation of flow fields in complex geometries, such as porous media [104]. Here we extend the LB method and develop a new algorithm to calculate molecular displacement distributions directly from LB simulations. This new scheme need only small modification on the existing solute dispersion LB code to track particles displacement in porous media. In this study, we obtain quantitative agreement with the experimental distributions reported and explain some outstanding questions in the experimental results [7].

8.2. Methodology

We solve the steady viscous flow in porous media using a Multi-Relaxation-Time (MRT) D3Q19 lattice Boltzmann (LB) method [31, 32]. The MRT scheme and was reported to have significantly improved numerical stability compared with the single relaxation time LB model [33] and to eliminate the unfavourable viscosity dependency of computed velocity on relaxation time [44]. The optimzied relaxation parameters in this MRT scheme were equivalent to the two-relaxation-time (TRT) scheme which is much simpler and efficient for flow calculation in porous media [153, 154]. To solve the solute dispersion, we treat the solute as a passive scalar for which the advection-diffusion is simulated by a second distribution function, corresponding closely to the normal fluid distribution function, except with a simpler equilibrium distribution [6]. It has been shown that solute concentration are readily exchanged in this algorithm [6]. In order to solve the convection-diffusion equation (adiabatic, without source term):

$$\frac{\partial \phi}{\partial t} + \frac{\partial}{\partial x_j}(v_i \phi) = \frac{\partial}{\partial x_i}(D_{ij} \frac{\partial \phi}{\partial x_j}), \tag{8.1}$$

where ϕ is the concentration of solute and D_{ij} is the molecular diffusion coefficient, the concentration and diffusion coefficient are analogues to fluid density and viscosity respectively [6]. The simulation of solute dispersion does not require all the 19 velocities in the D319 model. Instead, only 6 perpendicular velocities are adequate for 3D simulations [155, 54]. This simplified single relaxation time D3Q7 solute dispersion model significantly reduces the memory and computing cost and makes it a very efficient method. Similar to lattice Boltzmann method for flow calculation, the phase space and velocities of solute lattice Boltzmann model are discretized by a regular lattice in three dimensions. In this D3Q7 solute LB model [156], the discrete velocity is defined as: $\mathbf{e}_i = (0,0,0),(\pm 1,0,0),(0,\pm 1,0),(0,0,\pm 1), \quad (i = 0,1,2,3,4,5,6)$. The evolution of the set of solute distribution functions $g_\alpha(\mathbf{x},t), \quad \alpha = 0,1,\ldots,6$ is given as below:

$$g_\alpha(\mathbf{x}+\mathbf{e}_\alpha\delta t, t+\delta t) - g_\alpha(\mathbf{x},t) = -\mathbf{M}^{-1}\mathbf{S}(m_\alpha(\mathbf{x},t) - m_\alpha^{eq}(\mathbf{x},t)), \tag{8.2}$$

where the equilibrium moments are defined as:

$$\mathbf{m}_{eq} = (0, v_1\phi, v_2\phi, v_3\phi, 3/4\phi, 0, 0)^T. \tag{8.3}$$

The macroscopic solute concentration ϕ is obtained with $\phi = \sum_{\alpha=0}^{\alpha=6} g_\alpha$. The transformation matrix $\mathbf{M}$ and the relaxation matrix $\mathbf{S}$ are given as:

$$\mathbf{M} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 \\ 6 & -1 & -1 & -1 & -1 & -1 & -1 \\ 0 & 2 & 2 & -1 & -1 & -1 & -1 \\ 0 & 0 & 0 & 1 & 1 & -1 & -1 \end{bmatrix}, \tag{8.4}$$

$$\mathbf{S} = \begin{bmatrix} \tau_0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \tau_{11} & \tau_{12} & \tau_{13} & 0 & 0 & 0 \\ 0 & \tau_{21} & \tau_{22} & \tau_{23} & 0 & 0 & 0 \\ 0 & \tau_{31} & \tau_{32} & \tau_{33} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \tau_4 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \tau_5 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \tau_6 \end{bmatrix}. \tag{8.5}$$

τ_0 in the relaxation matrix $\mathbf{S}$ has no affection on the numerical solution, τ_4, τ_5, τ_6 does not affect the leading-order approximation but affect error term, τ_i were set as $\tau_i = 1, (i = 0, 4, 5, 6)$ in this study. The diffusion coefficient matrix is selected as:

$$\tau_{ij} = \frac{1}{2}\delta_{ij} + 4D_{ij}, \tag{8.6}$$

δ_{ij} is the Kronecker's delta. Asymptotic analysis showed that this solute LB model is second-order accurate in space and first-order accurate in time [157].

The parameters selected can recover the convection diffusion equation 8.1 [157, 156]:

Table 8.1.: Petrophysical parameters

	Sample size (in pixels)	Resolution	porosity	permeability
Beadpack	512x256x256	$5.0\mu m$	0.357	2624.53 mD
Bentheimer sandstone	512x256x256	$4.9\mu m$	0.23427	4755 mD
Portland carbonate	640x320x320	$9.0\mu m$	0.0917	355mD

The 3D images of the samples were obtained by using micro-CT scanning of dry cylindrical cores at different resolutions (Table 8.1). The raw pictures were filtered using a 3x3 median filter and then segmented according to Otsu's algorithm [158]. We mirror the geometry of the porous media in the flow direction in order to apply periodic boundary conditions for both flow calculation and solute dispersion simulation. The sizes, resolutions and petrophysical parameters of the sample is given in Table 8.1. The permeability and porosity of the sample in different sizes were investigated in order to find the representative element volume (REV) which is a vol-

ume big enough to represent the macroscopic property of the core but small enough for direct numerical simulation. A constant body force term [58] was used to generate the desired pressure gradient [73]. Several optimisations strategies were used to improve the efficiency of the code. The data storage is optimised by using a sparse storage technique which only stores the fluid mesh points. This reduces the memory usage and data reading/writing by 70%-80%, depending on the porosity of the sample. The collision and streaming steps were combined into a single loop and a balanced domain decomposition scheme was implemented for MPI parallel computing. This solute dispersion lattice Boltzmann method has been validated using several test cases for which the analytical solutions are available. These include Taylor-Aris dispersion, natural convection flow in a square cavity and Rayleigh-Bérnard convection [6, 102].

In order to probe the molecular displacements in pre-calculated Stokes flow through porous media, we have developed a novel scheme to set up the simulation. First of all, the velocity field is calculated by a single phase MRT-LBM code. It is worth noticing that to maintain the stability and accuracy of the single phase LB simulation, the maximum local Mach number need to less than 0.05, which can be controlled by using appropriate body force/pressure gradient applied on the fluid. Because the reaction is not considered in this study, this velocities are to be used in solute dispersion calculation as constants. Therefore, in the solute dispersion calculation, the velocity and pressure field is not needed to be solved, only the convection-diffusion equation (Equation 8.1) is solved, which can improve the overall efficiency of the simulation. As the LB-based solute dispersion model is only able to solve the concentration change of solute in the sample, we cannot track the displacement of the propagators explicitly. To solve this problem, we initialise the concentration of solute $\phi_{\sigma_i}(x)$, where x is the coordinate of x axis, as follows:

$$\phi_{\sigma_i}(x) = \begin{cases} 1, & \text{if } x = i \\ 0, & \text{if } x \neq i \end{cases} \tag{8.7}$$

where $\phi_{\sigma_i}(x)$ is the concentration of solute σ_i and x is the coordinate in the flow direction. This scenario is equivalent to injecting all the propagators

at the same position $x = i$ in the flow direction. The solute concentration is simulated by solving the advection-diffusion equation using the lattice Boltzmann method. If the solute concentration $\phi_{\sigma_i}(x')$ at position x' is found to be larger than zero, this indicates that a portion $\phi_{\sigma_i}(x')$ of the propagators has been displaced to x'. The displacement of these propagators can easily be calculated as $x' - i$ (where i is the initial position for all the propagators). However, this is slightly different from the NMR experimental setup [7] in which the tracking particles were distributed uniformly across the whole sample. To solve this problem, we add component $\sigma_1, \sigma_2, \ldots$ to simulate molecules starting from different positions $x = 1, 2, \ldots$. This configuration is consistent with the NMR experiments [7]. This configuration is equivalent to conducting several solute dispersion LB simulations (in this study, we solve 100 solute dispersion equations), but we solve convection-diffusion equations of ϕ_{σ_i} within a single simulation which can save time on data reading and processing. Although this setup needs more memory compared to the conventional solute dispersion LB calculation, the calculation is very efficient, provided the code is carefully optimised using the D3Q7 LB model. The Péclet number $Pe = \frac{LU}{D}$ is set to around 15, where L is the characteristic length chosen as resolution, U is the volumetric averaged velocity and D is the molecular diffusion coefficient. This is consistent with the NMR experiments, for all the simulations, by modifying the mean velocity and molecular diffusivity. For LB based dispersion calculation, we found that the Péclet number is related with the stability of simulation, high Péclet number (Pe>100) significantly affect the stability of simulation. Therefore, we use relatively low Péclet number for all simulations.

8.3. Results

We first simulate diffusion in the pore space of a bead pack and find a restricted diffusion coefficient $D_r = \sigma^2/2\Delta t = 0.63 * D_m$, where Δt is the diffusion time, D_m is the molecular diffusion coefficient and $\sigma^2 = \langle(\zeta - \langle\zeta\rangle)^2\rangle$ is the variance of the molecular displacement distribution $P(\zeta)$ [7]. This agrees with the expected value [159] and supports our simulation approach. We show the geometries of the bead pack, Bentheimer sandstone and Portland carbonate, obtained from micro-CT scanning in Figure 8.1. The corresponding LB velocity distributions are also shown in the same figure. We observe

that the degree of heterogeneity of the velocity distributions increases from bead pack to Portland carbonate. In the bead pack, the distribution of pore widths is quite narrow and the velocity distribution is generally uniform; for sandstone, we observe significant differences in the flow rate for different channels or pores; for the carbonate, most of the flow is concentrated in a few large pores.

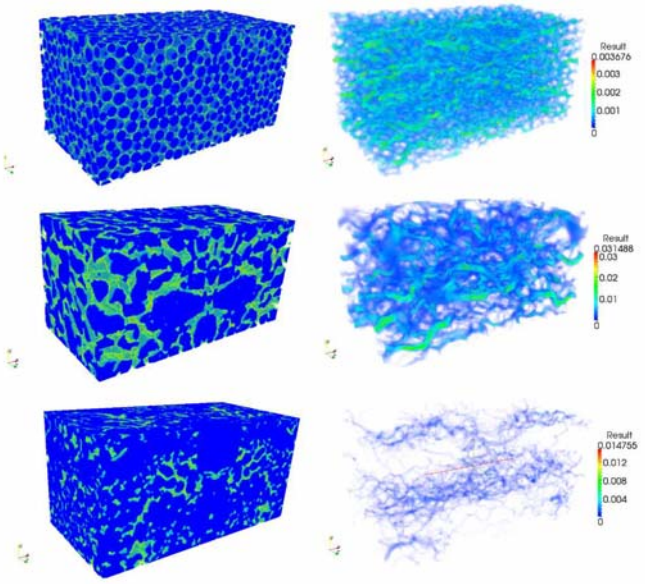


Figure 8.1.: Pore space images (left: pore is green, solid is blue) and velocity distributions (right) of porous media with increasing heterogeneity: 1) bead pack; 2) Bentheimer sandstone; 3) Portland carbonate. Resolutions are $5\mu m$, $4.9\mu m$ and $9\mu m$ respectively. Regarding the velocity distributions, red and blue indicate high and low velocities respectively.

