

COMPUTATIONAL ROCK PHYSICS: TRANSPORT PROPERTIES
IN POROUS MEDIA AND APPLICATIONS

A DISSERTATION

SUBMITTED TO THE DEPARTMENT OF GEOPHYSICS

AND THE COMMITTEE ON GRADUATE STUDIES

OF STANFORD UNIVERSITY

IN PARTIAL FULLFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

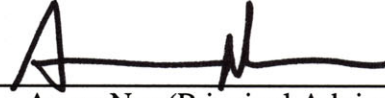
DOCTOR OF PHILOSOPHY

Youngseuk Keehm

January 2003

© Copyright by Youngseuk Keehm 2003
All Rights Reserved

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope of and quality as a dissertation for the degree of Doctor of Philosophy.



Amos Nur (Principal Adviser)

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope of and quality as a dissertation for the degree of Doctor of Philosophy.



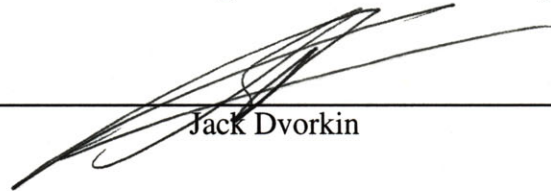
Gary Mavko

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope of and quality as a dissertation for the degree of Doctor of Philosophy.



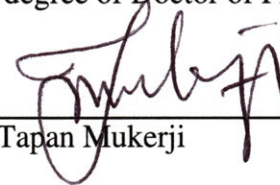
Khalid Aziz

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope of and quality as a dissertation for the degree of Doctor of Philosophy.



Jack Dvorkin

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope of and quality as a dissertation for the degree of Doctor of Philosophy.



Tapan Mukerji

Approved for the University Committee on Graduate Studies:

Abstract

Earth sciences is undergoing a gradual but massive shift from description of the earth and earth systems, toward process modeling, simulation, and process visualization. This shift is very challenging because the underlying physical and chemical processes are often nonlinear and coupled. In addition, we are especially challenged when the processes take place in strongly heterogeneous systems. An example is two-phase fluid flow in rocks, which is a nonlinear, coupled and time-dependent problem and occurs in complex porous media. To understand and simulate these complex processes, the knowledge of underlying pore-scale processes is essential. This dissertation focuses on building transport property simulators in realistic pore microstructures.

These pore-scale simulators will be modules of a *computational rock physics* framework, which will also include other simulators, such as acoustic, elastic, electrical and NMR property simulators. This computational environment can significantly complement the physical laboratory, with several distinct advantages: (1) rigorous prediction of the physical properties, (2) interrelations among the different rock properties in a given pore geometry, and (3) simulation of dynamic problems with coupled and nonlinear physical processes. This dissertation is initiative for the computational rock physics framework, which is a new paradigm of quantitative models for coupled, nonlinear, transient and complex behavior of earth systems.

A rigorous pore-scale simulator requires three important traits: reliability, efficiency, and ability to handle complex microgeometry. We implemented single-phase and two-phase flow simulators using the Lattice-Boltzmann (LB) algorithm, since it handles very complex pore geometries without idealization of the pore space. The single-phase flow simulator successfully replicates fluid flow in a digital representation of a real sandstone sample, and predicts permeability very accurately. This rigorous calculations result from the unique characteristics of the LB algorithm: high angular coverage with 18 velocity directions in 3D. This high angular coverage enables the Lattice-Boltzmann method to accurately solve the non-slip boundary condition without extensive meshing of the pore-grain boundaries in complex pore microstructures. There are two important length scales in digital porous media for the LB flow simulation: grid spacing (dx) and sample size (L). Although optimal choices of these scales are strongly dependent of the pore size distribution of the sample, we found following values are reasonable for well sorted and homogeneous sedimentary rocks in terms of mean pore size (a): $dx=a/10$ and $L=10a$.

Two applications using the single-phase flow simulator are also proposed: a permeability estimation technique from thin sections, and flow-dependent diagenesis modeling. The permeability estimation technique consists of two key components – stochastic 3D porous media construction and flow simulation using the LB method. This technique successfully predicted accurate permeabilities of seven sandstones from thin sections. The LB simulator, again, is very robust even with the stochastically reconstructed pore structures with statistical noises. The most important part is the stochastic construction of porous media from thin sections. We use a sequential indicator simulation technique with porosity and the autocorrelation function as simulation parameters. Although this indicator-based simulator works well with our sandstone samples, higher statistics, such as multipoint correlation functions, are essential for better 3D constructions and for other rock types. The flow-dependent diagenesis modeling is a basic framework for delineating trends of transport properties with a given diagenetic process. The immediate implication of this method will be incorporating more realistic diagenetic mechanisms, such as chemical deposition and dissolution. These two

applications show potential applicability of this robust pore-scale simulator to modeling complex, coupled and dynamic systems.

We extended the single-phase LB algorithm to two-phase systems and implemented two-phase LB flow simulator. The two-phase flow simulator replicates many two-phase flow phenomena, such as Laplace's law, capillary pressure, and snap-off in a pore doublet. We observed intermittent flow and nonlinearity between fluid flux and pressure gradient, which suggest that the extended Darcy's law is not always valid. A simple mixed-wet system was also successfully simulated. The simulator is also ready for exploring a wide range of two-phase flow phenomena more rigorously, including the effect of wettability, capillary pressure and hysteresis, and relative permeability estimation from thin sections.

To overcome the long calculation time of fluid flow simulations, parallel single-phase and two-phase flow simulators were also implemented for efficient and accurate calculation of fluid flow properties of rocks. We developed four optimization techniques customize for the LB algorithm. These optimizations give much better performance than a generic parallelization of the LB flow simulation. The optimized code runs 12 times faster with 14 processors than its serial counterpart, while the generic parallel code is only 4 times faster. With this efficient parallel two-phase flow simulator, we investigated the representative elementary volume for two-phase flow simulation. We found that $L=10a$ is not big enough to represent two-phase fluid flow of the whole rock. We suggest that the length scale of a digital rock should be greater than 20 time the mean pore size ($L=20a$) for two-phase flow simulation. These efficient parallel implementations can help us efficiently simulate very time-consuming applications, which involve multiple two-phase flow simulations, such as diagenesis modeling with two-phase flow and capillary pressure curve simulation.

Acknowledgements

Five and half years at Stanford gave me a tremendous amount of memory and, of course, countless people who helped me have such a pleasant stay.

First and foremost, I would like to express my sincere appreciation to my advisor, Amos Nur for having so much faith in me. His inspiration and motivating comments were always effective and crucial in my academic life. He allowed me to have such a freedom to do whatever I wanted to, and has been very patient. Amos, you are the best advisor for me and will be.

I would like to thank Gary Mavko for his outstanding teaching, and constructive and practical comments. His class was my favorite at Stanford and I learned a lot on how to make presentations clear and effective. I also thank Jack Dvorkin for his advice and encouragement. His help was very crucial to me, especially for my early work at Stanford.

I thank my committee members, Khalid Aziz and KJ Cho for their thorough review and constructive comments on my dissertation.

I would like to thank Tapan Mukerji and Ezequiel Gonzalez. They are my best friends, colleagues and coffee mates. We shared joy, fun, sorrow, and countless cups of coffee. Tapan has always been available for my questions, discussion and most of all, food. He also reviewed most of my papers. Ezequiel always make me feel like his brother

and has been a good listener for all my complaints. I have to thank him for having lunch together everyday, although there is only one choice, Thai café.

A special thank to Manika Prasad for helping me with laboratory data, books and tools. I always remember the kindness of Wendy Wempe. She was my mentor and always willing to help me. My officemates, Diana Sava and Mike Zimmer, have been very nice to me. I really appreciate them for allowing me, such a messy person, to stay with them. I feel honored to have wonderful people in SRB, Ran, Per, Bill, Isao, Haibin, Madumita, Mario, Andrés, Sandra, Juan, Aya, Futoshi, Daniel and Scarlet.

All faculty members, staffs and students in Geophysics gave me a wonderful environment for my study, research and social activity. Tasha and Agnes helped me a lot with all the paper works for my Ph.D.

I cannot thank Margaret Muir enough for her tremendous help all through the years. She helped me with administrative paper work to personal problems, and her help was always superb. Thanks you, Margaret, for all your help and for being my friend.

Last but not the least, I deeply thank all my family. My wife, Jeeyoung, understands my absence and has the endless patience. Without her love, support and encouragement, I could not finish. My little two angels, Munil and Elizabeth, are the main energy for me to endure the tough days and enjoy the good one. My brother, sisters and their family, Hak-Hee, Hyun-Young, Hee-Sun, Boo-In, Kyung-Hye and Yoon-Bum, has been very supportive all through my life and always take care of me. I also thank my nephews and nieces, Sung-Hwan, Soo-Min, Moon-Jin, Moon-Kang, Seo-Hyun, for having faith in me and giving me cheers every time I talk to them. Finally, I would like to share this doctorate with my mother, Joo-Ee. Although she is not with us, I am sure she is proud of me.

Contents

Abstract	iv
Acknowledgements	vii
Contents	ix
List of Tables	xiii
List of Figures	xiv
Chapter 1 Introduction	1
1.1 Rationale behind Pore-scale Modeling	1
1.2 Computational Rock Physics	2
1.3 Transport Properties and Pore-scale Simulation.....	4
1.4 Chapter Description	4
Chapter 2 Transport Properties in Porous Media	6
2.1 Introduction.....	6
2.2 Background of Transport Properties.....	7
2.3 Why Lattice-Boltzmann Flow Simulation.....	10
Chapter 3 Single-phase Lattice-Boltzmann Flow Simulation	14

3.1	Introduction.....	14
3.2	Theory and Implementation.....	15
3.2.1	Cellular Automata Theory	15
3.2.2	Lattice Boltzmann Method	17
3.3	Verification of the Implementation.....	21
3.3.1	Flow through Pipe.....	21
3.3.2	Transport properties of Random Dense Packs of Spheres.....	23
3.3.3	Permeability Simulation on a Digitized Fontainebleau Sandstone.....	27
3.4	Conclusions.....	28
Chapter 4 Permeability Estimation from Thin Sections		30
4.1	Abstract.....	30
4.2	Introduction.....	31
4.3	Method.....	33
4.3.1	Data.....	33
4.3.2	Image Processing	34
4.3.3	2D to 3D Porous Media by Stochastic Simulation	38
4.3.4	Permeability Estimation by Lattice-Boltzmann Flow Simulation	40
4.4	Comparison to the Specific Surface Area Method	44
4.5	Discussion.....	47
4.6	Conclusions.....	49
Chapter 5 Diagenesis Modeling and Transport Properties of Rocks		51
5.1	Abstract.....	51
5.2	Introduction.....	52
5.3	Diagenesis Modeling	53
5.4	Lattice-Boltzmann Method with Matrix Inversion.....	60
5.5	Conclusions.....	63
Chapter 6 Two-phase Lattice-Boltzmann Flow Simulation		64
6.1	Abstract.....	64

6.2	Introduction.....	65
6.3	Algorithm and Implementation.....	66
6.4	Verifications.....	69
6.4.1	Capillary Pressure.....	69
6.4.2	Wettability and Contact Angle.....	71
6.4.3	Two-phase Flow through Pipes.....	74
6.4.4	Simple Mixed-wet System.....	74
6.4.5	Drainage-type Snap-off.....	76
6.5	Two-phase Flow Simulation in Porous Media.....	77
6.5.1	Relative Permeability and Two-phase Flow Simulation.....	78
6.5.2	Steady State Simulation.....	79
6.5.3	Unsteady State Simulation.....	83
6.5.4	Representative Elementary Volume and Grid Spacing.....	87
6.5.4	Summary.....	88
6.6	Effect of Initial Distribution of Two Fluids.....	89
6.7	Future Applications and Implications.....	95
6.8	Conclusions.....	96
Chapter 7 Parallel Lattice-Boltzmann Flow Simulations and Its Implications		98
7.1	Abstract.....	98
7.2	Introduction.....	100
7.3	Motivations.....	101
7.4	Overview of Parallel Implementation with MPI.....	104
7.4.1	Parallel Machine Architecture.....	104
7.4.2	MPI (Message Passing Interface).....	105
7.4.3	General Parallel Implementation.....	105
7.5	Parallel Lattice-Boltzmann Method.....	107
7.5.1	Non-overlapping Boundary.....	109
7.5.2	Graph Partitioning.....	110
7.5.3	Optimized Communication Order.....	112

7.5.4	Communication Balancing – Weighted Graph Partitioning	115
7.6	Algorithms for Parallel LBM Flow Simulations	118
7.7	Simulations and Results	119
7.8	Effect of Optimizations	125
7.9	Conclusions	127
	References	129

List of Tables

Table 4.1: Porosity and permeability values from laboratory measurement and from the flow simulation with 3D porous media simulated from thin sections.	34
Table 7.1: Size of each communication pair. P1 denotes Partition 1 from 8 partitions..	113
Table 7.2: An example of communication ordering.	113
Table 7.3: Procedure to find an optimal order of communication with eight partitions. Numbers in parentheses are partition numbers. The number outside parenthesis is a cumulative communication size in a given partition and at a given step.....	115