We study transport in the porous media by calculating the probability distributions for molecular displacement from the lattice Boltzmann simulations. The simulation results for the three different porous media are shown in Figure 8.2 in direct comparison with the experimental results [7]. The probability distribution $P(\zeta)$ is rescaled into a non-dimensionalised form as $P(\zeta) \times \langle\zeta\rangle_0$, where $\langle\zeta\rangle_0 = v_{ave}\Delta t$ is the nominal mean (Darcy) displacement and v_{ave} is the Darcy velocity. The displacement ζ is rescaled to $\zeta/\langle\zeta\rangle_0$. In the bead pack (top in Figure 8.2), we observe that the simulation results

match the experimental data very well. A Gaussian distribution centred around the mean displacement is observed, according to expectation. From the results we conclude that all solute is displaced from the initial positions in the bead pack.

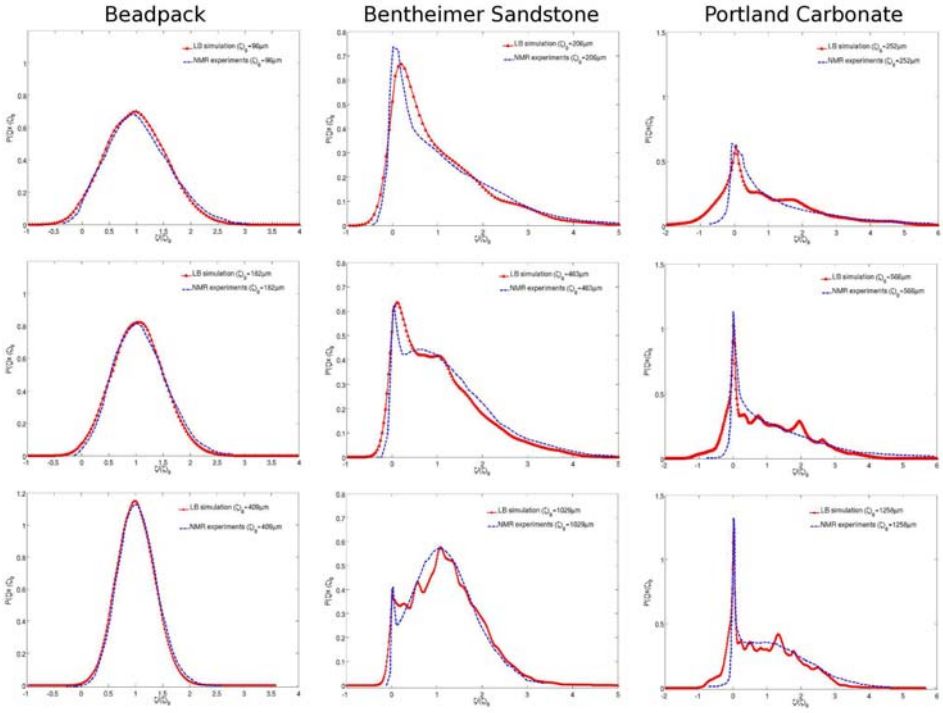


Figure 8.2.: Probability of molecular displacement in different porous media: bead pack (left), Bentheimer Sandstone (middle) and Portland Carbonate (right) for time t = 0.25s, 0.5s and 1.0s (from top to bottom) as a function of rescaled displacement. Simulation results (red) are compared with NMR experimental data (blue) from Scheven *et al.* [7]. The Péclect number is Pe=18 in the LB simulations.

In the Bentheimer Sandstone (middle in Figure 8.2), the simulation results generally agree well with the experimental data, except for a small deviation near the peak of the Gaussian distribution centred around the mean displacement $\zeta/\langle\zeta\rangle_0 = 1$. In addition to the Gaussian distribution around the mean displacement, a stagnant peak is found near $\zeta/\langle\zeta\rangle_0 = 0$. This indicates that a portion of the solute is trapped in slowly or non-flowing

regions, related to the heterogeneous geometry. Although the general distribution of probability of displacement match the experimental data well, the probability of displacement were found oscillating near the mean displacement, this might be due to the insufficient of initial solute pules, this initialization helps to save the computing source but only take into account part of the geometry initially whereas the experiments distribute the particle tracers uniformly across the sample. In the Portland carbonate (bottom in Figure 8.2), the simulation successfully predicts the displacement of the solute. The Gaussian distribution around the mean displacement is reduced significantly whereas the magnitude of the stagnant peak has grown. This means that more solute is trapped in slowly or non-flowing parts of the pore space. It is worth mentioning that the calculated displacement distributions for both Bentheimer sandstone and Portland carbonate show slightly bigger values than the experimental data for negative displacement regions $\zeta/\langle\zeta\rangle < 0$. To investigate the reason for this deviation, we plot the ratio of measured mean displacement divided by the nominal mean displacement $\theta = \langle\zeta\rangle/\langle\zeta\rangle_0$ as a function of evolution time (Figure 8.3) for all three rock samples. All the simulation results converge to 1, as expected, although a small deviation of $3\% - 5\%$ for Bentheimer sandstone and Portland carbonate is observed. From the figure, we can see that the ratio $\langle\zeta\rangle/\langle\zeta\rangle_0$ of Bentheimer sandstone decrease with time from a value of 3.3 whereas the Portland carbonate increases with time from a value of 0.76. These deviations from unity in early times are due to the different initial locations of the solute, in Bentheimer sandstone simulation, the solute were placed in a region with relative high velocity such as areas with well connected pores and throats while in Portland carbonate, the solute were initially put on areas with pores or throats with low velocities. As the solute displace and disperse with external flow, it spread to the whole sample, the ratio $\langle\zeta\rangle/\langle\zeta\rangle_0$ therefore eventually converge to 1. To investigate the deviation at early time region ($t < 0.5s$), we place the solute at different regions, we found that the initial value of ratio $\langle\zeta\rangle/\langle\zeta\rangle_0$ varies from 0.5 to 4, but after enough time, all converge to the same value which is close to 1. This is an evidence that the deviations at early time are caused by initial condition, but will not affect the final calculation results. In the NMR experiments, on the other hand, a significant deviation of $\langle\zeta\rangle/\langle\zeta\rangle_0$ from unity (up to $1.20 - 1.35$) was observed for the sandstone and carbonate. Scheven *et al.* [7] suggest that this may be

caused by surface relaxation effects, which lead to undercounts of stagnant spins from the measurement. Our calculations do not suffer from surface relaxation effects and therefore no undercount of stagnant spins is to be expected. Indeed, our stagnant peaks are wider than the experimental ones, particularly in the negative displacement region. In addition, we observe that the detailed agreement between simulation and experiment is better for the beadpack than for the Portland carbonate. The reason is that our computational domain is fixed (sample size = $5mm$ x $2.5mm$ x $2.5mm$), whereas the Representative Element of Volume (REV) presumably increases with increasing complexity. This issue will be evaluated in a forthcoming paper.

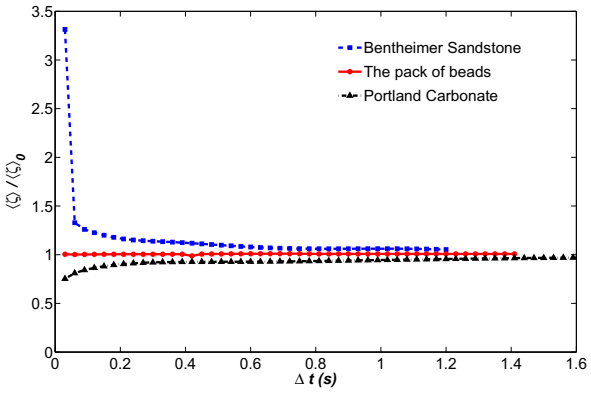


Figure 8.3.: Ratio of measured displacement $\langle\zeta\rangle$ and nominal mean displacement $\langle\zeta\rangle_0$ as a function of evolution time Δt for the beadpack, Bentheimer sandstone and Portland carbonate

To quantify the capability of solute trapping of different rocks, we calculate the percentage of solute with displacement less than $0.05 \times \langle\zeta\rangle_0$ as a function of time. The results are shown in Figure 8.4. As we can see in the figure, the concentration of slowly or non-flowing solute decreases with time and finally all the curves converge to a constant value. This means that this solute fraction can not be displaced by the flow and is trapped in the pores. No trapped solute is found in bead pack; the volume of trapped solute decreases to zero as all solute can be displaced by flow. A fraction of 1.49% solute is observed to be trapped in Bentheimer sandstone, increasing to 8.13% in Portland carbonate. This analysis suggests that the

solute trapping increases with an increasing degree of heterogeneity. Here we postulate the fraction of trapped solute as a new quantitative measure of heterogeneity. It is also worth noting that the convergence time (slope $<$ 0.01) of beadpack, Bentheimer sandstone and Portland carbonate increases from 0.86s to 1.67s and 1.81s. This suggests that the convergence time of solute transport in porous media (in terms of trapped solute in pores) increases with an increasing degree of heterogeneity.

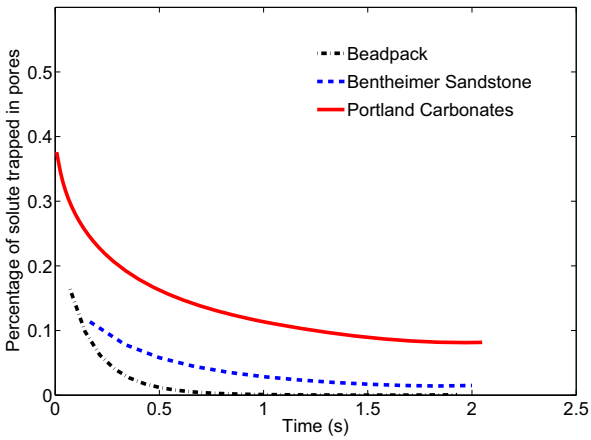


Figure 8.4.: Percentage of solute trapped in pores. The red solid line, blue dashed line and black dash-dotted line show the percentage of solute for which the displacement is less than 5% of the mean nominal displacement. The data is plotted as a function of time [seconds].

8.4. Summary

We use a lattice Boltzmann-based simulation to study flow and transport in three-dimensional realistic rock samples: a bead pack, Bentheimer sandstone and Portland carbonate. We develop a new algorithm based on the LB method to calculate both the flow field and solute transport in porous media. We compute the flow and displacement probability distributions for all porous media and observe good agreement with NMR experiments. We observe that the magnitude of the flowing Gaussian distribution reduces and the stagnant peak increases, as the degree of heterogeneity increases.

We quantitatively measure the trapping capability of three rock samples of increasing heterogeneity: the homogeneous beadpack does not trap any solute, while the Bentheimer sandstone and Portland carbonate retain 1.49% and 8.13% of solute, respectively, in stagnant pores. The experimental surface relaxation problem, leading to an undercount of stagnant spins, as found in the NMR experiments, is removed using the lattice Boltzmann simulations. The ratio of measured and nominal mean displacements of the lattice Boltzmann prediction converges to 1 for all the cases as expected. Compared to existing transport simulation tools such as streamline-based algorithms, the lattice Boltzmann-based method is easy to implement and handles the problem in an efficient way with very good accuracy. This method can be widely used to study transport problems in porous media, with applications in hydrology, chemical engineering, carbon storage and enhanced oil recovery.

9. Lattice Boltzmann Method Implementation

9.1. Introduction

A C++ code package was developed to carry out LBM simulations in 2D and 3D. An MRT scheme was used in the collision step to improve the numerical stability and eliminate the viscosity dependence of permeability calculation. Two separated versions of the code have been developed to simulate single phase and binary fluids system respectively. Several subroutines have been developed to compute the permeability, export simulation results in VTK format for 3D visualisation, refine the mesh and calculate the Reynolds number. The capability of this code package for single phase flow simulations (2D and 3D) can be summarised as:

- Simulate high Reynolds number flow (Re $<$ 2000, Karman Street Simulation)
- Simulate low Reynolds number flow with complex geometry
- Calculate permeability of porous media
- Refine the mesh

In order to simulate the flow of a binary fluids system, interfacial behaviours and wetting, three multi-component LBM models including the Shan-Chen pseudo potential model, the Colour Gradient Model, and the Free Energy Model were studied and implemented. Their performances were compared with theoretical predictions by numerical experiments. The multi-component LBM code package is able to simulate:

- Interface evolution with surface tension and wettability

- Binary fluids with different viscosity
- Binary fluids system with complex geometry
- Snap-off phenomena
- Capillary fingering
- Capillary pressure prediction
- Relative permeability prediction
- Initial-residual saturation prediction
- Residual saturation cluster size distribution prediction

9.2. Parallel LBM implementation

The LBM simulator is able to simulate various degrees of complex flow with extremely complicated boundaries. However, it is very time-consuming, especially for multiphase calculation, even with modern computers. The efficiency of LB calculation is crucial if the simulation is to be used for real industrial applications. In order to reduce the calculation time to a reasonably low level for realistic applications, careful optimisation is desired for algorithm implementation. In this chapter, introduction of parallel computing and three principal optimisation techniques used in LB code implementation are described. According to the features of reservoir rocks, three optimisation schemes are proposed:

- Sparse storage scheme to reduce the memory and CPU usage
- Parallel implementation of LB algorithms
- Local operation optimisations

9.2.1. Sparse storage scheme for LB simulations

Reservoir rock, as one type of porous media, is a material containing pores. The porosity, which is the fraction of the volume of voids over the total volume, varies from 2% to 40% [160]. The skeletal portion of the material, which is often called the matrix, is not involved in flow calculation (except

the fluid-solid surfaces). It forms 60%-90% of the sample volume, occupies most of the memory and computing resources and therefore leads to poor performance. In order to give these memory and computing resources to fluids nodes, the layout of data needs to be redesigned to reduce the memory requirements. Several studies were carried out on sparse storage schemes for LB simulations: the normal 3D arrays storing the data on cartesian coordinates are mapped to a continuous 1D array which only contains the data of fluid nodes. The neighbouring nodes information which is natural for the Cartesian coordinates system needs a separate 1D array to store [161, 162]. In this study, a compressed storage scheme for neighbouring nodes is used to minimise the extra memory usage.

We use a 2D porous medium with Cartesian coordinates as an example to describe the sparse scheme used in LB implementation. The extension to 3D geometries is straightforward. We consider a 2D porous medium in Figure 9.1, the colour black represents the matrix while the white represents the pores. The pore nodes are mapped into a 1D array $S[i]$ and the coordinates (i,j) are renumbered into an index from 0, Figure 9.2. The information of the the mapping procedure from a 2D coordinates system to 1D sparse storage array is saved in one 2D geometry array $G[i, j]$ and one 1D index array $R[i]$, Figure 9.3. These two arrays aim to convert the sparse index in 1D storage arrays to 2D coordinates and from 2D coordinates to 1D sparse storage. The 2D geometry array $G[i, j]$ is originally used to save the pore/solid information, and is added to the mapping information as: 0 for matrix nodes, $i > 1$ is the index of fluid nodes in the sparse 1D array. With this 2D geometry array, we can not only identify the solid surfaces but also easily convert the 2D coordinates (i, j) to the index of the fluids nodes in the 1D sparse array (Figure 9.2). It's easy to map from 2D to 1D, but 1D to 2D needs more careful consideration. A separate 1D array $R[i]$ is applied to store the mapping information from the 1D sparse array to 3D coordinates. It is defined as follows: if a fluid node on (i, j) is mapped to the 1D sparse array at location s, then $R[s] = i*ny+j$ (in 3D $R[s] = i*ny*nz+j*nz+k$) where nx, ny, nz is the size of the sample in x,y,z direction. For any index s in the 1D sparse array, to recover the Cartesian coordinates of this node, a simple calculation is carried out: $i = int(R[s]/ny)$ and $j = R[s]\%ny$. This definition compresses the two coordinates (i, j) (three (i, j, k) in 3D cases)

into a single integral number which saves half or more of the memory usage on neighbouring information storage. With $G[i, j]$ and $R[s]$, we can easily convert the index in the normal Cartesian coordinate system and the sparse system.

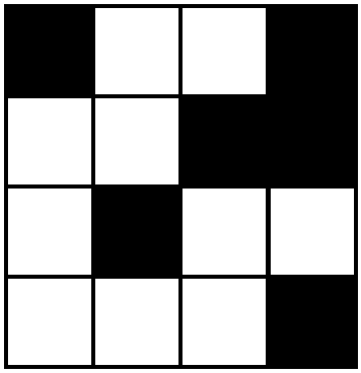


Figure 9.1.: Sample 2D porous medium

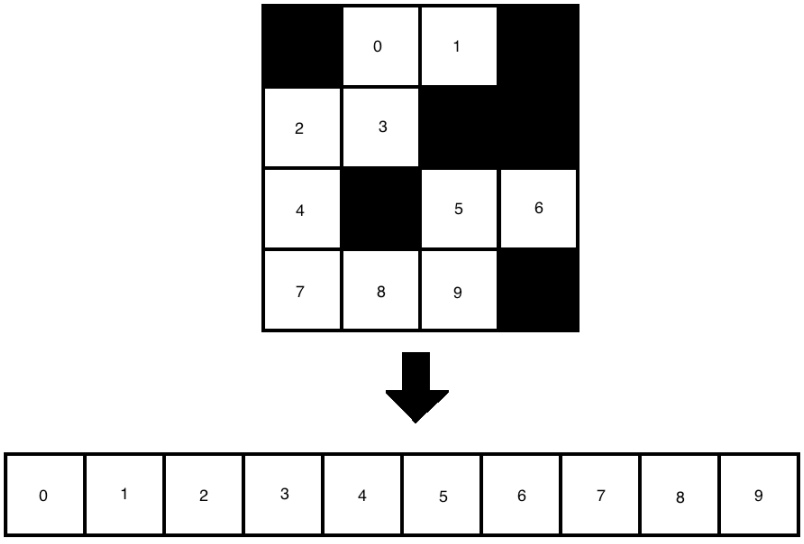


Figure 9.2.: Mapping from 2D porous medium (top, $G[i, j]$) to 1D sparse array (bottom, $S[i]$)

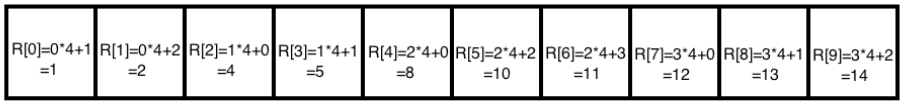


Figure 9.3.: Mapping information from 1D sparse array ($R[s]$) to 2D porous medium. $R[s]$ stores the compressed coordinates (i,j) which is used to map the 1D sparse array to its original coordinates in the 2D geometry.

9.2.2. Parallel implementation

In LB simulation, most of the operations are local, and therefore it is ideal for parallel computing. In the last 20 years, parallel machines and libraries have become widely available. These technologies make larger scale simulations possible. Several studies have been carried out to investigate the efficiency of the parallel lattice Boltzmann method. Skordos [161] compared two parallel CFD methods: finite differential method and the lattice Boltzmann method. An equal partition strategy was used in this study. He investigated the relationship between problem sizes, size of each partition, number of partitions and the performance of the code. A 50% level of efficiency is obtained if the number of processors is more than 15. Martys *et al.* [162] implemented a parallel multi-phase lattice Boltzmann code to simulate multiphase flows. A speedup slightly lower than linear was achieved.

The principal parallel machine/technology can be categorised into two groups: shared-memory machines and distributed memory machines. The shared-memory machines (Figure 9.4(a)) have multiple processors but all the processors share a single memory. As all the machines share one memory and BUS, the communication efficiency between processors is very high. However, it is very expensive to build a very large shared-memory machine. The distributed machine (Figure 9.4(b)) has multiple processors with independent memory, the communication between processors is carried out by an external network. The external network limits the latency and bandwidth of memory which might lead to poor performance of parallel computing. However, it is relatively easier and cheaper to build a large scale distributed-memory machine than a shared-memory machine. In this study,

we use a distributed-memory machine to accelerate the simulation. The optimised code is tested on a Viglen HPC system which consists of several HX525BTi Twin Blades which contain two Intel Xeon E5-2650 (Six Core 2.0GHz) Processors, 64GB of memory and the blades are connected by a QDR Infiniband Mezzanine network.

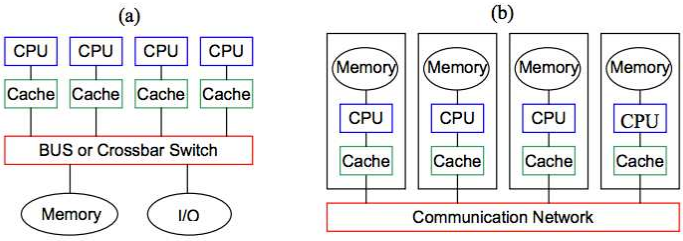


Figure 9.4.: Two typical parallel machines: (a) a shared-memory machine and (b) a distributed-memory machine [2]

Several parallel interfaces are available to implement parallel computing code, including MPI (Message Passing Interface), PVM (Parallel Virtual Machine), and HPF (High Performance Fortran). MPI is used in our parallel implementation mainly due to two reasons:

1. MPI has been developed for 20 years and is considered as a standard.
2. It is highly compatible with different types of machines which ensures the portability of the code.

To implement a parallel LB code, the computational domain is divided equally into several partitions in x direction. This is shown in Figure (9.5). Particles distribution functions (PDFs) from different partitions are illustrated with different colours. The PDFs in the dotted line area will stream to the adjacent nodes and therefore need to communicate with PDFs that are stored in another processor. An MPI subroutine is developed to exchange values for PDFs in this area.

There are two types of communication subroutines in MPI: blocking and non-blocking communication. Blocking communication (MPI_Send() and

MPI_Recv()) does not return (blocked) until the communication is finished. Receiving functions (MPI_Recv() etc.) returns when the receive buffer has been filled with valid data.

In contrast, non-blocking communication (MPI_Isend() and MPI_Irecv()) returns immediately (not blocked) even if the communication is not finished yet. MPI_Wait() or MPI_Probe() must be called to confirm the completion of communication. The blocking communication is easier to use but it blocks the processors when the communication is carried out which leads to unnecessary CPU idles. On the other hand, non-blocking functions allow overlapping of computation and communication which can generally improve computing performance.