List of Figures

Figure 1.1: Schematic diagram of the computational rock physics framework.	3
Figure 2.1: Permeability measurements from different formations of sandstone. Diamonds are Fontainebleau sandstones, bars are Troll sandstones, open squares are Ottawa sands, and circles are from Klimentos and McCann (1990). Solid lines are Kozeny-Carman curves with different parameters (Equation 2.8).	11
Figure 3.1: Schematic description of the Lattice-Boltzmann method. This mesoscale physical model (shown in the center column) shows the same macroscopic behavior as micorscale physics does.....	16
Figure 3.2: Schematic diagram for top-down and bottom-up approaches.....	16
Figure 3.3: The eight velocities in the two-dimensional FCHC model (on the left) and mass associated with each of these directions.	18
Figure 3.4: Eighteen possible particle directions in 3D FCHC. Bold lines denote the lines in which two particles may travel simultaneously.	18
Figure 3.5: Three pipe models with the same porosity (upper row) and corresponding cross-sectional flux distributions (lower row). The local flux values are normalized by the maximum from the biggest pipe model.	22
Figure 3.6: Simulated permeability by the Lattice-Boltzmann method (dots) and the theoretical prediction (solid line).	22
Figure 3.7: (a) A random dense pack of identical spheres (Finney, 1970). The porosity is about 37%. (b) 3D digital representation of the pack with 100×100×100 grid structure.....	24

Figure 3.8: (a) Permeability by the numerical flow simulation using the Lattice-Boltzmann method (solid line) and laboratory measurements with sintered glass beads (circles). (b) Electrical conductivity by the FEM numerical simulation and laboratory measurements. Notations are the same as (a).....	25
Figure 3.9: Formation factors from laboratory measurements (circles) and the FEM simulation (diamonds). Two curves with different exponents in Archie's law (Equation 2.14) are superimposed on the data.....	25
Figure 3.10: Cross plot of permeability and formation factor. A solid line denotes an empirical relation (Equation 2.15 in the form of $\kappa \propto \phi^2/F^2$).	26
Figure 3.11. Comparison between (a) local electric current, (b) local hydraulic flux. Electric current distribution is given by the finite-element method (FEM) and hydraulic flux distribution by the LBM.	26
Figure 3.12: (a) A microtomographic digital rock from a Fontainebleau sandstone sample. Darker area denotes pore space. Lab-measured porosity and permeability are 15.2% and 1100 mD, respectively. (b) Simulated permeabilities with different resolutions of discretization versus laboratory measurements.....	28
Figure 4.1: Scanned images from an epoxy-saturated thin section. Blue color denotes pores. Yellow scale bars denote 1mm. Samples #2-#7 are shown here and Sample #1 is shown in Figure 4.2. The sample numbers correspond to the one in Table 1.....	35
Figure 4.2: (a) Scanned image from an epoxy-saturated thin section (Sample #1). Blue color denotes pores. The yellow scale bar denotes 1mm. (b) Binary image after image processing. The pore space appears in black.....	36
Figure 4.3: (a) 2D autocorrelation function from the binary image. (b) Horizontal and vertical autocorrelation functions (1D). The autocorrelation length (a) of this sample is about 20 grid units.....	37
Figure 4.4: (a) Different square areas from a thin section image for estimating statistical parameters. a is the autocorrelation length. (b) Porosity given by different lengths of squares. Different lines denote different locations of the squares. (c) Autocorrelation functions from squares of different areas. When L is greater than $10a$, autocorrelation functions from different sized squares are very close to each other.....	37
Figure 4.5: Variogram modeling. Circles denote X-variogram (horizontal) and squares Y-variogram (vertical). Gray solid line is a variogram model using an exponential function.	39
Figure 4.6: (a) Conditional data for the sequential indicator simulation. The slices on both ends of the cube have 100×100 pixels and are obtained from the binary image of a	

thin section. (b) Isosurface plot of a 3D realization by the sequential indicator simulation.....	41
Figure 4.7: (a) Cross-sectional areas of a simulated 3D realization. Three areas (XY, YZ and ZX planes) were chosen to verify that the simulated 3D porous media have the same statistical properties. (b) Variograms of those three areas. The variograms of simulated areas of 3D porous media are close to the variogram model.	41
Figure 4.8: (a) Local mass flux by the Lattice-Boltzmann flow simulation. Flux values are normalized by the maximal value. (b) Cross-sectional plots of the high flux area.	42
Figure 4.9: Permeability estimation by the Lattice-Boltzmann flow simulation on stochastically generated 3D porous media. Laboratory measurement is shown as a square. Estimated permeabilities from different 3D realizations are in cross symbols and a circle is an averaged permeability estimation.	43
Figure 4.10: Calculated permeabilities versus laboratory measurements for all seven samples used in this study.	43
Figure 4.11: Steps of estimating the specific surface area as suggested by Blair et al. (1993). (a) High resolution scanned image. (b) Binary image. (c) Autocorrelation function. (d) Angular averaged autocorrelation function and estimation of the slope near the origin.	46
Figure 4.12: Permeability estimations from the seven thin sections by Blair et al.'s (1993) method (squares) and ours (dots).	47
Figure 5.1: Three digital pore structures used in the study. (a) A sinusoidal tube, (b) a random dense pack of identical spheres (Finney's pack) and (c) a digital Fontainebleau sandstone sample.....	55
Figure 5.2: Flux distribution of fluid mass (arrows) in a sphere-pack model and filled pore nodes (red area) according to the different pore-filling mechanisms. Gray area indicate the initial grains.	55
Figure 5.3: Permeability in different pore filling mechanisms.	57
Figure 5.4: Electrical conductivity of different groups. Notations are same as in Figure 5.2.....	58
Figure 5.5: Cross plot of permeability and electrical conductivity. Dotted lines show the power laws ($\kappa \propto F^{-n}$) with $n=1.5, 2$ and 3 , respectively.	58
Figure 5.6: Permeability and electrical conductivity curves for each group. For comparison, $1/F^2$ is used for electrical conductivity values. Legend in the last subplot is the same for all subplots.	59

Figure 5.7: Comparison between iterative version and matrix version of Lattice-Boltzmann method. Matrix version converges to the right value. The calculation time of matrix version is about same as 1000 iterations of iterative version.....	62
Figure 6.1: Schematic diagram of the surface tension calculation. (a) Initial state. Gray color denotes non-wetting fluid and white denotes wetting fluid. The target node is at the center, which is half non-wetting fluid and half wetting fluid. (b) Calculation of the local gradient from the distribution of neighboring nodes. Arrow denotes the direction of local surface tension. (c) Final state. The two fluids are redistributed by the local surface tension. Note that the total mass distribution of the node is deformed by the surface tension force.	68
Figure 6.2: (a) Four different sizes of non-wetting bubbles in a sea of wetting-phase fluid. (b) Capillary pressure vs. reciprocal of bubble radius. Simulated values (dots) agree well with the theoretical prediction (solid line).	70
Figure 6.3: Equilibrium at a line of contact (Craig, 1971).....	71
Figure 6.4: Different wettability situations in porous media: (a) completely water-wet, (b) water-wet, (c) neutral-wet, and (d) oil-wet systems.....	72
Figure 6.5: Wettability test with a simple box model. (a)Initial model has two layers of immiscible fluids separated horizontally. (b), (c) and (d) show simulated results for neutral wettability, weak wettability and strong wettability, respectively.....	72
Figure 6.6: Contact angles by the two-phase Lattice-Boltzmann simulation. Each plot is shown as a cross-sectional view. Initially the non-wetting fluid is located at the bottom center of the box. As wettability increases, the contact angle decreases accordingly. Eventually the non-wetting phase is detached from the bottom and forms a spherical bubble.	73
Figure 6.7: Imbibition (left column) and drainage (right column) through pipes. Fluid is pumping from the left-hand side of the pipe. Each row has a pipe of a different radius. Note that the contact angle at imbibition is different from that at drainage. The two lines ($A-A'$ and $B-B'$) in the middle row are for the pressure profiles in Figure 6.8.	75
Figure 6.8: Pressure profiles along the pipe model for imbibition and drainage. Dots are pressure values from the simulation. The $A-A'$ and $B-B'$ lines are from Figure 6.7. Black solid line denotes a constant pressure drop.....	75
Figure 6.9: Two-phase flow in a mixed-wet system. Plots are shown in cross-sectional views of a pipe model. The solid part has two regions of different wettability. The left half is water-wet and the right half is oil-wet. Initially the pore is filled with oil, and water is pumped from the left-hand side. Time in the title of each plot is in the	

simulation unit. The meniscus between oil and water changes when it meets the boundary between the water-wet and oil-wet solid.	76
Figure 6.10: Drainage-type snap-off in a doublet. (a) Initial stage. Pore space is completely saturated with water (wetting phase). (b) After 5000 iterations. Oil is replacing water in the system. (c) After 10000 iterations. (d) Final stage. There is a continuous oil phase through the bigger pore, while the water phase is trapped in the smaller pore.....	77
Figure 6.11: Schematic diagram of the steady-state simulation of two-phase fluid flow.	79
Figure 6.12: Distribution of two fluids after the steady-state simulation. White parts denote solid grains and greenish yellow denotes pore structure filled with water. The oil phase is shown as red.....	80
Figure 6.13: Normalized flux for non-wetting and wetting phases. Different symbols denote normalized flux values under different pressure gradients. The non-wetting phase shows higher sensitivity to the change in pressure gradient than the wetting phase.	81
Figure 6.14: Two-phase flow in a Fontainebleau sandstone (a) under low pressure gradient, and (b) under high pressure gradient. Fluid flows from right to left. Color notations are the same as in Figure 6.12. Each gray arrow represents a time lapse of 5000 iterations.....	82
Figure 6.15: Schematic diagram of the unsteady state simulation of two-phase fluid flow.	84
Figure 6.16: Snapshots during an unsteady-state simulation; drainage (a-d), irreducible water (e) and imbibition (f-h). Fluid flows from right to left. The blue in (e) denotes water. The other color notations are the same as in Figure 6.12.....	85
Figure 6.17: Normalized flux curves from the unsteady-state simulation.....	86
Figure 6.18: Detailed snapshots during drainage. Note that the movement of the oil phase is perpendicular to the direction of the pressure gradient.	86
Figure 6.19: Initial (left column) and final (right column) distributions of two fluids in Fontainebleau sandstone by (a) boundary, (b) random, and (c) patchy schemes. The wetting fluid saturation is 50% for all cases. Color notations are the same as in previous figures.....	90
Figure 6.20: Relative permeability curves from different initial distributions of two fluids. Different initial distributions of fluids are denoted by different symbols. The non-wetting phase permeability is more sensitive to the change of initial distribution of two fluids.	91

Figure 6.21: Final distributions of two fluids with $S_w=70\%$. (a) High flux pore path from a single-phase flow simulation. (b) Boundary case. (c) Random case. (d) Patchy case.....	92
Figure 6.22: Imbibition simulations on (1) a sample with patchy saturation, and (2) a sample from a drainage simulation.	94
Figure 6.23: Saturations curves for sweep efficiency. The two initial conditions come from imbibition simulation and from patchy saturation.	95
Figure 7.1: Pore geometry of a Fontainebleau sandstone (top). The green, blue and red boxes denote the size of the digital samples for two-phase flow simulations. The length scale of the sample (L) is about $10a$ (a : autocorrelation length). The relative permeability curves from different digital rock samples (bottom). The non-wetting relative permeability is sensitive to the choice of the rock sample location.....	102
Figure 7.2: The effective fluid path in a Fontainebleau sandstone sample. Green color denotes pores, and blue areas are effective flow paths where the local flux is not negligible.....	103
Figure 7.3: Two typical parallel machines: (a) a shared-memory machine and (b) a distributed-memory machine.	104
Figure 7.4: (a) Four partitions for 2D digital data. (b) Partitions with sharing layers. Red square denotes the area of Partition 1.	107
Figure 7.5: Pore geometry of a Fontainebleau sandstone. Gray denotes pore.....	108
Figure 7.6: Procedures of the Lattice-Boltzmann algorithm in a 2D FCHC grid. (a) Initial state of the density distribution. (b) Collision step. (c) Propagation step.	109
Figure 7.7: Schematic diagram of the non-overlapping partition geometry.	110
Figure 7.8: (a)Bisectional partitioning and (b) Metis partitioning with eight partitions. Different colors denote different partitions.....	111
Figure 7.9: Example of bisectional partitioning and graph partitioning on a small digital pore geometry. The graph partitioning method set the partition boundary that has smallest area (pore throats) with good load balancing.....	111
Figure 7.10: Load balances and sizes of communication by the bisectional partitioning and the Metis graph partitioning. (a) Graph partition gives perfect load balancing, while the bi-sectional partitioning shows more than 10% fluctuation in the number of nodes of each partition. (b) The communication cost is really reduced by the graph partitioning method, by almost an order of magnitude. The communication size means the number of data to be communicated.....	112
Figure 7.11: Schematic diagram of communication by the order given in Table 2.....	114

Figure 7.12: Total communication size at each partition by a conventional ordering (circles), the optimized ordering (diamonds) and the optimized ordering with a weighted graph partitioning (squares).	116
Figure 7.13: Improvement of total communication sizes with each optimization.....	117
Figure 7.14: Calculation times of single-phase flow simulation (left) and two-phase flow simulation (right).	120
Figure 7.15: Speedup results for single-phase flow simulation (left) and two-phase flow simulation (right). So-called super-scalability is observed on both a shared-memory machine and a distributed-memory machine with the single-phase flow simulation.	121
Figure 7.16: Efficiency curve from single-phase flow simulations (left) and from two-phase flow simulations (right). Either case has an efficiency better than 80%.	122
Figure 7.17: Two-phase flow in a Fontainebleau sandstone. Isosurface plots for (a) drainage and (b) imbibition. The red part is imbibed oil and green denotes pores filled with water. The sample size is about twice bigger than one that we used in Chapter 6.....	123
Figure 7.18: Bigger digital rock sample for testing the representative elementary volume for two-phase fluid flow (top). Relative permeability curves on two different bigger rock samples (bottom).....	124
Figure 7.19: Performance results from single-phase flow simulations. Circles denote results from the shared boundary partition scheme, diamonds from a bisectional partitioning with non-overlapping partition scheme, and squares from full optimizations. The arrows denotes the performance increase from each optimization.	126
Figure 7.20: Performance results with two-phase flow simulations. Notations are the same as in Figure 7.19.	126
Figure 7.21: Comparison of efficiency (left) and communication time (right) with two-phase flow simulations. The communication time is normalized by total calculation time. With the shared boundary partition, the simulation will spend 70% of the total calculation time in communication.	127

Chapter 1

Introduction

“There is one thing even more vital to science than intelligent methods; and that is, the sincere desire to find out the truth, whatever it may be.” (C. S. Pierce)

1.1 Rationale behind Pore-scale Modeling

A macroscopic physical property of matter depends on its constituents’ microscopic physics. Understanding and simulating the underlying microscopic physics is very useful for determining macroscopic physical properties, as we can see from the advances in material sciences and biology that have resulted from quantum and molecular simulations. As geophysicists, our primary interest lies in determining and predicting physical properties of rocks, which are the main components of the complex and evolving earth crust. At our scale of interest, minerals and pore space are the microscopic constituents of rocks. Thus understanding and simulating pore-scale physics is the key to accurate determination and prediction of physical properties of rocks. Especially, nonlinear and coupled problems, such as two-phase fluid flow in porous media, are very difficult to handle without understanding the pore-scale physics.

Transport properties are probably the most important parameters in many geophysical and engineering situations, including hydrothermal systems, pollutant migration, under-

ground storage of fluids and CO₂, nuclear repositories, and of course oil reservoirs. Because of the complexity of porous media, modeling and calculation of transport properties are often very challenging without simplification or modification of the pore geometry. The primary objective of this dissertation is building rigorous pore-scale simulators for transport properties in realistic pore microstructures. We pay special attentions to three aspects in developing the rigorous pore-scale simulators: *reliability*, *efficiency* and *ability to handle complex microgeometry*.

These rigorous pore-scale simulators will be modules of a *computational rock physics* framework, which will also include simulators for other physical properties, such as acoustic and elastic properties, electrical conductivity, and NMR response. This dissertation is the initiative of the computational rock physics framework, which is a new paradigm of quantitative models for coupled, nonlinear, transient and complex behavior of earth systems.

1.2 Computational Rock Physics

A new computational experimentation environment, the computational rock physics framework, is shown in Figure 1.1. The computational rock physics framework is based on digital representations of rocks, which consist of minerals and fluids, and may evolve with time. It also contains modular physical property simulators, with which we can directly simulate physical properties of rocks. This computational environment can significantly complement the physical laboratory, with several distinct advantages: (1) rigorous prediction of the physical properties, (2) interrelations among the different rock properties using shared digital porous media, and (3) simulation of dynamic problems with multiple physical responses.

Three physical property simulators were completed in this work, a single-phase flow simulator (permeability), an electrical property simulator (electrical conductivity), and a two-phase-flow simulator (relative permeability). Furthermore, two applications using the single-phase flow simulator were proposed: (1) a permeability estimation method from thin sections, and (2) diagenesis modeling with fluid flow. These two applications show the potential applicability of the robust pore-scale simulators to more complex multi-

physics simulations, which involve nonlinear and coupled processes. To overcome the long calculation time of flow simulations, parallel single-phase and two-phase flow simulators were also implemented. Customized optimizations for the Lattice-Boltzmann algorithm give very good performance.

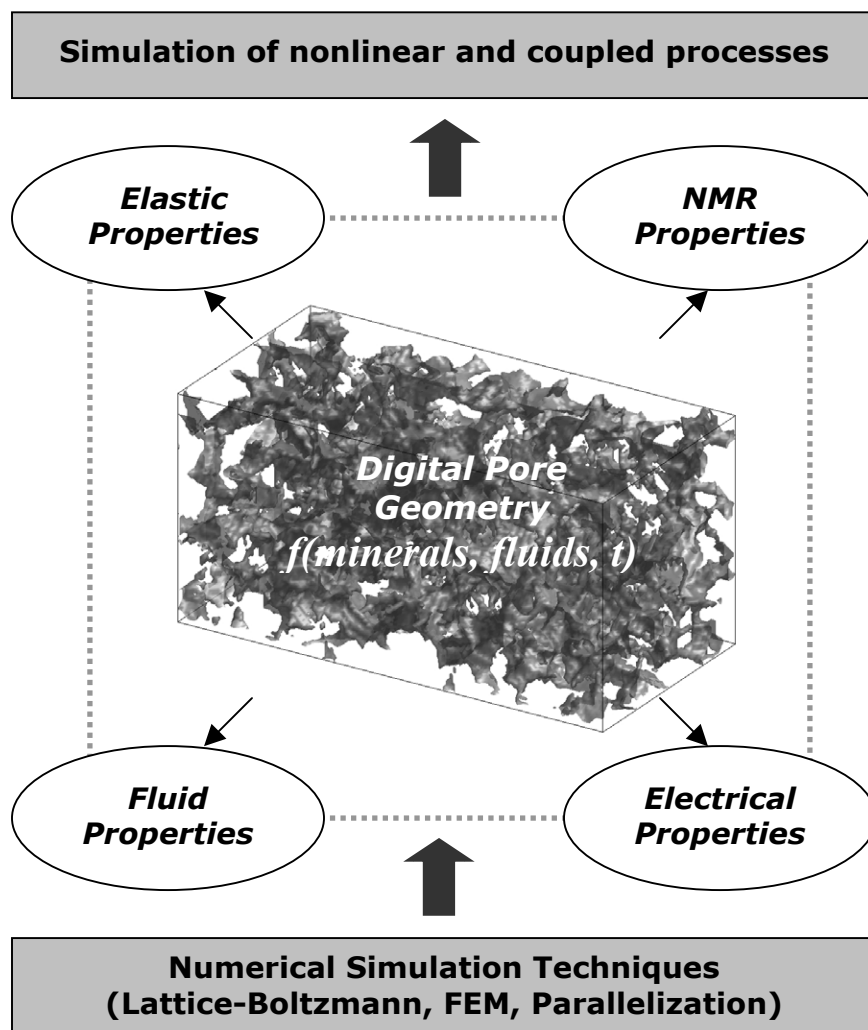


Figure 1.1: Schematic diagram of the computational rock physics framework.

1.3 Transport Properties and Pore-scale Simulation

Because of the complexity of pore geometry, the modeling and calculation of transport properties are often very difficult. Thus the transport properties of rocks have been estimated experimentally in laboratories or through the use of empirical relations. Laboratory experiments can be slow and costly, and empirical relations have not proved to be always effective for many complex processes, such as multiphase flow. As modern computing power increases, numerical simulation techniques have been introduced as an alternative solution. However, implementation of the differential equation can become very complicated with complex geometry of porous media. Either simplification of the geometry or extensive meshing is generally required. A robust and simple tool that can handle complex pore space without over-simplification of the pore geometry is essential.

The Lattice-Boltzmann method (LBM) has the strength to handle complex pore geometry without any modification of the complex pore structure (Cancelliere et al., 1990; Ladd, 1994a; Martys and Chen, 1996; Koponen et al., 1998). It can solve the Navier-Stokes equation accurately with real pore geometries. It has been shown that the LBM can give an accurate solution equivalent to the one by the finite-element method (Kandhai et al., 1998 and 1999) without extensive meshing. It has also been shown that the LBM gives more accurate solutions than a finite-difference method (Ladd, 1994b). Another advantage with the LBM is simple implementation. Moreover, most operations in the LBM are local, thus it is ideal for parallel implementation. Chapter 2 covers the details on transport properties and rationale for choosing the LBM as our algorithm for fluid flow simulations.

1.4 Chapter Description

The following is a brief description of each chapter in this dissertation.

Chapter 1 presents the background and motivation of pore-scale simulations for physical rock properties. It describes a new computational experimentation environment, computation rock physics. The benefits of the Lattice-Boltzmann (LB) method for fluid flow simulators are also briefly covered.

Chapter 2 provides the theoretical background of transport properties. It focuses on the need for a robust numerical algorithm for fluid simulation at the pore scale, and why we choose the LB method for our flow simulations. The advantages of the LBM over other numerical method are discussed.

Chapter 3 describes the details on the LB method, including history, brief theoretical background and implementation. It also shows several numerical experiments to verify that the implementation replicates laboratory measurements and theoretical predictions. Two important scales for the simulation, grid spacing and sample size, are discussed.

Chapter 4 presents a method of permeability estimation from thin sections, which consists of the stochastic construction of 3D porous media from thin section images, and direct flow simulation for permeability estimation. The construction process includes a stochastic simulation technique based on indicator simulators. The method is then compared with a conventional permeability prediction method using two-point spatial correlation without flow simulation.

Chapter 5 presents flow-dependent diagenesis modeling, which is a basic tool for delineating trends of transport properties. It shows how permeability and electrical conductivity change during different diagenetic histories. This chapter also presents a new LB implementation using sparse matrix inversion. The new implementation shows better convergence in low-porosity and low-permeability rocks.

Chapter 6 describes two-phase flow simulation using the LB method. It presents the implementation of the two-phase flow simulator by extending the single-phase algorithm. It then shows numerical experiments for verification of the implementation, and simulation results on a digital Fontainebleau sandstone sample. The effect of initial distribution of two fluids and future applications are also covered.

Chapter 7 presents parallel implementations of single-phase and two-phase flow simulators. It describes various optimization techniques customized for the LB algorithm. Our parallel code with these optimizations especially customized for the LB algorithm performs much better than a conventional parallel implementation. Using the parallel two-phase flow simulator, the elementary representative volume (REV) for two-phase fluid flow is explored and discussed.

Chapter 2

Transport Properties in Porous Media

2.1 Introduction

Porous media, such as rocks in the earth, consist of pore space and a solid matrix. The pore spaces are usually connected, which allows fluid flow and mass transfer to take place. Permeability and electrical conductivity, as well as other parameters, are strongly influenced by the pore structure and pore-scale physics. In fact, we can determine the transport properties of a porous medium exactly if we can solve the physical equations on a real pore space. However, this approach requires a detailed description of the pore geometry, which is rarely known. Even if it were known, the mathematics and computation would become impractical. Thus it is common to ignore the microscopic mechanical details and to describe porous media with volume-averaged laws that treat a porous medium as a macroscopically homogeneous continuum. At this scale, the transport properties of rocks have been estimated experimentally in laboratories or through the use of empirical relations. However, the experiments can be slow and costly and the empirical relations have not always proved to be effective for many complex processes, such as multiphase flow. As modern computing power increases, numerical simulation techniques have been introduced as an alternative solution. Network modeling (Fatt, 1956; Bryant et al., 1993) is one widely used technique, which simplifies the pore

space into pores and connecting bonds. This simplification gives two benefits: a reasonable amount of pore geometry information can be incorporated, and the flow simulation is simple enough for very large networks. The most significant limitation of this approach is dividing the pore structure into discrete pores and pore throats, which arbitrarily alter the real pore structure. The finite-element method (FEM) is robust and accurate, but requires extensive meshing and special numerical libraries. The finite-difference method (FDM) is simple to implement, but does not give as accurate solutions as the FEM does. A simple method that can calculate the transport properties accurately, and can handle complex pore space without over-simplification of the pore geometry is essential for building our rigorous transport property simulators.

This chapter briefly reviews transport properties of rocks, hydraulic permeability and electrical conductivity. It will cover the governing equations and empirical relations with other pore geometry parameters, such as porosity, specific surface area and tortuosity. This chapter then will discuss the need for numerical simulation techniques and cover a few techniques including the Lattice-Boltzmann method (LBM). Finally, the benefits of the LBM, which is the main reasons to choose the LBM for our simulator, will be described.