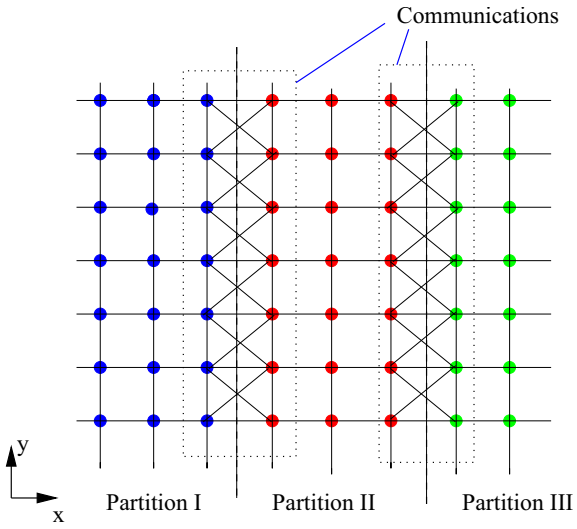


Figure 9.5.: Schematic diagram of partition geometry

In the existing implementation, the partition is only carried out in X direction but with load balance consideration. The workload between subdomains is minimised to achieve a balanced partition. The computing kernel implementation of single phase parallel LB can be summarised as:

- Local operation: collision term calculation
- Local operation: streaming to neighbouring nodes
- MPI operation: boundary nodes communications

- Local operation: distribution function update, macroscopic quantities calculation

Another potential efficient implementation of LB is using graphical processing units (GPUs). A GPU is specifically designed to be extremely fast at processing floating point operations, it can provide equivalent computing power to a small supercomputer since a larger portion of their transistors are used for floating-point arithmetic and it has a higher memory bandwidth than CPUs [163]. However it suffers several disadvantages. First of all, the code portability is poor compared to CPU code, the code should be implemented and optimised for a certain GPU structure, an update on the hardware requires corresponding updates of source code. This makes the code maintenance expensive. What's more, the memory of GPU is currently limited to 8GB, which is a relatively small value for realistic porous medium flow calculations. Although the hybrid GPU/CPU technology can overcome this shortage by using multi GPUs, the bandwidths between CPUs/GPUs limit further efficiency improvements.

9.2.3. Local operation optimisations

To fully explore how the code performs at the macro scale and investigate the efficiency of the LB code, it was profiled using a profiling tool, GNU profiler. The GNU profiler is able to produce accurate information on the time spent on every line of our code, it helps to find the functions on which the program is spending most of its execution time. Then bottleneck functions, which are called most frequently or use most execution time, should be optimised. The profiler helps developers to understand their code better and achieve a more targeted optimisation.

The LB code package was profiled and analysed carefully to target the bottleneck of the code. The profiling showed that the MPI operations use only between $7\% - 15\%$ execution time, whereas the matrix operations in the Multi-Relaxation-Time scheme use more than 30% computing time. This was identified as one of the bottlenecks of the code. According to the profiling report, the transformation matrix M and its inverse M^{-1} were found to be the cause of the low efficiency. In normal LB implementation, M and M^{-1} are defined in the initialisation part, and stored in arrays. In

the collision step, the transformation matrix value is called very frequently in the equation to calculate the collision term Ω:

$$\Omega = M^{-1}S((Mf) - M_{eq}) \tag{9.1}$$

The frequent calling for values of arrays M, M^{-1} and distribution function values f lead to massive movement of pointers between each other which significantly slows down the computing speed. The movement of pointers should be reduced to minimise the time spent on memory reading. Therefore, the value of matrix M and M^{-1} should be given explicitly in the code to avoid frequent pointer movement. It is easy to explicitly give M values, but difficult for M^{-1}. Because the structure of a modern computer is based on a binary system, the decimal format input for M^{-1} is not accurate enough for LB simulation and will cause numerical instability. Instead, a hexadecimal (hex) format of M^{-1} can be used to explicitly input array values. Numerical tests showed that this local operation optimisation can reduce 40% of the computing time which significantly improves the code performance.

9.2.4. Scaling test and results

The opimised parallel LB code is applied to study single phase flow in Bentheimer sandstone, the image of which is obtained by x-ray microtomography scanning with a resolution of $4.9\mu m$. The size of the sample is 1024x512x512. The calculation time is recorded for the scaling test. Parallel computing performance can be evaluated using Speedup (S_p):

$$S_p = \frac{t_1}{t_p} \tag{9.2}$$

where t_1 is the calculation time on a single processor, t_p is the calculation time on p processors. The efficiency E_p is defined as:

$$E_p = \frac{S_p}{p} \tag{9.3}$$

Under ideal conditions, the speedup and efficiency will be p and 1 respectively. The simulation tests were carried out on a Viglen HPC cluster, the detailed configuration of the machine was given at the beginning of this chapter. Figure 9.6 shows the calculation time for LB simulations. The

computing time decreases as the number of processors increases. Figure 9.7 and Figure 9.8 show the speedup and efficiency results. We observe very good scaling performance in the figure, as the speedup is very close to the ideal value if the number of processors is below 120. It means that the code can benefit from the increase of processor numbers on performance improvement. The speedup and efficiency are slightly lower than the ideal value for more than 120 processors. The decrease in speedup and efficiency is caused by the simple partition scheme. The simple bisectional partition scheme is used in the existing implementation. It only optimises the workload of processors, while the MPI communication load (number of nodes which need to communicate with other processors) is not taken into account in the bisectional partition scheme. A more robust partition scheme, such as the graph partitioning method can be used to further improve the parallel performance. This not only balances the workload of each processor but also minimises the communication load [164, 165, 166].

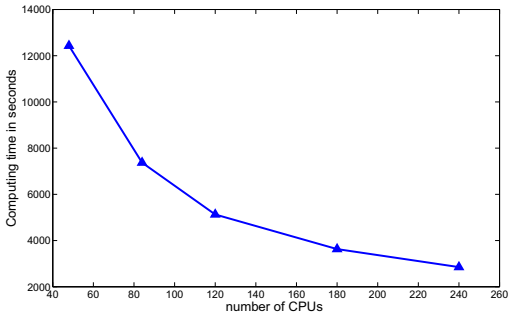


Figure 9.6.: Single phase LB simulation for Bentheimer sandstone, computing time VS number of CPUs

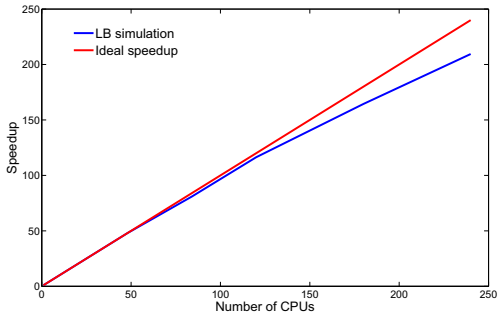


Figure 9.7.: Speedup test results (blue dotted line) compared to ideal speedup (red line)

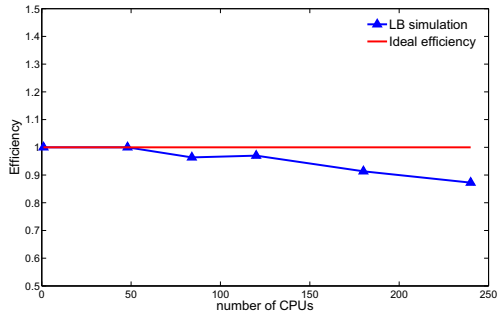


Figure 9.8.: Efficiency test results (blue dotted line) compared to ideal speedup (red line)

10. Conclusions and outlook

This PhD project has presented an optimised parallel numerical tool based on the lattice Boltzmann method for complex fluid flow simulation in porous media. A C/C++/MPI code package has been developed to simulate single phase flow, multiphase flow and thermal/solute dispersion in porous media. The representation of the porous medium is taken by x-ray microtomography scanning at voxel sizes of several microns. The code has been highly optimised in order to simulate large computing scale reservoir rock samples (sample size >1024x512x512). In this chapter, the contribution of this PhD project will be summarised and then followed by a brief discussion of the drawbacks of the model and outlook on future work.

10.1. Conclusions

This thesis presents a lattice Boltzmann model to solve pore scale complex flow in porous media. The description of the models, validations, numerical experiments on reservoir rocks and novel optimisation schemes have been presented.

The first part (Chapter 3.90-2) briefly introduced the background of this project, the importance of the study of pore scale flow and the development of numerical techniques for pore scale simulation.

The second part (Chapter 3) discussed in detail the algorithm of the lattice Boltzmann method, a brief study on boundary conditions was carried out to compare different boundary conditions in terms of accuracy, and the compatibility for porous medium flow calculations. A Multi-Relaxation-Time (MRT) collision scheme was introduced to improve the accuracy and stability of the simulation. The three most used multiphase/multi-component lattice Boltzmann methods and a thermal/solute dispersion lattice Boltz-

mann method were also discussed. Lastly, the recovery of macroscopic control equations was given showing the connection between the lattice Boltzmann method and Navier-Stokes equations.

The third part (Chapter 4) includes extensive validation examples, followed by numerical experiments to predict the transport properties of single phase flow in different types of reservoir rock samples:

- Validation tests have been carried out including channel flow, and fibrous porous media permeability prediction. The calculation results have been compared to analytical solutions and showed good agreement.
- Various boundary conditions used for permeability prediction calculation of porous media have been studied, the results were analysed and the most reliable boundary condition for porous media permeability calculation has been suggested.
- A systematic study of the effect of system size has been carried out. I find that, as the system size of the computational sub-sample increases, the values of the permeability measurements as a function of porosity tend to concentrate in a narrower region of the porosity.

In the fourth part (Chapter 6), multiphase/multi-component flow is studied extensively by validation examples and numerical experiments on micromodels and reservoir rock samples. The most relevant conclusions are:

- Various validation simulations have been performed including calculation of capillary pressure, and relative permeability. The results have been compared to analytical solutions and showed good agreement.
- A comparison study of three different multi-component Lattice Boltzmann models was carried out to explore their capability of describing binary immiscible fluid systems in porous media. Poiseuille flow of layered immiscible binary fluids and capillary fingering phenomena were investigated and the results were evaluated against analytical solutions. In addition, the capability of the various models to simulate fluids with significant viscosity and density contrast and suitable

interface thickness was systematically studied. The Colour Gradient model is found the most efficient model for binary immiscible flow in porous media due to its thin interfaces, capability for high viscosity ratio and high level of numerical stability.

- The spontaneous imbibition of a decane/air system in two sets of micro-models with increasing complexity: single junction models with equal, unequal arms and a pore structure based on actual Berea sandstone were studied. The simulation results were compared to corresponding micro fluidic experiments. The simulations were successfully validated against the micro-fluidic experiments on the displacement processes of imbibition and primary drainage.

- The relative permeability of realistic reservoir rock samples was studied. Two relative permeability prediction schemes were introduced and compared to available experimental data. The simulation results were generally in good agreement with the experimental data. However, some small discrepancies of non-wetting phase relative permeability were found in the imbibition cycle. These discrepancies might be caused by limited mesh resolution. The accuracy can be improved by using a finer resolution which will, however, increase the computing time significantly.

- Residual cluster sizes in porous medium, caused by capillary trapping, were studied using a multi-component lattice Boltzmann method. The simulations were carried out to simulate the generation of non-wetting residual clusters. The shape was analysed and the size distribution was calculated and compared to corresponding experimental data. The simulation generally agrees with the experimental findings although parts of very small and large residual clusters were not captured due to the limited resolution of the mesh.

Chapter 7 looked at solute/heat transfer diffusivity in porous media using the lattice Boltzmann method. The method was firstly validated by convection-diffusion flow in a cavity and the simulation results were compared to experimental data. Then flow and solute transport were simulated directly on pore space images of rock cores with an increasing degree of heterogeneity: a bead pack, Bentheimer sandstone and Portland carbonate.

A novel scheme is proposed to predict probability distributions for molecular displacements using the lattice Boltzmann method to calculate both the flow field and solute dispersion. The results were compared directly with NMR experimental data for molecular displacements and excellent agreement was found. The fraction of solute particles trapped by integrating over the stagnant peak were calculated to quantify the degree of heterogeneity.