2.2 Background of Transport Properties

Permeability is a measure how easily fluid flows through a given porous medium. The mechanics of the flow is given by the Navier-Stokes equation:

$$\frac{d}{dt}(\rho \mathbf{u}) + \nabla \cdot (\rho \mathbf{u} \mathbf{u}) = -\nabla P + \eta \nabla^2 \mathbf{u}, \quad (2.1)$$

where ρ is density, \mathbf{u} is the velocity, ∇P is pressure gradient and η is the dynamic viscosity of the fluid. Permeability can be defined by Darcy's law:

$$\langle \mathbf{q} \rangle = -\frac{\mathbf{K} \cdot \nabla \langle P \rangle}{\eta}, \quad (2.2)$$

where $\langle \mathbf{q} \rangle$ and $\langle P \rangle$ are volumetric averages of fluid flux and pressure, and \mathbf{K} is the permeability tensor. The equation can be reduced to

$$\langle q_x \rangle = -\frac{\kappa}{\eta} \frac{dP}{dx} \quad (2.3)$$

in the 1D case, when the pressure gradient is only applied along the x -direction.

Since permeability of rocks is difficult to measure or simulate, empirical relations are often used to estimate permeability through other pore structure parameters that can be measured more easily. Kozeny and Carman (Carman, 1961) provided a good empirical relation between fluid permeability and other measurable properties of porous media. The derivation is based on flow through a pipe with radius R . The flux in the pipe can be written as

$$Q = qA = -\frac{\pi R^4}{8\eta} \frac{dP}{dx} \quad (2.4)$$

where Q is the volumetric flow rate and A is the cross-sectional area. Comparing with Darcy's law (Equation 2.3), the effective permeability becomes

$$\kappa = \frac{\pi R^4}{8A} = \left(\frac{\pi R^2}{A} \right) \frac{R^2}{8}. \quad (2.5)$$

The porosity ϕ , and the specific area S for the pipes are expressed as

$$\phi = \frac{\pi R^2}{A}, \quad S = \frac{2\pi R}{A}, \quad (2.6)$$

then the Kozeny-Carman relation becomes (Mavko et al., 1998)

$$\kappa = \frac{B \phi^3}{\tau^2 S^2} = \frac{B \phi^3}{\tau} d^2 \quad (2.7)$$

where B is a geometric factor, τ is the tortuosity, and d is a characteristic length scale of pores. Another widely used variation of the Kozeny-Carman relation is,

$$\kappa = a \phi^n. \quad (2.8)$$

The coefficient a and the exponent n are empirically determined. Although these empirical relations are intended to relate permeability with other more easily measurable parameters, the relation frequently introduces empirical parameters that are very difficult to quantify. Tortuosity and the geometrical factor B are still not very easy to quantify in

realistically complex pore structures. Extensive details of this empirical relation and other variations can be found in Mavko et al. (1998), Guéguen and Palciauskas (1994), and Walsh and Brace (1984).

The effective (macroscopic) *electrical conductivity* of porous media can be expressed by Ohm's law:

$$\mathbf{J}(r) = \sigma(r)\mathbf{E}(r) = -\sigma(r)\nabla V(r), \quad (2.9)$$

where \mathbf{J} is electrical current density (flux), σ is the local conductivity, \mathbf{E} is the electrical field, and V is the electrical potential. For time-independent electrical flux, the equation for charge conservation becomes

$$\nabla \cdot \mathbf{J}(r) = -\nabla \cdot [\sigma(r)\nabla V(r)] = 0. \quad (2.10)$$

The pore-scale potential function $V(r)$, can be computed by solving Equation 2.10 using the input parameter $\sigma(r)$ for rock and fluid. The effective conductivity is then computed from

$$\sigma_{eff} = -\frac{\langle \mathbf{J} \rangle}{\Delta V} \quad (2.11)$$

where σ_{eff} is the effective electrical conductivity of the rock and $\langle \mathbf{J} \rangle$ is the volumetric average flux. The differences in conductivity between mineral grains are usually very small compared to the differences in conductivity between the minerals and saline water. Therefore a water-saturated rock can be regarded as a two-component medium: solid grains and saline water. Since the conductivity of grains is much smaller than that of the fluid, the effective conductivity of a rock saturated with fluid will be proportional to the conductivity of fluid σ_w as follows:

$$\sigma_{eff} = \frac{\sigma_w}{F} \quad (2.12)$$

where F is known as the formation factor (Archie, 1942).

Since detailed information on pore microstructure is not usually known, we have to rely on empirical relations between the formation factor and porosity. One widely used relation is the so-called Archie's law,

$$F = \phi^{-m} \quad \text{or} \quad F = (\phi - \phi_o)^{-m} \quad (2.13)$$

where ϕ_o is percolation porosity. The exponent m varies approximately from 1.3 to 2.5 for most sedimentary rocks and is close to 2 for sandstones. Archie's law is sometimes written as,

$$F = a\phi^{-m} \quad \text{or} \quad F = a(\phi - \phi_o)^{-m} \quad (2.14)$$

where a is an empirical constant close to 1.

Since electrical conductivity is usually easier to measure than permeability *in situ* or in the laboratory, there have been efforts to relate permeability to electrical conductivity. Walsh and Brace (1984) modified the Kozeny-Carman relation using the formation factor:

$$\kappa = \frac{\phi^2}{c F^2 S^2}, \quad (2.15)$$

where c is another geometrical factor, with $c = 2$ for circular pores and $c = 3$ for flat cracks. There are many other variations of the empirical relations (Mavko et al., 1998; Guéguen and Palciauskas, 1994; Walsh and Brace, 1984). However, no single relation could fit all different data from different rock samples.

2.3 Why Lattice-Boltzmann Flow Simulation

Figure 2.1 shows laboratory measurements of porosity and permeability for several types of sandstones with clay contents less than 5%. Several empirical relations are superimposed on laboratory measurements with different exponents (Equation 2.8). We first notice that the permeability ranges very widely, spanning several orders of magnitude even with the same porosity. Some data sets can be fitted with an empirical relation, while some others are impossible to fit with these empirical curves. In addition, these empirical relations give very little physical explanation about the relation between permeability and microstructure of porous media. It is clear that we need to consider pore-scale physics to find a rigorous relation between permeability and other pore geometry parameters. This pore-scale physics will also help us understand which pore geometry parameter has the most effect on permeability in a specific rock type. However,

this problem is very difficult to solve analytically because of the complexity of the geometry and the governing equations. Thus numerical simulation is a practical choice. Several techniques such as network modeling, finite element and finite difference methods and the Lattice-Boltzmann method have been used for simulating fluid flow in porous media.

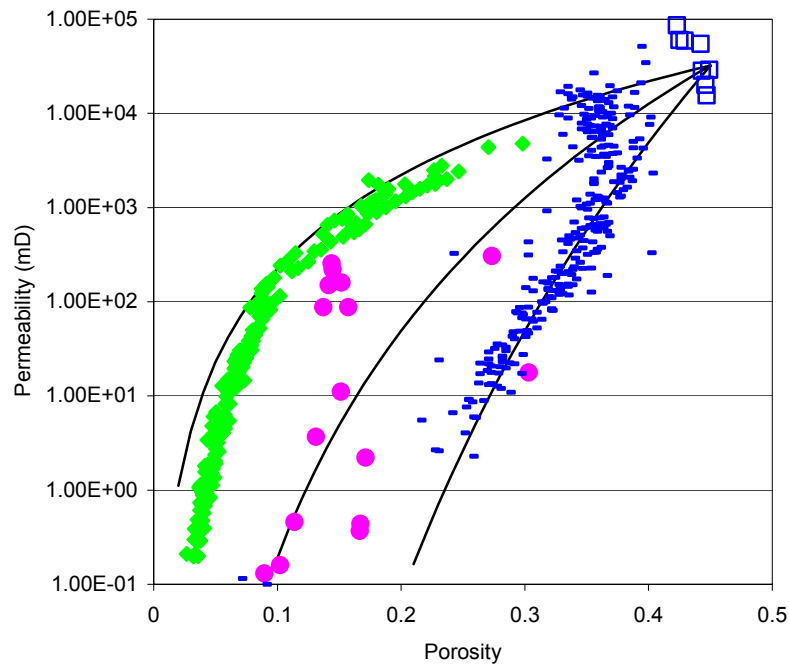


Figure 2.1: Permeability measurements from different types of sandstone. Diamonds are Fontainebleau sandstones, bars are Troll sandstones, open squares are Ottawa sands, and circles are from Klimentos and McCann (1990). Solid lines are Kozeny-Carman curves with different parameters (Equation 2.8).

The main idea of network models is to represent a complex pore structure as a simple network. This network has a certain amount of pore structure information, such as connectivity and pore size distribution. At the same time, it provides a model in which simple flow calculations can be made. The network consists of nodes that represent pore junctions and bonds that connect nodes as capillary conduits. Network models were initially introduced to study capillary pressure (Fatt, 1956) on regular 2D or 3D grid

structures. Modern network models are more representative of real pore structures (Bryant et al., 1993; Bakke and Øren, 1997; Liang et al., 1999 and 2000; Dillard et al., 2000). The biggest limitation of network models is the need to divide the pore space into discrete pores and pore throats. In real pore space, the distinction between pores and throats is not very clear, so the choice of the distinction is always somewhat arbitrary. The bottom line is that a network model is not a real pore space.

Several other techniques using finite-element methods (FEM) or finite-difference methods (FDM) have been introduced. Adler et al. (1990) used an alternating-direction-implicit (ADI) finite-difference scheme on simulated 3D porous media to solve the partial differential equation for Stokes flow of a Newtonian fluid. Saeger et al. (1995) solved the Stokes equation using the finite-element method. Generally, the FEM involves quite complicated meshing for media with complex geometry. The FDM can use simple regular grid structures, but requires very fine grid structure for accurate calculation with complex geometry (Ladd, 1994b). The complex pore structure is often simplified to solve the problem with these methods. Moreover, multiphase flow at the pore scale is very challenging with these conventional methods (Rothman, 1988).

The Lattice-Boltzmann (LB) method, on the other hand, chooses a different approach. The LB method is based on cellular automata theory, which describes a complex system by the interaction of a massive number of cells following simple local rules (Doolen, 1990; Chen et al., 1992b; Chopard and Droz, 1998). The rules in the Lattice-Boltzmann method recover the Navier-Stokes equation at the macroscopic scale (Ladd, 1994a and 1994b; Rothman and Zaleski, 1997). While other methods, such as network models, the FEM or the FDM, discretize the model and the governing equations (top-down approach), the Lattice-Boltzmann method recovers the governing equation from the rule in the discrete model (bottom-up). The biggest advantage of the Lattice-Boltzmann method is that it is readily applied to any arbitrary discrete geometry (Cancelliere et al., 1990; Ladd, 1994a; Martys and Chen, 1996; Koponen et al., 1998). In addition, it describes fluid flow in porous media very accurately. This strength comes from the characteristics of the LB algorithm. Although the LB method uses a regular Cartesian grid like the FDM, one node has 18 different velocity directions in 3D. This high angular coverage can give an

accurate boundary condition on the pore-grain boundaries, equivalent to the one by the FEM with extensive meshing. In essence, the LBM can simulate fluid flow with the high accuracy of the FEM but with the simplicity of the FDM (Sangani and Acrivos, 1982; Ladd 1994b; Kandhai et al., 1998 and 1999; Fredrich et al., 1999). The LBM has a very significant advantage when the pore structure becomes complex. Another advantage with the LB method is that the implementation is very simple, and portable for different computer platforms. It is also ideal for parallel implementation, since most operations in the LB method are local. Lastly, it may be easier to incorporate more mesoscopic physics, such as mass transfer, chemical reaction and diagenesis, in the LB method than in the Navier-Stokes equations themselves, since the lattice Boltzmann equation is closely related to the continuum Boltzmann equation of nonequilibrium statistical mechanics (Succi, 2001). For certain applications, such as multiphase flow in porous media, there is a growing consensus that the LB method is a strong contender for the best fluid-simulation approach currently available (Fredrich et al., 1999; Succi, 2001).

Chapter 3

Single-phase Lattice-Boltzmann Flow Simulation

3.1 Introduction

This chapter covers theoretical background and implementation of two numerical simulation methods: the Lattice-Boltzmann method for single-phase fluid flow simulation and a finite element method for the electrical conductivity simulation. We will focus on the details of the Lattice-Boltzmann method (LBM) for single-phase flow simulation in this chapter. The two-phase Lattice-Boltzmann simulation technique, which is extended from the single-phase algorithm, will be described in Chapter 6. We used the Diffpack software library (Langtangen, 1999) for the finite element method to simulate the electrical conductivity of porous media, by simply solving Equation 2.10. Details on the finite element method will not be covered in this work, since it is a well-established technique and it is not the focus of this dissertation. We will describe extensive numerical experiments and compare the results with laboratory measurements and theory. The main objective of this chapter is to show the feasibility and applicability of the LBM.

The LBM is based on statistical (macroscopic) description of microscopic phenomena. It describes fluid motion as collisions of imaginary particles (Figure 3.1), which are much

bigger than the real fluid molecules. These particles have nearly nothing in common with real fluids, but they show almost the same behavior at a macroscopic scale. In fact, the LBM have been shown to recover the Navier-Stokes equation at the macroscopic scale (Frisch et al., 1986; Chen et al., 1992a; Ladd, 1994a). Because the LBM describes macroscopic fluid flow by the collective behavior of many fictitious molecules but not by full molecular dynamics, it is referred to as a mesoscopic description of microscopic physics (Figure 3.1).

The biggest advantage of the LBM comes from its unique derivation (Figure 3.2). While other methods, such as network models, the finite-element method (FEM) and the finite-difference method (FDM), discretize the model and the governing equations (top-down approach), the Lattice-Boltzmann method recovers the governing equation from the rules in the discrete model (bottom-up approach). Since originated from the discrete lattice, we do not have to discretize the governing equation of the model; *i.e.* it is readily applied to any arbitrary discrete geometry.

3.2 Theory and Implementation

3.2.1 Cellular Automata Theory

Cellular automata (CA) theory is an idealization of a physical system in which space and time are discrete, and the physical quantities take only a finite set of values. In brief, cellular automata can describe a complex system by the interaction of a massive number of cells following simple local rules. At the end of 1940s, von Neumann suggested an abstraction of self-reproducing mechanisms using a fully discrete universe made up of cells (von Neumann, 1966). Each cell is characterized by an internal state and evolves in discrete time steps according to a simple rule that is a function of the states of neighboring cells. HPP (Hardy-Pomeau-de Pazzis) lattice gas models were developed in the 1970s (Hardy et al., 1973 and 1976). These models consist of simple and fully discrete collections of particles moving and colliding on a two-dimensional square lattice so as to conserve momentum and particle number. Because the HPP does not have the property of isotropic relaxation of the stress tensor, it was replaced by the famous FHP (Frisch-Hasslacher-Pomeau) model (Frisch et al., 1986).

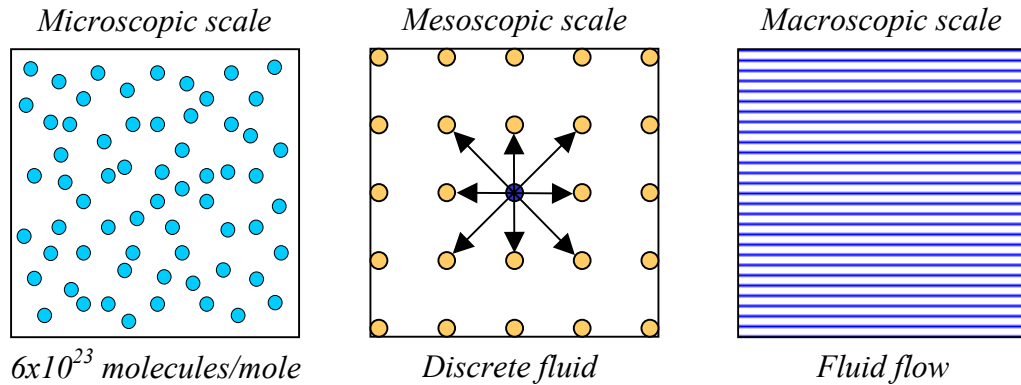


Figure 3.1: Schematic description of the Lattice-Boltzmann method. This mesoscale physical model (shown in the center column) shows the same macroscopic behavior as micorscale physics does.

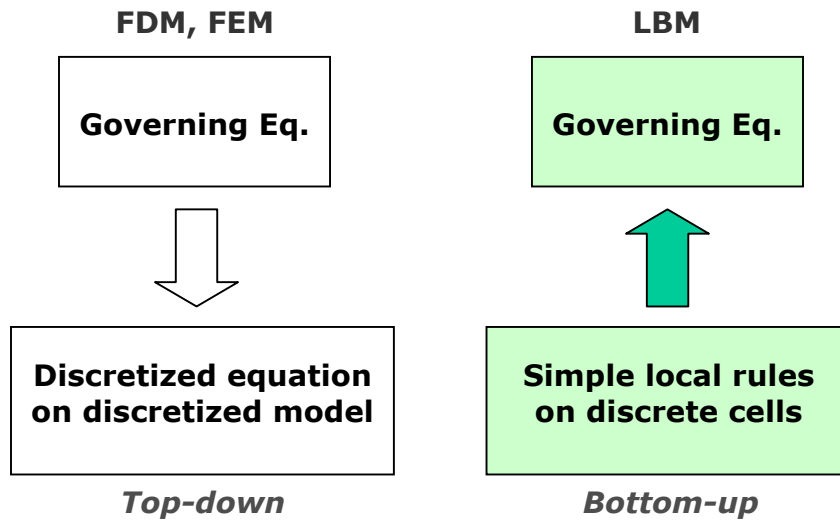


Figure 3.2: Schematic diagram for top-down and bottom-up approaches.

Frisch et al. (1986 and 1987) showed that their model follows the behavior of the Navier-Stokes equation of hydrodynamics in the long-time and large-scale limits, in spite of the discrete dynamic description of the model (Chen et al., 1992a; Ladd, 1994a). These two limits are related to low Mach and Knusen numbers (Qian et al., 1997; Ladd, 1994a). Simply, the low Knusen number means that the grid spacing is much smaller than

characteristic length scale (e.g. mean pore size), and the low Mach number means that the fluid velocity is much smaller than the speed of sound. These models are often referred to as lattice-gas automata (LGA) since they were initially developed as theoretical models to study fundamental statistical properties of an interacting gas.

LGA fluids represent idealized N-body systems. The time evolution can be performed exactly without any approximations or round-off errors. However, the LGA has several weaknesses. Due to Boolean dynamics, the LGA generates much noise in the local distribution, even in steady state. Since the velocity field of the LGA is quite noisy, one has to average large patch of cells (e.g. 32×32 squares) over several consecutive time steps. This process slows down the algorithm and requires finer lattice structure. In addition, the LGA has little flexibility to adjust parameters in order to describe a wide range of physical situations. After one chooses a collision rule, the viscosity is fixed. If we want to change the viscosity of the fluid, a new collision rule must be implemented.

3.2.2 Lattice Boltzmann Method

The Lattice-Boltzmann method (LBM) describes the state quantity (particle density) as a continuous function. The density of a particle is no longer a Boolean variable but a probability of presence that is ranged continuously in the interval $[0,1]$. There are several lattice models (D’Humières et al., 1986); we use a face-centered hypercube (FCHC) grid in this study (Higuera et al., 1989). In order to have isotropic relaxation of the stress tensor required by fluid equations, the grid should be formed in 4D and be projected into 2D or 3D (D’Humières et al., 1986; Chen et al., 1992a). The FCHC grid used in this work has 24 different velocity directions in 4D, which are expressed as permutations of $(\pm 1, \pm 1, 0, 0)$. After projection, the 2D FCHC has 8 velocities (Figure 3.3) and the 3D FCHC has 18 different velocities (Figure 3.4). More details about the FCHC and other variations can be found in D’Humières et al. (1986), Higuera et al. (1989), Ladd (1994b), Nannelli et al. (1992), and Chen et al. (1992a).

The Navier-Stokes equation can be recovered by solving the discrete Boltzmann equation for the particle density distribution, $n_i(\mathbf{r}, t)$ at time t and a location of r . The subscript i denotes each different direction, which is defined by the velocity vector, \mathbf{c}_i . Then the local mass density ρ , momentum density \mathbf{j} , and momentum flux tensor Π are

expressed as follows:

$$\rho(\mathbf{r}, t) = \sum_i n_i(\mathbf{r}, t), \tag{3.1}$$

$$\mathbf{j}(\mathbf{r}, t) = \sum_i \mathbf{c}_i n_i(\mathbf{r}, t), \tag{3.2}$$

$$\Pi_{\alpha\beta}(\mathbf{r}, t) = \sum_i c_{i\alpha} c_{i\beta} n_i(\mathbf{r}, t), \tag{3.3}$$

where $v_{i\alpha}$ is the α component of velocity vector \mathbf{c}_i .

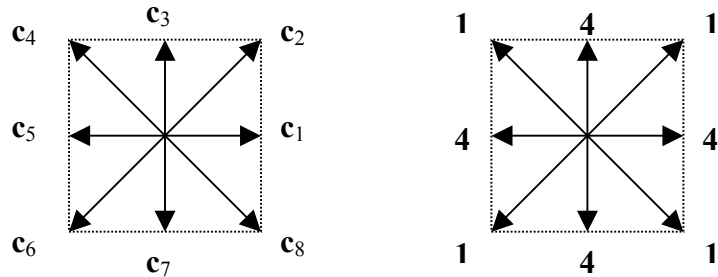


Figure 3.3: The eight velocities in the two-dimensional FCHC model (on the left) and mass associated with each of these directions.

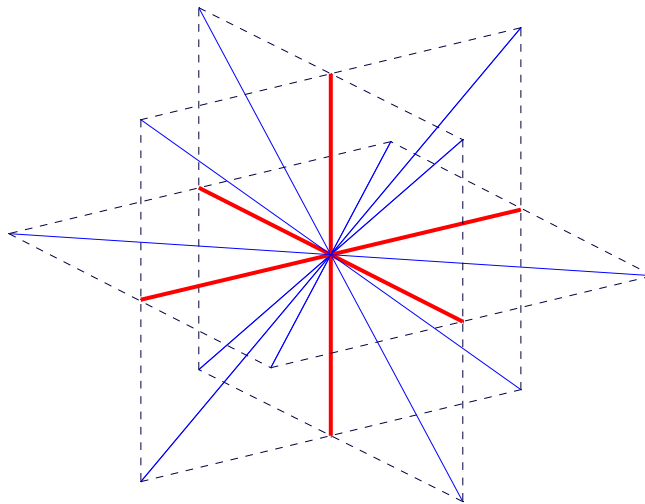


Figure 3.4: Eighteen possible particle directions in 3D FCHC. Bold lines denote the lines in which two particles may travel simultaneously.

The time evolution of the distribution function $n_i(\mathbf{r}, t)$ is described by the discrete Boltzmann equation:

$$n_i(\mathbf{r} + \mathbf{c}_i, t + 1) = n_i(\mathbf{r}, t) + \Omega_i(\mathbf{r}, t), \quad (3.4)$$

where $\Omega_i(\mathbf{r}, t)$ is a collision operator. Then $n_i(\mathbf{r} + \mathbf{c}_i, t + 1)$ is the distribution of mass density at the next time step. This evolution of the distribution function is a two-step process consisting of collision and propagation. In the collision step, the mass and momentum are redistributed by the collision operator at each lattice node. The updated distribution at location \mathbf{r} streams to the neighboring location, $\mathbf{r} + \mathbf{c}_i$ in the propagation step.

Exact collision operators have been derived for several different lattice-gas models (Frisch et al., 1987). However, linearization of the collision operator is useful for numerical calculations (Higuera et al., 1989). After linearization, the linearized collision operator (L) satisfies the following eigenvalue equations (Ladd, 1993):

$$\begin{aligned} \sum_i L_{ij} &= 0, \quad \sum_i \mathbf{c}_i L_{ij} = 0, \\ \sum_i \text{off}(c_{i\alpha} c_{i\beta}) L_{ij} &= \lambda \text{off}(c_{j\alpha} c_{j\beta}), \quad \sum_i c_i^2 L_{ij} = \lambda_B c_j^2 \end{aligned} \quad (3.5)$$

where $\text{off}(\cdot)$ denotes the off-diagonal part of the tensor, (\cdot) . The first two equations follow from the conservation of mass and momentum, and the last two equations describe the isotropic relaxation of the stress tensor. The eigenvalues λ and λ_B are related to the shear and bulk viscosities. After setting up the eigenvalues according to the properties of the fluid, the post-collision distribution is expressed by

$$\begin{aligned} n_i(\mathbf{r}, t) + \Omega_i(\mathbf{r}, t) &= \\ a_0 \rho + a_1 j_\alpha c_{i\alpha} + a_2 \Pi'_{\alpha\beta} (c_{i\alpha} c_{i\beta} - \text{tr}(c_{i\alpha} c_{i\beta})) + a_3 (\Pi'_{\alpha\alpha} - 3\rho c^2) \end{aligned} \quad (3.6)$$

where Π' is an updated momentum flux tensor. The new distribution can be obtained after determining the coefficients, a_0 , a_1 , a_2 , and a_3 , according to the properties of the fluid.

The two-step process is repeated until the distribution reaches steady state. After steady state is achieved, the local flux (\mathbf{q}) can be calculated as follows:

$$\mathbf{q}(\mathbf{r}) = \sum n_i(\mathbf{r}) \mathbf{c}_i. \quad (3.7)$$

From the local flux, we calculate a volume-averaged flux. Then, the macroscopic permeability is calculated by Equation 2.3. In summary, the algorithm for the single-phase Lattice-Boltzmann method is as follows:

- (1) Calculate local density, momentum and stress by Equation 3.1-3.3.
- (2) Calculate post-collision distribution by Equation 3.6 (collision step.)
- (3) Apply a force according to the pressure gradient (∇P).
- (4) Move the distribution to neighboring nodes by Equation 3.4 (propagation step)
- (5) Repeat (1)-(4) until steady state.