In the last part of the thesis (Chapter 9), issues of computing were addressed. A special sparse storage method was designed for porous media, parallel operation and local collision optimisations were introduced, and a scaling test was carried out to analyse the efficiency of the algorithm. In general, the optimised lattice Boltzmann implementation has a very good efficiency, as close to ideal scalability was observed. However, the efficiency decreases after 120 CPUs, which is caused by the application of a simple bisectional partition scheme. The graph partitioning method [165, 166] which not only minimises the local workload but also minimises the communication load is suggested for future implementation.

10.2. Outlook and future work

The present work can be applied to study reaction in porous media, for example by adding a random walk particle tracking module based on the random walk method. The lattice Boltzmann method calculates the flow field and the random walk module accounts for the convection and dispersion. A proper reaction model might enable the simulation to predict the precipitation and dissolution in porous media which is of great importance for both carbon storage and the oil and gas industry.

The present model is not of high efficiency for multiphase flow calculation in porous media, which is caused by limited mesh resolution near small pores and throats. These small pores and throats also lead to high capillary pressure and high spurious velocity which significantly slow down the simulation and affect the numerical stability. Adaptive mesh refinement may be one solution for these problems. The location of droplets interfaces are identified and the local mesh refinement is carried out near interfaces. The fine mesh for multiphase simulation can significantly reduce the spurious

velocity and improve the numerical stability [167]. However this adaptive scheme will be an expensive scheme, the positioning of the droplets interfaces and local refinement need complex algorithms and also increase the MPI parallel operation workload.

Another possible solution is developing an algorithm that can predict the drainage and imbibition without solving the full multiphase flow problem. The new algorithm should only account for the local capillary pressure and the local velocity should not be taken into account. The velocity field from a single phase calculation can be used to approximate the local velocity in capillary pressure simulation. A proper method need to be developed to predict the local velocity using the reference velocity in single phase calculation. This scheme avoid the velocity field update which is the most expensive computing part in capillary pressure measurement. Therefore it should be a ideal method for the capillary pressure prediction for realistic reservoir rocks.

In terms of the implementation, the graph partitioning method is suggested for future implementation. This scheme sets the boundaries at the throats which not only minimises the computing workload of the cores but also minimises the communication load. This is of significant importance for the parallel efficiency of large scale computing.

Bibliography

[1] X. Shan. Analysis and reduction of the spurious current in a class of multiphase lattice Boltzmann models. *Phys. Rev. E*, 73(4):047701, Apr 2006.

[2] Y. Keehm. Computational rock physics: Transport properties in porous media and applications. *PhD thesis*, 2003.

[3] T. Ramstad, P.E. Øren, and S. Bakke. Simulation of two-phase flow in reservoir rocks using a lattice boltzmann method. *SPE J*, 15(4):917–927, 2010.

[4] S. Iglauer, M.A. Fern, P. Shearing, and M.J. Blunt. Comparison of residual oil cluster size distribution, morphology and saturation in oil-wet and water-wet sandstone. *J. Colloid. Interf. Sci.*, 375(1):187 – 192, 2012.

[5] N.C. Markatos and K.A. Pericleous. Laminar and turbulent natural convection in an enclosed cavity. *Int. J. Heat Mass Transf.*, 27(5):755–772, 1984.

[6] M. C. Sukop and D. T. Thorne. *Lattice Boltzmann Modeling: An introduction for geoscientists and engineers*. Springer Verlag, 2007.

[7] U.M. Scheven, D. Verganelakis, R. Harris, M.L. Johns, and L.F. Gladden. Quantitative nuclear magnetic resonance measurements of preasymptotic dispersion in flow through porous media. *Phys. Fluids*, 17(11):117107–1, 2005.

[8] J.B.Walsh and W. F. Brace. The effect of pressure on porosity and the transport properties of rock. *J. Geophys. Res.*, 89:9245–9431, 1984.

[9] J.Dvorkin G.Mavko, T.Mukerji. *The rock physics handbook*. Cambridge University Press, New York, 2006.

[10] Ingrain. http://www.ingrainrocks.com/multiphase-flow/.

[11] D.Kandhai, A.Koponen, and A.G.Hoekstra. Lattice-Boltzmann hydrodynamics on parallel systems. *Comput. Phys. Commun.*, 111:14–26, 1998.

[12] R.B.Saeger, L.E.Scriven, and H.T.Davis. Transport processes in periodic media. *J. Fluid Mech.*, 299:1–15, 1995.

[13] A.J.C.Ladd. Numerical simulations of particulate suspensions via a discretized Boltzmann equation: Part 1. theoretical foundation. *Phys. Rev. Lett.*, 271:285–309, 1994.

[14] P.M.Adler, C.G.Jacquin, and J.A.Quiblier. Flow in simulated porous media. *Int. J. Multiphase Flow*, 16:691–712, 1990.

[15] I.Fatt. The network model of porous media: I. capillary pressure characteristics. *Pet. Tran. AIME*, 207:144–159, 1967.

[16] H. Dong and M. J. Blunt. Pore-network extraction from micro-computerized-tomography images. *Phys. Rev. E*, 80(3):036307, 2009.

[17] M.J.Blunt. Flow in porous media – pore-network models and multiphase flow. *Curr Opin Colloid In*, 6(3):197 – 207, 2001.

[18] P.King M.Blunt. Relative permeabilities from two- and three-dimensional pore-scale network modelling. *Transport Porous Med*, 6(4):407–433, 1991. 10.1007/BF00136349.

[19] M.A.Knackstedt, C.H.Arns, and A.P.Sheppard. Pore scale analysis of electrical resistivity in complex core material. *Proc. Of the Soc. Core Analysts, Calgary, Canada*, 2007.

[20] A.Cancelliere, C.Chang, and E.Foti. The permeability of a random medium: Comparison of simulation with theory. *Phys. Fluids*, 2:2085, 1990.

[21] D.Silin and T.Patzek. Pore space morphology analysis using maximal inscribed spheres. *Physica A*, 371:336–360, 2006.

[22] R.E.Larson and J.Hipon. A periodic grain consolidation model of porous media. *Phys. Fluids A*, 1:38–46, 1988.

[23] G.D. Doolen. *Lattice gas methods for partial differential equations: a volume of lattice gas reprints and articles, including selected papers from the workshop on large nonlinear systems, held August, 1987 in Los Alamos, New Mexico.* Santa Fe Institute studies in the sciences of complexity. Addison-Wesley, 1990.

[24] S. Chen, Z. Wang, X. Shan, and G. D. Doolen. Lattice Boltzmann computational fluid dynamics in three dimensions. *J. Stat. Phys.*, 68(3/4):379–400, 1992. 10.1007/BF01341754.

[25] Y.Pomeau U.Frisch, B.Hasslacher. Lattice-gas automata for the navier stokes equation. *Phys. Rev. Lett.*, 56:1505–1508, 1986.

[26] H.Chen, S.Chen, and W.H.Matthaeus. Recovery of the navier-stokes equations using a lattice-gas Boltzmann method. *Phys. Rev. A.*, 45:5339–5342, 1992.

[27] D. d'Humires, P.Lallemand, and U.Frisch. Lattice-gas models for 3d hydrodynamics. *Europhys. Lett.*, 2:291–297, 1986.

[28] F.Higuera, S.Succi, and R.Benzi. Lattice gas dynamics with enhanced collisions. *Europhys. Lett.*, 9:345–349, 1989.

[29] Y. H. Qian, D. d'Humières, and P. Lallemand. Lattice BGK models for Navier-Stokes equation. *Europhys. Lett.*, 17(6):479–484, 1992.

[30] Y. H. Qian and Y. F. Deng. A lattice bgk model for viscoelastic media. *Phys. Rev. Lett.*, 79:2742–2745, 1997.

[31] D d'Humières. Generalised lattice Boltzmann equations. *Progr. Astronaut Aero.*, 159:450–458, 1992.

[32] D. d'Humières, I. Ginzburg, and M. Krafczyk. Multiple-relaxation-time lattice Boltzmann models in three dimensions. *Phil. Trans. R. Soc*, 360:437–451, 2001.

[33] P. Lallemand and L.-S. Luo. Theory of the lattice Boltzmann method: Dispersion, dissipation, isotropy, galilean invariance and stability. *Phys. Rev. E*, 61:6546, 2000.

[34] Andrew K. Gunstensen, Daniel H. Rothman, Stéphane Zaleski, and Gianluigi Zanetti. Lattice Boltzmann model of immiscible fluids. *Phys. Rev. A*, 43(8):4320–4327, Apr 1991.

[35] D. Grunau, S. Chen, and K. Eggert. A lattice Boltzmann model for multiphase fluid flows. *Phys. Fluids A*, 5(10):2557–2562, 1993.

[36] J. Tölke, M. Krafczyk, M. Schulz, and E. Rank. Lattice Boltzmann simulations of binary fluid flow through porous media. *Phil. Trans. R. Soc. Lond. A*, 360:535, March 2002.

[37] U. D'Ortona, D. Salin, Marek Cieplak, Renata B. Rybka, and Jayanth R. Banavar. Two-color nonlinear Boltzmann cellular automata: Surface tension and wetting. *Phys. Rev. E*, 51(4):3718–3728, Apr 1995.

[38] Michael R. Swift, E. Orlandini, W. R. Osborn, and J. M. Yeomans. Lattice Boltzmann simulations of liquid-gas and binary fluid systems. *Phys. Rev. E*, 54(5):5041–5052, Nov 1996.

[39] C. M. Pooley and K. Furtado. Eliminating spurious velocities in the free-energy lattice Boltzmann method. *Phys. Rev. E*, 77(4):046702, Apr 2008.

[40] Xiaowen Shan and Hudong Chen. Lattice Boltzmann model for simulating flows with multiple phases and components. *Phys. Rev. E*, 47(3):1815–1819, Mar 1993.

[41] A. L. Kupershtokh, D. A. Medvedev, and D. I. Karpov. On equations of state in a lattice Boltzmann method. *Comput. Math. Appl.*, 58(5):965–974, 2009.

[42] A. L. Kupershtokh, D. I. Karpov, D. A. Medvedev, C. P. Stamatelatos, V. P. Charalambakos, E. C. Pyrgioti, and D. P. Agoris. Stochastic models of partial discharge activity in solid and liquid dielectrics. *IET Sci. Meas. Technol.*, 1(6):303–311, 2007.

[43] G. Jin, T. W. Patzek, and D. B. Silin. Direct prediction of the absolute permeability of unconsolidated and consolidated reservoir rocks. *paper SPE*, 90084:26–29, 2004.

[44] C. Pan, L.-S. Luo, and C. T. Miller. An evaluation of lattice Boltzmann schemes for porous medium flow simulation. *Comput. Fluids*, 35:898–909, 2006.

[45] C. Pan, M. Hilpert, and CT Miller. Lattice-Boltzmann simulation of two-phase flow in porous media. *Water Resour. Res.*, 40(1):W01501, 2004.

[46] T. Ramstad. Simulation of two phase flow in reservoir rocks using a lattice Boltzmann method. *paper SPE*, 124617, 2009.

[47] P. A. Skordos. Parallel simulation of subsonic fluid dynamics on a cluster of workstations. *High Performance Distributed Computing 95, 4th IEEE Int. Symp*, 1995.

[48] P. Giovanni, F. Massaioli, and S. Succi. High-resolution lattice-Boltzmann computing on the IBM SP1 scalable parallel computer. *Comput. Phys.*, 8:705, 1994.

[49] G. Amati, S. Succi, and R. Piva. Massively parallel lattice-Boltzmann simulation of turbulent channel flow. *J. Mod. Phys.*, C 8:869, 1997.