The external force is implemented as perturbation of density distribution after collision (Ladd, 1994a; Martys et al., 1999; Halliday et al., 2001). The non-slip boundary condition of fluid flow is implemented by the well-known bouncing-back scheme (Ladd, 1994a; Inamuro et al., 1995; Maier et al., 1996). We use a no-flow boundary condition at sidewalls of the sample, which are perpendicular to the pressure gradient direction. For the inlet and outlet areas, we use a periodic boundary condition, *i.e.* outflow at the outlet becomes inflow at the inlet, since it is easy for mass conservation. To eliminate a boundary effect with this periodic boundary condition, we mirror 3D cube of porous media, in which we have the same geometry at the inlet and the outlet.

The algorithm is easy and simple to implement. The operation count is linear to the number of pore nodes where the fluid resides. Regardless of the complexity of the pore structure, the operation count per iteration is the same for the same number of pore nodes. The biggest advantage of the Lattice-Boltzmann method is that it can handle complicated pore geometry without simplifications or modifications (Ladd, 1994b; Spaid and Phelan, 1997). It is also easy to incorporate external fields, such as pressure gradients or gravity. In addition, since most intensive numerical operations are local, calculations can be parallelized to take advantage of modern multiprocessor computers. Other advantages of the LBM were already covered in the last of Chapter 2.

Many studies have successfully applied the LBM to a wide range of fluid phenomena: single-phase flow (Bosl et al., 1998; Keehm et al., 2001), two-phase flow (Rothman and

Keller, 1988; Gunstensen and Rothman, 1993), colloidal particles (Ladd, 1993), liquid-gas phase transition (Shan and Chan, 1994), Rayleigh-Taylor instability (Nie et al., 1998) and flow in random fiber webs (Koponen et al., 1998). The LBM has also been applied to other types of simulation: electrical conductivity (Küntz and Mareschal, 1997), elastic properties (Küntz et al., 1997), and wave propagation (Chopard et al., 1997).

3.3 Verification of the Implementation

This chapter shows results of numerical experiments with simple porous models, the results of which are known from laboratory measurements or from theory. Both electrical conductivity and permeability are simulated and verified by laboratory measurements or by theory.

3.3.1 Flow through Pipe

The first model is a simple cylindrical tube model. Figure 3.5 shows three different cylindrical pore structures with different diameters. All of them have the same porosity. The local fluid velocities are normalized by the maximum in the largest pipe. The fluid velocity has a parabolic profile like in the Poiseuille flow. This laminar Poiseuille flow has an exact analytic solution for the cylindrical pipes as follows:

$$Q = \frac{\pi R^4}{8\eta} \frac{\partial P}{\partial x} \quad (3.8)$$

where η is the dynamic viscosity, R is a radius of the tube and $\partial P/\partial x$ is a pressure gradient. After applying Darcy's law (Equation 2.3), the permeability (κ) for the tube is given by

$$\kappa = \frac{\pi R^4}{8A} . \quad (3.9)$$

where A is the cross-sectional area. Figure 3.6 shows calculated permeability from the LBM simulation (dots) and predicted permeability from Equation 3.9. The simulated values show excellent agreement with the theoretical prediction. The small discrepancy at small diameter comes from the discretization effect.

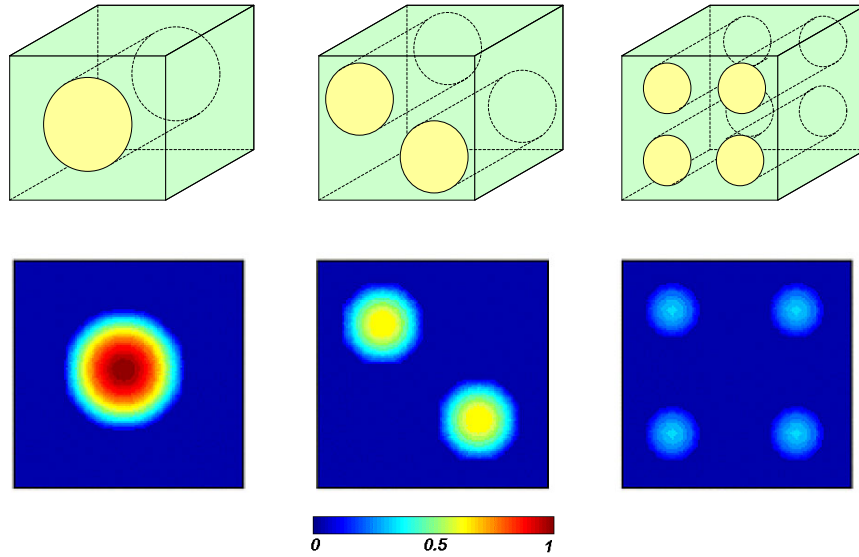


Figure 3.5: Three pipe models with the same porosity (upper row) and corresponding cross-sectional flux distributions (lower row). The local flux values are normalized by the maximum from the biggest pipe model.

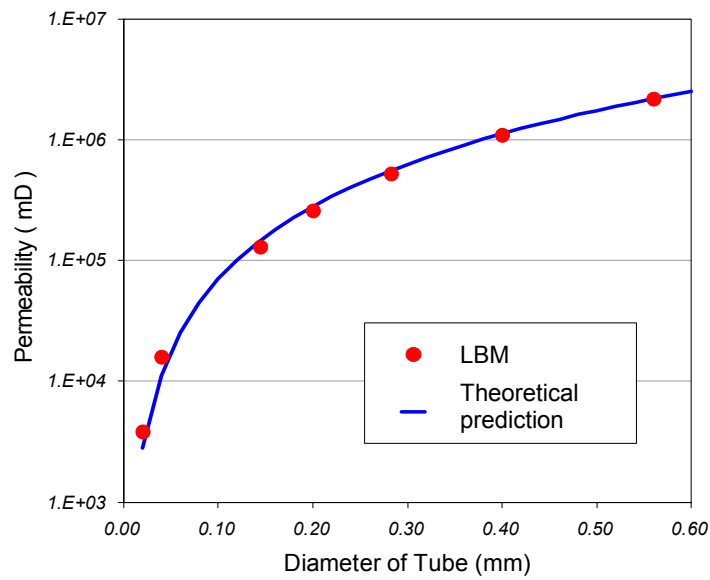


Figure 3.6: Simulated permeability by the Lattice-Boltzmann method (dots) and the theoretical prediction (solid line).

3.3.2 Transport properties of Random Dense Packs of Spheres

Random dense packs of spheres are very useful for testing an implementation of fluid flow, for several reasons. Firstly, they are reasonable first approximations to real sandstones, and yet are well defined mathematically. Secondly laboratory measurements can be easily obtained with sintered glass beads, and both theoretical formula and laboratory measurements are widely available (Wong et al., 1984; Roberts and Schwartz, 1985). Thirdly, a wide range of porosities can be obtained both in the laboratory using sintered glass beads, and in simulations by uniform growth or shrinking of the spheres of the packing. Thus they are good models to test a new implementation for fluid flow simulation.

We use Finney's pack for our digital porous model (Finney, 1970). Finney's pack is random, dense and spherical packing of precision ball bearings. It contains more than 8,000 identical ball bearings with a radius of 1mm. The spatial coordinates of the bearing centers have been experimentally measured and are available in the digital format. We select a rectangular domain from the pack, which will be our digital rock sample for flow simulation. Figure 3.7 shows an example of a random dense pack of spheres and a 3D digital rock sample obtained by digitization of the pack. The cube is 8mm on each side and has $100 \times 100 \times 100$ grid structure. The grid spacing was chosen to be smaller than 0.1mm, which is equivalent to more than 20 grid points per sphere diameter. The effects of sample size and grid size are briefly covered in Bosl (1999) and will be covered in detail later in this chapter and in Chapter 4.

Laboratory measurements of permeability and electrical conductivity using sintered glass beads were obtained from Schlumberger-Doll Research Lab (Roberts and Schwartz, 1985). Three sizes of glass beads were used, whose diameters are $48.5\mu\text{m}$, $96.5\mu\text{m}$ and $193.5\mu\text{m}$, and whose porosities range from 2.5% to 40%. Each sample contains only one size of glass beads. To eliminate the scale effect from different sizes of glass beads, all permeability values are normalized by the square of the diameters, which makes the permeability dimensionless. The Lattice-Boltzmann flow simulations and finite-element electrical conductivity simulations were performed with identical sphere packings. The wide range of porosity was obtained by uniform expansion or shrinking of the spheres.

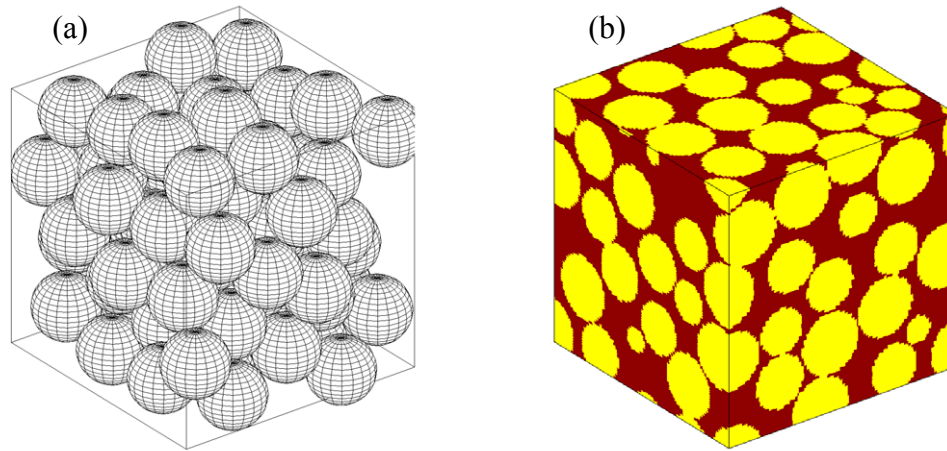


Figure 3.7: (a) A random dense pack of identical spheres (Finney, 1970). The porosity is about 37%. (b) 3D digital representation of the pack with $100 \times 100 \times 100$ grid structure.

Figure 3.8(a) shows simulated permeability from the numerical flow simulation and laboratory measurements. The agreement between simulated values and laboratory measurements is exceptionally good. The electrical conductivity by the FEM simulator also shows very good agreement with laboratory measurements in Figure 3.8(b).

Figure 3.9 shows Archie's law (Equation 2.14) superimposed on the calculated and laboratory measured electrical conductivity. We clearly see that the measured formation factors are not explained by a single Archie's law. Roberts and Schwartz (1985) and Wong et al. (1984) measured the electrical conductivity of sintered glass beads, and showed the same results – the exponent in Archie's law is smaller at high porosity. The simulated values show exactly the same trend, which suggests that the numerical simulations replicate the laboratory measurements very accurately. A cross plot of permeability and formation factor is shown in Figure 3.10. Simulated values show good agreement with laboratory measurements. A solid line denotes an empirical relation curve (Equation 2.15). The empirical relation fits both data sets very well. One advantage of our pore-scale simulations on porous media is shown in Figure 3.11. After simulations, we obtain the local hydraulic flux and electrical current, which can be used for exploring and comparing the details of fluid flow and electric current at the pore scale.

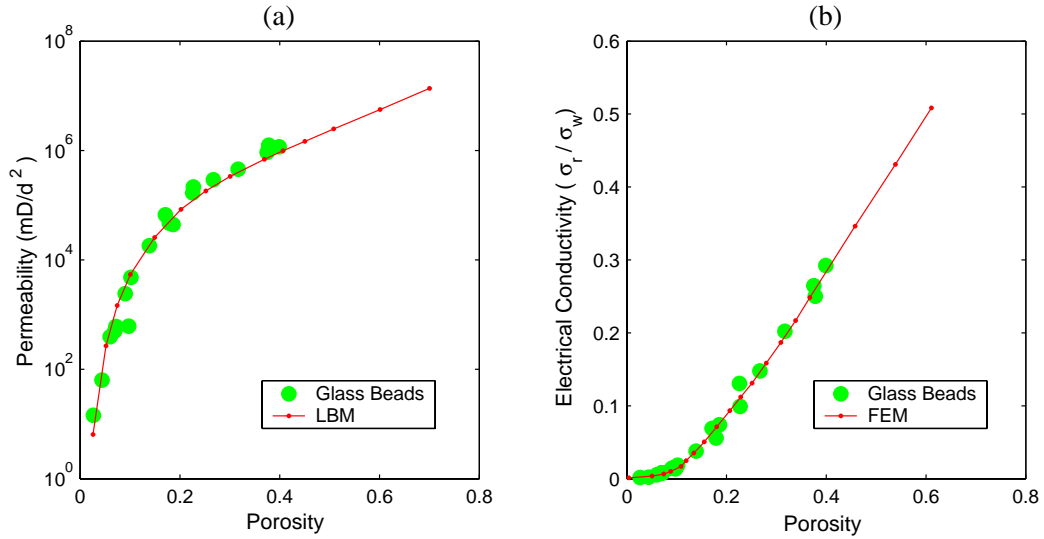


Figure 3.8: (a) Permeability by the numerical flow simulation using the Lattice-Boltzmann method (solid line) and laboratory measurements with sintered glass beads (circles). (b) Electrical conductivity by the FEM numerical simulation and laboratory measurements. Notations are the same as (a).

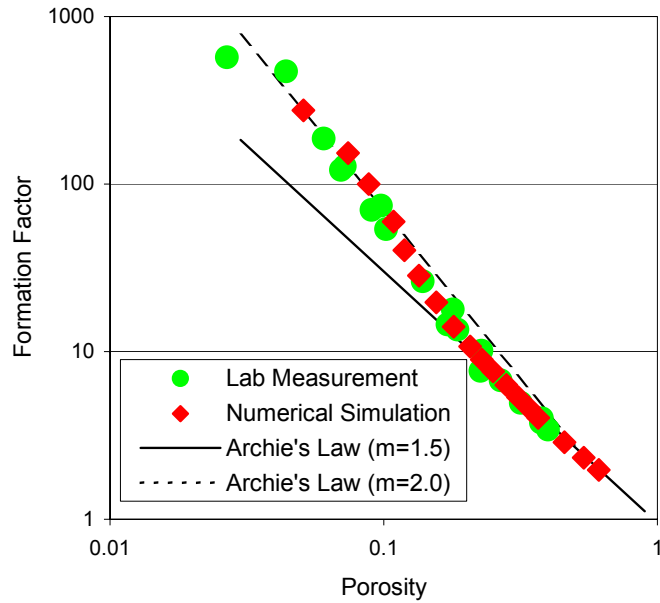


Figure 3.9: Formation factors from laboratory measurements (circles) and the FEM simulation (diamonds). Two curves with different exponents in Archie's law (Equation 2.14) are superimposed on the data.

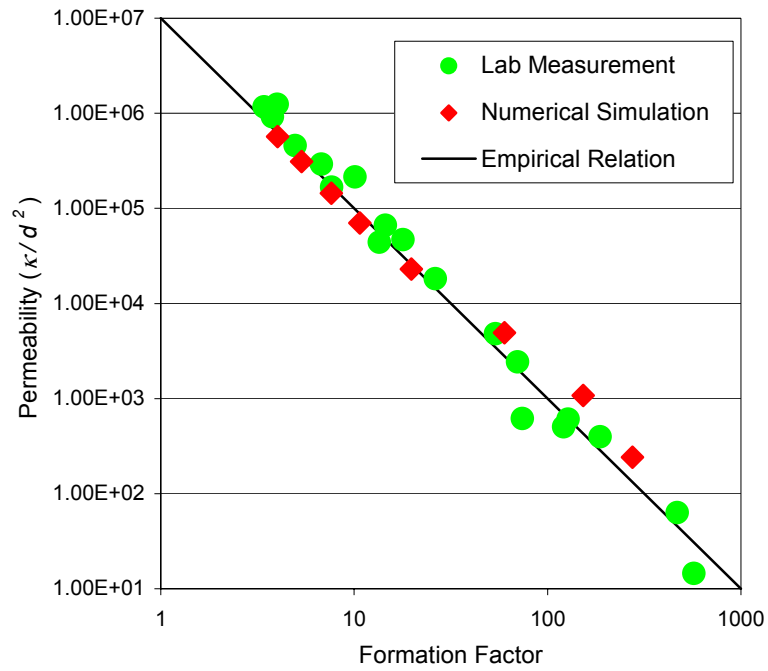


Figure 3.10: Cross plot of permeability and formation factor. A solid line denotes an empirical relation (Equation 2.15 in the form of $\kappa \propto \phi^2/F^2$).

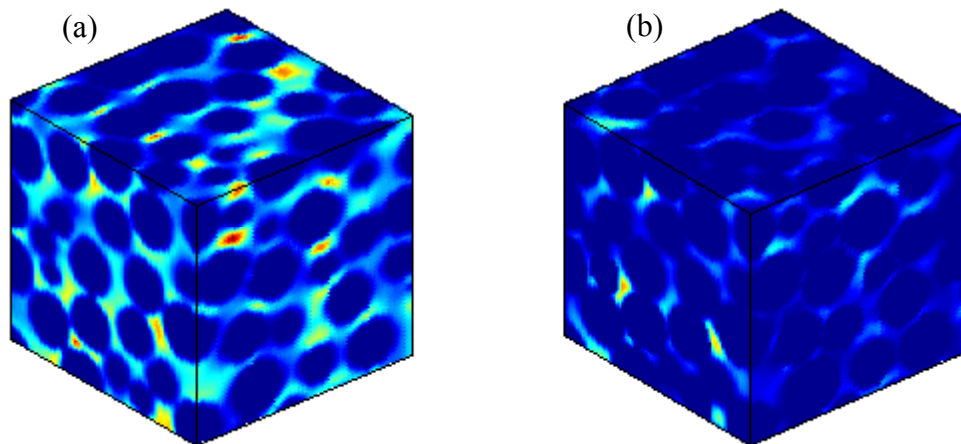


Figure 3.11. Comparison between (a) local electric current, (b) local hydraulic flux. Electric current distribution is given by the finite-element method (FEM) and hydraulic flux distribution by the LBM.

3.3.3 Permeability Simulation on a Digitized Fontainebleau Sandstone

For this last example, we use a digital sample of a real rock. Recent advances in imaging techniques using X-ray microtomography make it possible to obtain 3D pore structures at high resolution. The digital rock sample was reconstructed from X-ray images of Fontainebleau sandstone at $7.5\mu\text{m}$ spatial resolution. Cubic subsets of the digital rock sample are used in other chapters of this dissertation. One subset of the digital Fontainebleau sandstone is shown in Figure 3.12 with $100\times 100\times 100$ cells.

The LBM flow simulation is performed on cubic subsets of the digital rock with different resolutions to investigate the digitization effect. To determine the digitization effect, we need to define a few length scales. Let a be mean pore size and dx be grid spacing. We will investigate three digital rocks with different resolution scales: $dx=a/5$, $dx=a/10$, and $dx=a/20$, respectively. Figure 3.12(b) shows calculated permeabilities of different resolutions of rocks, compared to laboratory measurements.

Knackstedt and Zhang (1994), and Zhang et al. (2000) showed that the simulated permeability using the Lattice-Boltzmann method would be overestimated with a coarse grid resolution. They showed that the calculated permeability converges to true permeability when the grid spacing is infinitely small. The estimated permeability would increase very gradually as the grid spacing increases until the grid spacing cannot resolve characteristic pore scale.

We observe the same results in Figure 3.12. As the grid is coarser and coarser, the calculated permeability increases. Although all our choices of grid spacing give reasonable predictions of permeability, the grid spacing of $dx=a/5$ has non-negligible overestimation. In homogeneous sandstone samples, such as our Fontainebleau sandstone sample, we found that a grid spacing with $dx=a/10$ is a reasonable choice for single-phase flow simulation. There is another important scale, the size of digital rock sample. If a digital sample is not big enough to represent fluid flow of the whole rock, the calculated value will be meaningless. This issue will be covered in Chapter 4 (representative elementary volume for single-phase fluid flow).

The comparison between the calculated permeability by the LBM and the laboratory measurements shows very good agreement, especially with the fine grid. Even with the

medium grid, the discrepancy between the LBM simulation and laboratory measurement are only 10%. From the results of numerical experiments, we concluded that the LBM gives us a very accurate permeability prediction and is a very robust tool for flow simulation in complex pore structures.

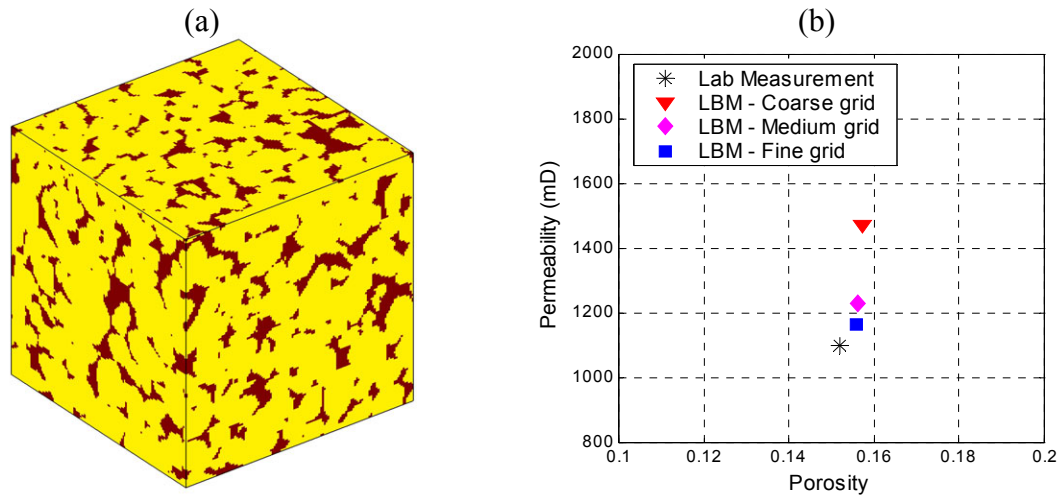


Figure 3.12: (a) A microtomographic digital rock from a Fontainebleau sandstone sample. Dark area denotes pore space. Lab-measured porosity and permeability are 15.2% and 1100 mD, respectively. (b) Simulated permeabilities with different resolutions of discretization versus laboratory measurements.

3.4 Conclusions

We have provided a brief background and implementation of the Lattice-Boltzmann method. The biggest benefit of the method is that it can naturally handle very complex pore geometry, which is ideal to explore the pore-scale physics of fluid flow. We observed that the numerical simulators – the Lattice-Boltzmann method for fluid flow simulation and the finite element method for electrical conductivity simulation – can replicate fluid flow and electric current for various models successfully and accurately. Calculated permeability and electrical conductivity showed excellent agreement with laboratory measurements or predictions by theory.

There are two important length scales in digital porous media for the LB flow

simulation: grid spacing (dx) and sample size (L). The grid spacing should be small enough to realistically resolve the grain-pore boundary condition, and the sample size should be big enough to represent fluid flow of the whole rock. Although optimal choices of these scales are strongly dependent of the pore size distribution of the sample, we found that $dx=a/10$ and $L=10a$ are reasonable choices for well sorted and homogeneous sedimentary rocks in terms of mean pore size (a). As long as we satisfies these two conditions, the LBM gives very accurate results on flow simulation.

Chapter 4

Permeability Estimation from Thin Sections

4.1 Abstract

This chapter introduces a new methodology for estimating permeability from thin sections. The method is based on two key components – constructing 3D porous media from 2D thin sections and a direct 3D flow simulation using the Lattice-Boltzmann method. From the thin section, statistical parameters such as porosity and autocorrelation function are calculated through image processing. We generate 3D porous media using a geostatistical method of sequential indicator simulation conditioned to the thin section. We then perform flow simulations on those 3D realizations without idealizing or simplifying the complex pore geometry. The Lattice-Boltzmann method can successfully handle realistic and very complicated 3D pore geometries. We apply our method for seven thin sections of sandstone samples from the Daqing oil field. Estimated permeabilities show excellent agreement with laboratory measurements. We then compare ours to another method of permeability estimation based on thin sections without flow simulation. Our method gives better results and is less sensitive to statistical noise from thin sections. More importantly, our method does not require any empirical

calibration parameters. The proposed method is a more direct and robust technique for the estimation of permeability from thin sections.

4.2 Introduction

The prediction of permeability remains one of the most important challenges in quantitative rock physics. Many relatively successful and common methods are empirical ones, such as the Kozeny-Carman relation based on simple cylindrical pore geometry (Carman, 1961). These models have been widely applied because they are easy to use and are simple to understand (Dullien, 1992; Mavko and Nur, 1997). However, these models are difficult to relate in detail to real rocks and sediments. Furthermore, these models are only valid for a certain property, in this case, fluid-flow in rocks. We may need a totally different model to calculate other physical properties, such as elastic moduli or NMR response. Thus, it is very crucial to have common sharing model – digital representations of real rocks – not only for accurate prediction of physical properties, but also for interrelations among the different physical responses. Though we have imaging techniques using X-ray microtomography for obtaining 3D pore structures at high resolution, it is still a very expensive and time-consuming task. Thin sections, though it is 2D, are widely available and are very easy to obtain. To predict permeability from thin sections has great impact on oil reservoirs and underground systems. Two main objectives of this study are (1) to build 3D porous media from thin sections, which will be sharing objects for other physical property simulations, and (2) to estimate permeability from the 3D porous media extended from thin sections.