[50] D.R.Noble, S.Chen, and J.G.Georgiadis. A consistent hydrodynamic boundary condition for the lattice-Boltzmann method. *Phys. Fluids*, 7:203, 1995.

[51] J.Ma, K.Wu, and Z.Jiang. Shift:an implementation for lattice Boltzmann simulation in low-porosity porous media. *Phys. Rev. E*, 81:056702, 2010.

[52] G.A. Bird. *Molecular Gas Dynamics and the Direct Simulation of Gas Flows*. Oxford science publications. Oxford University Press, Incorporated, 1994.

[53] P.L. Bhatnagar, E.P. Gross, and M. Krook. A model for collision processes in gases. *Phys. Rev.*, 94:511–525, 1954.

[54] D. A. Wolf-Gladrow. *Lattice-gas cellular automata and lattice Boltzmann models*. Lecture notes in mathematics. Springer, 2000.

[55] T. Inamuro, M. Yoshino, and F. Ogino. A non-slip boundary condition for lattice-Boltzmann simulations. *Phys. Fluids*, 7(12):2928–2930, 1995.

[56] M.Breure, J.Bernsdorf, and T.Zeiser. Accurate computations of the laminar ow past a square cylinder based on two different methods: lattice-Boltzmann and finite-volume. *Int. J. Heat Fluid Flow*, 21:186–196, 2000.

[57] D.P. Ziegler. Boundary conditions for lattice-Boltzmann simulations. *J. Stat. Phys.*, 71:1171–1177, 1993.

[58] Z. Guo, C. Zheng, and B. Shi. Discrete lattice effects on the forcing term in the lattice Boltzmann method. *Phys Rev. E*, 65(4):046308, 2002.

[59] A. L. Kupershtokh. New method of incorporating a body force term into the lattice Boltzmann equation. In *Proc. 5th International EHD Workshop, University of Poitiers, Poitiers, France*, pages 241–246, 2004.

[60] Xiaowen Shan and Gary Doolen. Multicomponent lattice-Boltzmann model with interparticle interaction. *J. Stat. Phys.*, 81(1):379–393, 1995.

[61] H. Huang, M. Krafczyk, and X. Lu. Forcing term in single-phase and Shan-Chen-type multiphase lattice Boltzmann models. *Phys Rev. E*, 84(4):046710, 2011.

[62] Z. Yu and L. S. Fan. An interaction potential based lattice Boltzmann method with adaptive mesh refinement (AMR) for two-phase flow simulation. *J. Comp. Phys.*, 228(17):6456–6478, 2009.

[63] Z. Yu. *A Novel Lattice Boltzmann Method for Direct Numerical Simulation of Multiphase Flows.* PhD thesis, The Ohio State University, 2009.

[64] D. Kehrwald. *Numerical analysis of immiscible lattice BGK.* PhD thesis, Dissertation, Fachbereich Mathematik, Universität Kaiserslautern, 2002.

[65] J. Tölke, S. Freudiger, and M. Krafczyk. An adaptive scheme using hierarchical grids for lattice Boltzmann multi-phase flow simulations. *Comput. Fluids*, 35(8):820–830, 2006.

[66] B. Ahrenholz, J. Tölke, P. Lehmann, A. Peters, A. Kaestner, M. Krafczyk, and W. Durner. Prediction of capillary hysteresis in a porous material using lattice-Boltzmann methods and comparison to experimental data and a morphological pore network model. *Adv. Water Resour.*, 31(9):1151–1173, 2008.

[67] J. Tölke. *Gitter-Boltzmann-Verfahren zur Simulation von Zweiphasenströmungen*. PhD thesis, TU München, 2001.

[68] M. Yoshino and T. Inamuro. Lattice Boltzmann simulations for flow and heat/mass transfer problems in a three-dimensional porous structure. *Int. J. Numer. Methods Fluids*, 43(2):183–198, 2003.

[69] T. Inamuro, M. Yoshino, H. Inoue, R. Mizuno, and F. Ogino. A lattice Boltzmann method for a binary miscible fluid mixture and its application to a heat-transfer problem. *J. Comput. Phys.*, 179(1):201 – 215, 2002.

[70] Takaji Inamuro, Masato Yoshino, and Fumimaru Ogino. Accuracy of the lattice Boltzmann method for small knudsen number with finite reynolds number. *Phys. Fluids*, 9(11):3535–3542, 1997.

[71] H. Kusumaatmaja. *Lattice Boltzmann Studies of Wetting and Spreading on Patterned Surfaces*. PhD thesis, University of Oxford, 2008.

[72] S. Succi. The lattice Boltzmann equation. *For Fluid Dynamics and Beyond*, 2001.

[73] J.M. Buick and C.A. Greated. Gravity in a lattice Boltzmann model. *Phys. Rev. E*, 61(5):5307, 2000.

[74] Anton W. J. Heijs and Christopher P. Lowe. Numerical evaluation of the permeability and the kozeny constant for two types of porous media. *Phys. Rev. E*, 51:4346–4352, 1995.

[75] F. M. Auzerais, J. Dunsmuir, B. B. Ferreol, N. Martys, J. Olson, T.S. Ramakrishnan, D.H. Rothman, and L.M. Schwartz. Transport

in sandstone: A study based on three-dimensional microtomography. *Geophys. Research Lett.*, 23(7):705–708, 1996.

[76] Q. Kang, D. Zhang, and S. Chen. Unified lattice-Boltzmann method for flow in multiscale porous media. *Phys. Rev. E*, 66:056307, 2002.

[77] J. Chin, E. S. Boek, and P. V. Coveney. Lattice Boltzmann simulation of the flow of binary immiscible fluids with different viscosities using the Shan-Chen microscopic interaction model. *Phil. Trans. R. Soc. Lond. A*, pages 547–558, 2002.

[78] A. Sengupta, P.S. Hammond, D. Frenkel, and E.S. Boek. Error analysis and correction for lattice Boltzmann simulated flow conductance in capillaries of different shapes and alignments. *J. Comput. Phys.*, 2011.

[79] Z.L.Yang, T.N.Dinh, and R.R.Nourgaliev. Evaluation of the Darcy's law performance for two-fluid flow hydrodynamics in a particle debris bed using a lattice-Boltzmann model. *Heat and Mass Transfer*, 36:295–304, 2000.

[80] F.J.Phelan and G.Wise. Analysis of transverse flow in aligned fibrous porous media. *Compos. Part A: Appl. Sci. Manufac.*, 27:25–34, 1996.

[81] R. Cornubert, D. d'Humieres, and D. Lavermore. A Knudsen layer theory for lattice gases. *Physica D*, 47:241–259, 1991.

[82] D. P. Ziegler. Boundary conditions for lattice Boltzmann simulations. *J. Stat. Phys.*, 71:1171–1177, 1993.

[83] I. Ginzbourg and P.M. Adler. Boundary flow condition analysis for the three-dimensional lattice Boltzmann model. *J. Phys. II France*, 4:191–214, 1994.

[84] Q. Zou, S. Hou, and G. D. Doolen. Analytical solutions of the lattice Boltzmann BGK model. *J. Stat. Phys.*, 81:319–334, 1995.

[85] X. He, Q. Zou, L.-S. Luo, and M. Dembo. Analytic solution of simple flows and analysis of non-slip boundary conditions for the lattice Boltzmann BGK model. *J. Stat. Phys.*, 87(1-2):115–136, 1997.

[86] T. Inamuro, M. Yoshino, and F. Ogino. A non-slip boundary condition for lattice Boltzmann simulations. *Phys. Fluids*, 7(12):2928–2930, 1995.

[87] D. R. Noble, S. Chen, J. G. Georgiadis, and R. O. Buckius. A consistent hydrodynamic boundary condition for the lattice Boltzmann method. *Phys. Fluids*, 7:203–209, 1995.

[88] S. Chen, D. Martinez, and R. Mei. On boundary conditions in lattice Boltzmann methods. *Phys. Fluids*, 8:2527–2536, 1996.

[89] R. Verberg and A.J.C. Ladd. Accuracy and stability of a LB model with subgrid scale boundary conditions. *Phys. Rev. E*, 65:016701, 2001.

[90] N.S.Martys and H.Chen. Simulation of multicomponent fluids in complex three-dimensional geometries by the lattice Boltzmann method. *Phys. Rev. E*, 53(1):743–750, 1996.

[91] M. Venturoli and E. S. Boek. Two-dimensional lattice Boltzmann simulations of single phase flow in a pseudo two-dimensional micromodel. *Physica A*, 362:23–29, 2006.

[92] B. Ferrèol and D. H. Rothman. Lattice-Boltzmann simulations of flow through Fontainebleau sandstone. *Trans. Porous Media*, 20:3–20, 1995.

[93] A. Koponen, D. Kandhai, E. Hellen, M. Alava, A. G. Hoekstra, M. Kataja, K. Niskanen, P. M. A. Sloot, and J. Timonen. Permeability of three-dimensional random fiber webs. *Phys. Rev. Lett.*, 80(4):716–719, 1998.

[94] C. Manwart, U. Aaltosalmi, A. Koponen, R. Hilfer, and J. Timonen. Lattice Boltzmann and finite-difference simulations for the permeability for three-dimensional porous media. *Phys. Rev. E*, 66, 2002.

[95] E.S. Boek and M. Venturoli. Lattice-boltzmann studies of fluid flow in porous media with realistic rock geometries. *Comput. Math. Appl.*, 59:2305–2314, 2010.

[96] P. M. Adler, C. G. Jacquin, and J. A. Quiblier. Flow in simulated porous media. *Int. J. Multiphase Flow*, 16:691, 1990.

[97] J.-F. Thovert, F. Yousefian, P. Spanne, C. G. Jacquin, and P. M. Adler. Grain reconstruction of porous media: Application to a low-porosity fontainebleau sandstone. *Phys. Rev. E*, 63:061307, 2001.

[98] O. Gerbaux, F. Buyens, V. V. Mourzenko, A. Memponteil, A. Vabre, J.-F. Thovert, and P. M. Adler. Transport properties of real metallic foams. *J. Colloid Interf. Sci.*, 342:155–165, 2010.

[99] J. Bear. *Dynamics of fluids in porous media*. Dover publications, 1988.

[100] D. Halpern and D. P. Gaver. Boundary element analysis of the time-dependent motion of a semi-infinite bubble in a channel. *J. Comp. Phys.*, 115(2):366 – 375, 1994.

[101] P. Yuan and L. Schaefer. Equations of state in a lattice Boltzmann model. *Phys. Fluids*, 18:042101, 2006.

[102] X. Shan. Simulation of Rayleigh-Bénard convection using a lattice Boltzmann method. *Phys Rev. E*, 55(3):2780–2788, 1997.

[103] Z.Guo and C.Zheng. *Theory and Applications of Lattice Boltzmann Method*. Beijing:Science Press, 2008.

[104] E.S. Boek and M. Venturoli. Lattice-Boltzmann studies of fluid flow in porous media with realistic rock geometries. *Comput. Math. Appl.*, 59(7):2305–2314, 2010.

[105] G. Mason and N.R. Morrow. Capillary behavior of a perfectly wetting liquid in irregular triangular tubes. *J. Colloid. Interf. Sci.*, 141(1):262–274, 1991.

[106] C. Marle. *Multiphase flow in porous media*. Éditions technip, 1981.

[107] D. G. Avraam and A. C. Payatakes. Flow mechanisms, relative permeabilities, and coupling effects in steady-state two-phase flow through porous media. the case of strong wettability. *Ind. Eng. Chem. Res.*, 38(3):778–786, 1999.

[108] T. Ramstad, N. Idowu, C. Nardi, and P.E. Øren. Relative permeability calculations from two-phase flow simulations directly on digital images of porous rocks. *Transport Porous Med.*, 94(2):487–504, 2012.

[109] D.G. Longeron. Influence of very low interfacial tensions on relative permeability. *Old SPE Journal*, 20(5):391–401, 1980.

[110] H. Huang, Z. Li, S. Liu, and X. Lu. Shan-and-chen-type multiphase lattice Boltzmann study of viscous coupling effects for two-phase flow in porous media. *Int. J. Numer. Methods Fluids*, 61(3):341–354, 2009.

[111] H.L. Goldsmith and S.G. Mason. The flow of suspensions through tubes. ii. single large bubbles. *J. Colloid Sci.*, 18(3):237–261, 1963.

[112] R. Lenormand, C. Zarcone, and A. Sarr. Mechanism of the displacement of one fluid by another in a network of capillary ducts. *J. Fluid Mech.*, 135:337, 1983.

[113] Jill S Buckley. Multiphase displacements in micromodels. *Interfacial Phenomena in Petroleum Recovery*, 36:157–189, 1991.

[114] N. González-Segredo, M. Nekovee, and P. V. Coveney. Three-dimensional lattice-Boltzmann simulations of critical spinodal decomposition in binary immiscible fluids. *Phys. Rev. E*, 67(046304), 2003.

[115] MM Honarpour, F Koederitz, and A Herbert. Relative permeability of petroleum reservoirs. 1986.

[116] D. Tiab and E.C. Donaldson. *Petrophysics: theory and practice of measuring reservoir rock and fluid transport properties*. Gulf professional publishing, 2011.

[117] S.E. Buckley and M.C. Leverett. Mechanism of fluid displacement in sands. *Trans. AIME*, 146(107):1–7, 1942.

[118] M.N. Ibrahim and L.F. Koederitz. Two-phase steady-state and unsteady-state relative permeability prediction models. *SPE 68065*, 2001.

[119] E.F. Johnson, D.P. Bossler, and V.O. Naumann. Calculation of relative permeability from displacement experiments. *Trans. AIME*, 216:370–372, 1959.

[120] F.A.L. Dullien. *Porous media: fluid transport and pore structure*, volume 2. Academic press San Diego, 1992.

[121] J. Yang and E. S. Boek. A comparison study of multi-component lattice boltzmann models for flow in porous media applications. *Comput. Math. Appl.*, 65(6):882 – 890, 2013.

[122] B. Metz, O. Davidson, H.C. De Coninck, M. Loos, and L.A. Meyer. IPCC special report on carbon dioxide capture and storage: Prepared by working group III of the intergovernmental panel on climate change. *IPCC, Cambridge University Press: Cambridge, United Kingdom and New York, USA*, 2, 2005.

[123] C. H. Pentland, R. El-Maghraby, S. Iglauer, and M. J. Blunt. Measurements of the capillary trapping of super-critical carbon dioxide in Berea sandstone. *Geophys. Res. Lett.*, 38(6):L06401, 2011.

[124] C. D. Lorenz and R. M. Ziff. Precise determination of the bond percolation thresholds and finite-size scaling corrections for the sc, fcc, and bcc lattices. *Phys. Rev. E*, 57(1):230, 1998.

[125] S. Iglauer, S. Favretto, G. Spinelli, G. Schena, and M. J. Blunt. X-ray tomography measurements of power-law cluster size distributions for the nonwetting phase in sandstones. *Phys. Rev. E*, 82(5):056315, 2010.

[126] M. Blunt, M. J. King, and H. Scher. Simulation and theory of two-phase flow in porous media. *Phys. Rev. A*, 46:7680–7699, Dec 1992.

[127] I. Chatzis, N. Morrow, and H. Lim. Magnitude and detailed structure of residual oil saturation. *Old SPE Journal*, 23(2):311–326, 1983.

[128] S. Iglauer, A. Paluszny, C. H. Pentland, and M. J. Blunt. Residual CO2 imaged with x-ray microtomography. *Geophys. Res. Lett.*, 38(21), 2011.

[129] M. Prodanović, W.B. Lindquist, and R.S. Seright. 3D image-based characterization of fluid displacement in a berea core. *Adv. Water Resour.*, 30(2):214–226, 2007.

[130] Z. T. Karpyn, M. Piri, and G. Singh. Experimental investigation of trapped oil clusters in a water-wet bead pack using x-ray microtomography. *Water Resour. Res.*, 46(4):W04510, 2010.

[131] R. I. Al-Raoush. Impact of wettability on pore-scale characteristics of residual nonaqueous phase liquids. *Environ. Sci. Technol.*, 43(13):4796–4801, 2009.

[132] C.J. Landry, Z.T. Karpyn, and M. Piri. Pore-scale analysis of trapped immiscible fluid structures and fluid interfacial areas in oil-wet and water-wet bead packs. *Geofluids*, 11(2):209–227, 2011.

[133] M. J. Blunt and H. Scher. Pore-level modeling of wetting. *Phys. Rev. E*, 52(6):6387, 1995.

[134] P.M. Adler. *Porous media: Geometry and transports*. Butterworth-Heinemann, 1992.

[135] B. Bijeljic and M.J. Blunt. Pore-scale modeling and continuous time random walk analysis of dispersion in porous media. *Water Resour. Res.*, 42(1):W01202, 2006.

[136] C. Bruderer and Y. Bernabé. Network modeling of dispersion: Transition from taylor dispersion in homogeneous networks to mechanical dispersion in very heterogeneous ones. *Water Resour. Res.*, 37(4):897–908, 2001.

[137] B. Bijeljic, A.H. Muggeridge, and M.J. Blunt. Pore-scale modeling of longitudinal dispersion. *Water Resour. Res.*, 40(11):W11501, 2004.

[138] B. Bijeljic and M.J. Blunt. Pore-scale modeling of transverse dispersion in porous media. *Water Resour. Res.*, 43(12):W12S11, 2007.

[139] R.C. Acharya, S. Van der Zee, and A. Leijnse. Approaches for modeling longitudinal dispersion in pore-networks. *Adv. Water Resour.*, 30(2):261–272, 2007.

[140] M. Knackstedt, C. Arns, A. Ghous, A. Sakellariou, T. Senden, A. Sheppard, R. Sok, H. Averdunk, V. Pinczewski, G. Padhy, and M. Ioannidis. 3D imaging and flow characterization of the pore space of carbonate core samples. *Society of Core Analyists*, 23, 2006.

[141] D. Coelho, J.F. Thovert, and P.M. Adler. Geometrical and transport properties of random packings of spheres and aspherical particles. *Phys. Rev. E*, 55:1959–1978, 1997.

[142] P.M. Adler and J.F. Thovert. Real porous media: Local geometry and macroscopic properties. *Appl. Mech. Rev.*, 51:537, 1998.

[143] R.S. Maier, D.M. Kroll, Y.E. Kutsovsky, H.T. Davis, and R.S. Bernard. Simulation of flow through bead packs using the lattice boltzmann method. *Phys. Fluids*, 10:60, 1998.

[144] U. M. Scheven, R. Harris, and M. L. Johns. Intrinsic dispersivity of randomly packed monodisperse spheres. *Phys. Rev. Lett.*, 99:054502, 2007.

[145] B. Bijeljic, P. Mostaghimi, and M.J. Blunt. Signature of non-fickian solute transport in complex heterogeneous porous media. *Phys. Rev. Lett.*, 107(20):204502, 2011.

[146] D. F. Young, B. R. Munson, T. H. Okiishi, and W. W. Huebsch. *A brief introduction to fluid mechanics*. Wiley. com, 2010.

[147] F. M. White. *Viscous Fluid Flow (3rd. ed.)*. Tata McGraw-Hill Education, 1974.

[148] M. Hortmann, M. Perić, and G. Scheuerer. Finite volume multigrid prediction of laminar natural convection: Bench-mark solutions. *Int. J. Numer. Methods Fluids*, 11(2):189–207, 1990.

[149] G. de Vahl Davis. Natural convection of air in a square cavity: a bench mark numerical solution. *Int. J. Numer. Methods Fluids*, 3(3):249–264, 1983.

[150] G. Taylor. Conditions under which dispersion of a solute in a stream of solvent can be used to measure molecular diffusion. *Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences*, pages 473–477, 1954.

[151] R. Aris. On the dispersion of a solute in a fluid flowing through a tube. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, 235(1200):67–77, 1956.

[152] H.W. Stockman, C. Li, and J.L. Wilson. A lattice-gas and lattice Boltzmann study of mixing at continuous fracture junctions: Importance of boundary conditions. *Geophys. Res. Lett.*, 24(12):1515–1518, 1997.

[153] I. Ginzburg and D. d'Humieres. Multireflection boundary conditions for lattice Boltzmann models. *Phys Rev E*, 68:066614, 2003.

[154] L. Talon, D. Bauer, N. Gland, S. Youssef, H. Auradou, and I. Ginzburg. Assessment of the two relaxation time lattice-boltzmann scheme to simulate stokes flow in porous media. *Water Resour. Res.*, 48(4), 2012.

[155] H.W. Stockman, R.J. Glass, C. Cooper, and H. Rajaram. Accuracy and computational efficiency in 3D dispersion via lattice-boltzmann: models for dispersion in rough fractures and double-diffusive fingering. *Int. J. Mod. Phys. A*, 9(8):1545–1558, 1998.

[156] L. Li, R. Mei, and J.F. Klausner. Boundary conditions for thermal lattice Boltzmann equation method. *J. Comput. Phys.*, 237:366–395, 2013.

[157] H. Yoshida and M. Nagaoka. Multiple-relaxation-time lattice Boltzmann model for the convection and anisotropic diffusion equation. *J. Comput. Phys.*, 229(20):7774–7795, 2010.

[158] N. Otsu. A threshold selection method from gray-level histograms. *Automatica*, 11(285-296):23–27, 1975.

[159] R. W. Mair, G. P. Wong, D. Hoffmann, M. D. Hürlimann, S. Patz, L. M. Schwartz, and R. L. Walsworth. Probing porous media with gas diffusion NMR. *Phys. Rev. Lett.*, 83:3324–3327, 1999.

[160] S. N. Ehrenberg and P. H. Nadeau. Sandstone vs. carbonate petroleum reservoirs: A global perspective on porosity-depth and porosity-permeability relationships. *AAPG bulletin*, 89(4):435–445, 2005.

[161] P. A. Skordos. Modeling flue pipes: Subsonic flow, lattice Boltzmann, and parallel distributed computers. Technical report, DTIC Document, 1995.

[162] N. S. Martys, J. G. Hagedorn, D. Goujon, and J. E. Devaney. Large-scale simulations of single-and multicomponent flow in porous media. In *SPIE's International Symposium on Optical Science, Engineering, and Instrumentation*, pages 205–213. International Society for Optics and Photonics, 1999.

[163] Eirik O Aksnes and Anne C Elster. Porous rock simulations and lattice boltzmann on gpus. In *PARCO*, pages 536–545, 2009.

[164] G. Karypis and V. Kumar. Metis-unstructured graph partitioning and sparse matrix ordering system, version 2.0. 1995.

[165] G. Karypis and V. Kumar. Multilevel k-way partitioning scheme for irregular graphs. *J. Parallel Distr. Com.*, 48(1):96–129, 1998.

[166] G. Karypis and V. Kumar. Multilevel algorithms for multi-constraint graph partitioning. In *Proceedings of the 1998 ACM/IEEE conference on Supercomputing (CDROM)*, pages 1–13. IEEE Computer Society, 1998.

[167] J.Tölke, S.Freudiger, and M.Krafczyk. An adaptive scheme for LBE multiphase flow simulations on hierarchical grids. *Chem. Eng. Sci.*, 35:820–830, 2006.

A. Conversion of lattice units

In lattice Boltzmann simulations, all the quantities used in the simulation are in lattice units. In order to convert parameters between physical and lattice units, the following reference scale must be defined: reference length L_r, reference density ρ_r and reference velocity u_r. We use L, ρ, t, ν, c_s to represent length, density, time, viscosity and speed of sound in lattice units respectively, while $L', \rho', t', \nu', c_s'$ represent these quantities in real physical units. The reference units define the relation between the lattice units and physical unit:

$$L_r = \frac{L'}{L}, \quad \rho_r = \frac{\rho'}{\rho}, \quad u_r = \frac{c_s'}{c_s} \tag{A.1}$$

In a simulation, L, ρ, ν, c_s are known, while the real physical quantities ρ', ν', c_s' are also known according to measurements or literature. The only quantities that need to be determined are L' and L_r. A dimensional analysis is conducted to determine the physical and reference lengths. The Reynolds number in the simulation and in the real physical problem should be the same because it is a dimensionless number. Therefore, we can obtain another equation as:

$$\frac{vL}{\nu} = Re = \frac{v'L'}{\nu'} \tag{A.2}$$

Equation (A.2) is equivalent to:

$$\frac{\nu'}{\nu} = \frac{v'L'}{vL} = L_r u_r \tag{A.3}$$

Combining Equation (A.1) and (A.3), we obtain four equations to solve four unknowns: ρ_r, u_r, L_r, L'. The conversion in time can be carried out using equation (A.4):

$$\frac{L_r}{u_r} = \frac{t'}{t} \tag{A.4}$$

With L_r, ρ_r, u_r, it is possible to determine the real time interval $\Delta t'$, distance between nodes $\Delta x', \Delta y'$ and velocity v'.

B. A short manual for single phase/multiphase LB simulations

The parallel lattice Boltzmann simulation package is coded in C/C++ and MPI is used for parallel communication. It uses text-based input files to initialise the simulation. There are two principal source code files:

- `SINGLE_PHASE_MPI_Software_Spars.cpp` : for Single phase simulation.
- `Multi_Component_CG_MPI_SPARSE_Software.cpp` : for multi-component simulation

The source code can be compiled using the following command (provided C/C++/OpenMPI have been installed properly):

```
mpic++ SINGLE_PHASE_MPI_Software_Spars.cpp -o Single_Phase
mpic++ Multi_Component_CG_MPI_SPARSE_Software.cpp -o Multi_Phase
```

B.1. Geometry file

The geometry is imported by an ASCII file containing 0 and 1, where 0 represents a fluid node and 1 represents a solid node. It is assumed that the geometry information is stored in a 3D array $G[i][j][k]$, where (i, j, k) are the coordinates of the mesh point. The format of the geometry input file is in Z,Y,X order. The geometry file can be generated by the following code:

```
for (int k=0;k<nz;k++)
        for (int j=0;j<ny;j++)
                for (int i=0;i<nx;i++)
                cout<<G[i][j][k]<<endl;
                end
       end
 end
```

B.2. Initial phase distribution file

In multi-component calculations, the initial distribution of the wetting phase/non-wetting phase needs to be specified prior to the simulation. A similar phase distribution input file (ASCII file) to the geometry input file is used to initialise the wetting/non-wetting phases. In the phase distribution input file, 1 represents wetting phase whereas -1 represents non-wetting phase. The solid nodes can be specified to have any value. The code for generating the geometry input file can be easily modified to generate the phase distribution input file.

B.3. Single phase flow calculation

To carry out a single phase flow calculation, the initial parameters need to be imported from an input file: `INPUT.dat`. It is a text-based file, input parameters are given at the beginning of every line and comments about the parameters are given after. Space is needed between the parameters and the comments. The following parameters are compulsory for single phase calculations:

- `BC.dat  :Geometry file name`
  Line 2. The file name of the geometry input file which stores the geometry information of the porous medium is specified here, it can include the path of the file if preferred.

- `20 30 3  :nx ny nz`
  Line 3. The size of the geometry is given here, in this case a porous

medium of 20x30x3 is computed.

- `300000 :Maximum time step`
  Line 4. Define the maximum time step you want to calculate. When the program runs beyond this value, it will stop.

- `7.0 :x=1 (um) Resolution`
  Line 5. The resolution of the porous medium is given here. The unit is $\mu m/pixel$.

- `1.0e-6 0.0 0.0 :body force for x,y,z`
  Line 7. The body force applied on the fluid in X,Y,Z direction. Space is needed between parameters.

- `0.05 :Viscosity`
  Line 14. Viscosity is defined here in lattice units.

- `1 :Direction of Permeability Calculation (1=X, 2=Y, 3=Z)`
  Line 18. The direction of the permeability calculation is defined here, 1 denotes permeability calculation is to be carried out in X direction, 2 for Y and 3 for Z.

- `50 :Frequency of results writing`
  Line 19. Calculation summary export frequency in time steps, in this case, every 50 time steps, a summary of the calculation including maximum velocity, averaged velocity, permeability and relative error will be exported on monitor and to a data file.

- `2000 :Frequency of velocity field writing`
  Line 21. Velocity field VTK file export frequency in time steps. In this case, every 2000 time steps, a VTK file `LBM_velocity_Vector_xxxx.vtk` is generated for velocity field visualisation using ParaView[1] xxxx is the time step at which the velocity field exports.

- `2000 :Fre of density field writing`
  Line 22. Export freqency for density field.

[1]ParaView is an open source multiple-platform application for interactive, scientific visualisation.

- `/home/LBM/1_ :OUTPUT PATH`
  Line 25. Output path definition. It is a prefix before every output file. If the velocity file `velocity.vtk` is to be exported, a prefix `/home/LBM/1_` will be put before every output file name. For example, the velocity output file `velocity.vtk` will become `/home/LBM/1_velocity.vtk`. It helps to distribute the outputs to your preferred directory.

- `0 :Geometry Reading format`
  Line 45. Put 0 here to use ASCII geometry input file.

B.4. Multi phase simulation

The multiphase calculation input file (`INPUT_CG_SPARSE.dat`) is similar to the single phase. But more parameters are needed to define the two phase system.

- `phase.dat     :Initial components distribution`
  Line 3. File name of initial phase distribution input file.

- `0 :Psi constant BC`
  Line 8. Use constant phase boundary? If 0 is set, a periodic boundary for wetting/non-wetting phase is used. This means any wetting/non-wetting phase that flows from the outlet will flow into the inlet and keeps the same properties (still wetting/non-wetting). If 1 is set, it means several constant phase boundary conditions are active, more details can be found in Line 16,17,18.

- `0 0 :Psi constant BC in X`
  Line 16. Constant phase boundary on outlet and inlet (we assume that $i = nx$ is outlet and $i = 0$ is inlet.). 0=OFF; if it is set as 1, all the liquid flowing out will flow in periodically to the inlet but the phase colour (blue for wetting and red for non-wetting) will be changed into the colour given by the initial phase distribution function. If it is set as 2, then it will be coloured by the value of its neighbouring nodes.

- Line 17,18 defines the boundary condition for Y and Z direction.

- `0.05    :Viscosity (Component A, psi=1)`
  `0.05    :Viscosity (Component B, psi=-1)`

Line 19,20. Viscosity of the phase two phases.

- `0.7 :#Contact Angle`
  Line 21. This line defines the static contact angle used in the simulation. It should be between -1 and 1. It is the cosine of the static contact angle. If the contact angle is θ, then $\cos(\theta)$ needs to be specified there.

- `1.0e-2 :#Surface tension (Kappa)`
  Line 22. Surface tension definition in lattice units. It should not be larger than 1e-1 to avoid numerical instabilities.

- `2989816.0 :Permeability (Single Phase mD)`
  Line 24. Absolute permeability of the sample, this value is used for relative permeability calculation.

Keep all the other parameters as they were from the original copy.

B.5. Run simulations

To run the simulation:

- Single phase:
  `mpirun -np 8 Single_Phase INPUT.dat`
  where 8 is the number of processors you want to use, `Single_Phase` is the compiled binary executive file. `INPUT.dat` is the input file.

- Multi-phase: `mpirun -np 8 Multi_Phase INPUT_CG_SPARSE.dat`

B.6. Sample Codes

Sample codes including input files, initial geometry files and initial phase distribution files were packed in `sample_codes.tar.bz2`. Sample of different cases are listed in different directories.

B.6.1. Single phase permeability calculation

In directory `Single_Perm_Bentheimer`, the geometry file of bentheimer sandstone and corresponding input file are given. Run the code using commands in Chapter B.5. The permeability of every 50 time steps are saved in the file: `Permeability_error_local_Perm.txt`

The first column is the calculated permeability.

B.6.2. Capillary fingering calculation

In directory `multiphase_capillary_fingering`, the input file and initial geometry and phase files are given. The geometry is a 2D channel of 512x32. Binary fluid with viscosity 0.2 and 0.05 is displaced by a bodyforce of 7.0e-6. The code exports the phase distribution VTK file every 500 time steps. The inlet and outlet boundary condition was fixed colour boundary. All the fluid flowing out is recoloured to wetting fluid before re-entry into the inlet. This boundary condition is defined at line 8 and 16 in the input file.

B.6.3. Drainage/imbibition in a Micro model

The sample input file and initial geometry and phase distribution file are given in directory `micro_model_imbibition`. The geometry is converted from a bmp image file using a matlab code `SD.m`. Periodic boundary conditions are applied in all directions. A reservoir near the inlet was set for the wetting phase. No external force is applied to both phases, the displacement of wetting phase is driven only by capillary force. The surface tension and contact angle are set in line 21 and 22.

B.6.4. Relative permeability calculation of Bentheimer sandstone

Sample code of relative permeability calculation of bentheimer sandstone is given in directory `relative_permeability_bentheimer`. The initial saturation is defined in line 38. The wetting/non-wetting phase is generated randomly according to the pre-defined saturation. Periodic boundaries are applied and we mirror the geometry in order to apply the periodic boundary condition. The initial phase distribution is not required in this calculation.

The calculated relative permeability of component 1/2 can be found in open `Relative_Permeability_Component1.txt`. The first column is the calculated relative permeability of component 1/2, the second column is the saturation of component 1/2

B.6.5. Drainage and imbibition calculation

The drainage and imbibition of Ketton carbonate is simulated using geometry files and initial files in `directory drinage_imbibition`. The drainage process in Ketton rock is simulated with input file `INPUT_CG_SPARSE_dri.dat`. The recolouring boundary condition is applied on inlet and outlet boundary. The fluid flowing out will be recoloured to non-wetting before re-entry into the inlet. Two buffer areas were set near the inlet and outlet and a porous plate was set at the outlet of the rock sample to prevent the non-wetting phase flowing out, which is similar to the experimental configuration in the lab.

The imbibition simulation is slightly different from drainage. The porous medium plate was removed from the outlet to let the non-wetting phase free to flow out. The initial fluid distribution was produced by the drainage simulation. The boundary condition is the same as drainage simulation.

B.6.6. Solute dispersion in the Bentheimer sandstone sample

The solute dispersion is simulated using the geometry and input file in directory `solute_dispersion_bentheimer`. The initial configuration was described in Chapter 8.2. The number of pulse peaks can be adjusted by changing the first value of line 49. The initial location of peaks is set at the third column of line 49. The results are exported in ASCII files: `General_disp_concentration_X_xxxx.sta`. The probability of different displacement of tracking solute is stored in this file for future analysis.