Walsh and Brace (1984) related the permeability to porosity, a geometrical factor, the formation factor and the specific surface area. They applied the relationship for low porosity and low permeability granites. These kinds of relationships provide a way of relating the permeability of porous media to other rock parameters, such as porosity, specific surface area, tortuosity and grain size. However, most of these parameters are not easy to measure from samples or thin sections. Porosity is the main controlling parameter in the relationship and is relatively easy to measure, while the specific surface area, the next important parameter, is very difficult to measure accurately from thin sections.

Debye et al. (1957) and Berryman (1987) suggested a relationship for estimating the specific surface area from spatial correlation functions of thin sections. Blair et al. (1993) calculated the porosity and the specific surface area of sandstone thin section using Berryman's formula (Berryman, 1987), and then estimated permeability using the relationship by Walsh and Brace (1984). Estimated permeability agreed well with laboratory measurement. However, the calculation of the specific surface area from images is sensitive to the resolution of thin section images. A small change in image resolution gives a relatively large error in estimated permeability. Though the porosity and the specific surface area may be estimated from thin sections, the formation factor and the geometrical factor in their formula are not directly measurable from thin sections. Empirical estimates or laboratory measurements are required for these parameters.

Adler et al. (1990) chose a different approach. They measured porosity and autocorrelation function from binary thin sections and generated 3D structures based on the unconditional truncated Gaussian method. Adler's stochastic simulation uses a two-step process: first simulating a Gaussian random field with right correlation and then truncating it at an appropriate threshold. The threshold is chosen so as to give the right porosity. Truncating the continuous Gaussian field creates a binary 3D cube by assigning 0 and 1 to values below and above the threshold, respectively. One slight complication comes in at the estimation of the correlation function from the binary thin section, since the correlation of the binary image is not the same as that of the Gaussian random function. Adler et al. (1990) then applied an alternating-direction-implicit (ADI) finite difference scheme to the simulated 3D porous media to solve the partial differential equation for Stokes flow of a Newtonian fluid. Five Fontainebleau sandstone samples were used for their study, and estimated permeability agreed reasonably well with laboratory measurements. However, the size of simulated cubes does not seem to be big enough to represent fluid flow of the whole rock. Moreover, the use of the Gaussian random function for the correlation of the binary images is not always appropriate. As pointed out at the end of Chapter 3, the FDM method is not as good as the LBM for flow simulation.

Liang et al. (1999) suggested a similar Gaussian truncation method to generate 3D

porous media from thin sections. However, they processed 3D pore geometry further into skeletons by a thinning algorithm that can preserve connectivity. They then perform network modeling from the skeleton to estimate permeability. Even though they start with the full 3D pore geometry, they ultimately simplify it into a network, because they use network modeling for flow simulation.

Our method uses a technique based on binary images of thin sections. However, rather than resorting to an empirical relation or network modeling, we generate realizations of 3D porous media by conditional sequential indicator simulation with statistical parameters from the 2D binary image. Since we want binary 3D porous media, an algorithm based on a binary indicator function rather than the continuous Gaussian function is more natural and straightforward. After obtaining multiple stochastic realizations, we perform fluid flow simulations using the Lattice-Boltzmann method on the simulated 3D pore geometries. Finally, we estimate the permeability from the computed fluid flux. The Lattice-Boltzmann method is robust enough to handle very complicated simulated 3D structures with statistical noise. Seven different thin section samples from the Daqing oil field were used for this study (Prasad and Nur, submitted).

In the following sections, we will show the procedures for generating 3D porous media from thin sections by image processing and stochastic simulation, and finally estimating permeability by the Lattice-Boltzmann flow simulation. Laboratory measurements are compared to the numerically estimated permeability. We also compare our method to the one by Blair et al. (1993). We find that our method shows better agreement with laboratory measurements and is less sensitive to statistical error from thin section imaging. In addition, we should emphasize that our method needs no adjustable constant or empirical calibration parameters.

4.3 Method

4.3.1 Data

Thin sections were obtained from Daqing oil field sandstone samples. They are relatively clean sandstones with clay content between 4-10%. Laboratory measurements, including porosity and permeability, were made on cores of one-inch diameter and two-

inch length. Porosity and Klinkenberg-corrected air permeability were measured at room pressure and at 5.5 MPa confining pressure under controlled room temperature. The porosity varies from 20% to 30% and the range of the permeability is 40 mD to 1,700 mD (Table 4.1).

Table 4.1: Porosity and permeability values from laboratory measurement and from the flow simulation with 3D porous media simulated from thin sections.

Sample No.	Porosity (%)			Permeability (mD)		
	Laboratory	Estimation	Difference	Laboratory	Estimation	Difference
1	30.0	28.7	1.3	1,766	1,679	87 (4.9%)
2	26.9	27.8	0.9	1,030	1,431	401 (38.9%)
3	25.2	25.6	0.4	818	946	128 (15.6%)
4	28.1	29.0	0.9	591	566	25 (4.2%)
5	23.4	24.7	1.3	295	320	25 (8.5%)
6	22.0	20.8	1.2	91	101	10 (11.0%)
7	20.1	20.0	0.1	39	61	22 (56.4%)

4.3.2 Image Processing

Thin sections were obtained from sandstone samples saturated by epoxy. The pore space is clearly distinguishable since the epoxy was dyed blue. Images of thin sections were made with an optical microscope. The position of the image was randomly chosen from the thin section in order to prevent operator bias. In each thin section, multiple images were taken and we chose one based on the quality of the image. Figure 4.1 shows thin section images (Samples #2-#7). Blue color denotes pore spaces and others are grains. The images consist of 600×400 pixels and yellow scale bars denotes 1mm. From the color image of the thin section, a binary image can be obtained using simple image processing (Figure 4.2). We converted the true color image into an index image and selected proper indices for grains and pore space.

The binary image can be represented by an indicator function $f(r)$,

$$f(r) = \begin{cases} 1 & \text{if } r \text{ belongs to the pore space,} \\ 0 & \text{otherwise,} \end{cases} \quad (4.1)$$

where r denotes a spatial location within the binary image.

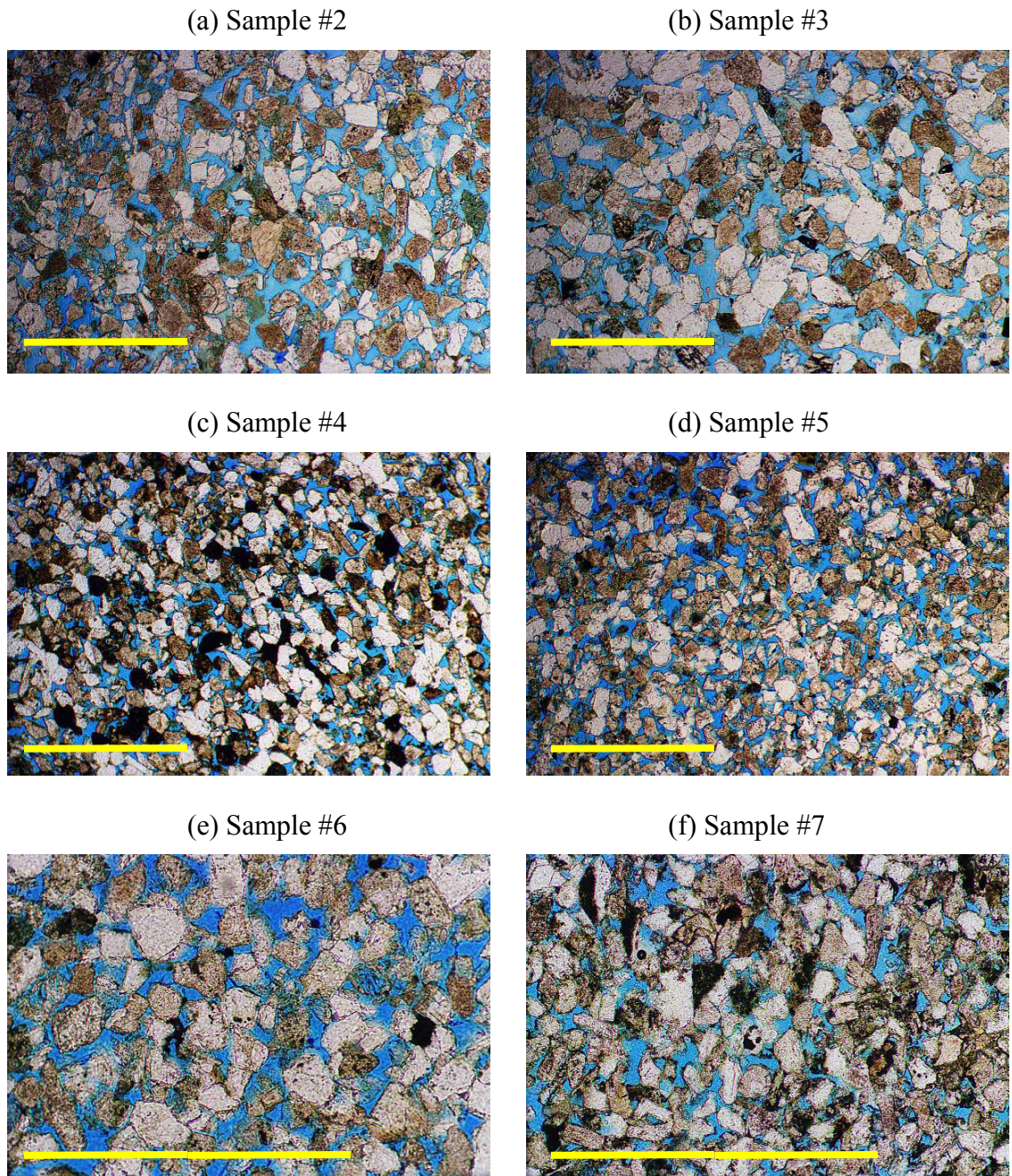


Figure 4.1: Scanned images from an epoxy-saturated thin section. Blue color denotes pores. Yellow scale bars denote 1mm. Samples #2-#7 are shown here and Sample #1 is shown in Figure 4.2. The sample numbers correspond to the one in Table 1.

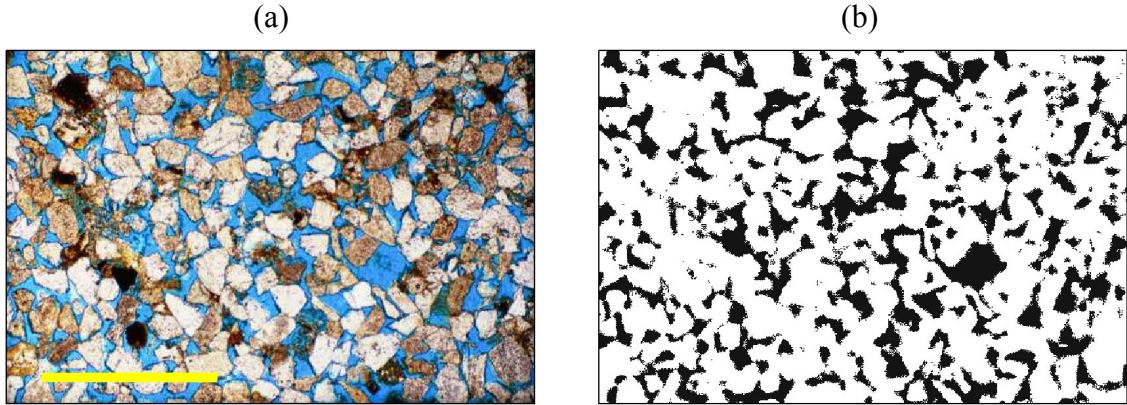


Figure 4.2: (a) Scanned image from an epoxy-saturated thin section (Sample #1). Blue color denotes pores. The yellow scale bar denotes 1mm. (b) Binary image after image processing. The pore space appears in black.

The porosity ϕ and the autocorrelation function $A(\mathbf{h})$, (*i.e.* the two-point correlation function) can be defined by the statistical averages, denoted by $\langle \rangle$,

$$\phi = \langle f(r) \rangle \quad (4.2)$$

$$A(\mathbf{h}) = \langle f(r)f(r+\mathbf{h}) \rangle \quad (4.3)$$

where \mathbf{h} is a lag vector between two data points. Two important properties of the autocorrelation function of the binary image are (Blair et al., 1993)

$$A(0) = \phi \quad (4.4)$$

$$\lim_{|\mathbf{h}| \rightarrow \infty} A(\mathbf{h}) = \phi^2. \quad (4.5)$$

The autocorrelation function can be easily obtained using Fourier transforms,

$$A(\mathbf{h}) = F^{-1} \{ F\{f(r)\} \times F^* \{f(r)\} \} \quad (4.6)$$

where $F\{\}$, $F^{-1}\{\}$ denotes Fourier and inverse Fourier transforms and $*$ denotes the complex conjugate. Figure 4.3 shows a 2D autocorrelation function of the binary image shown in Figure 4.2(a). Two orthogonal 1D autocorrelation functions (horizontal and vertical) in Figure 4.3(b), indicate that the binary image is nearly isotropic since the difference is negligible. The autocorrelation length (a) is shown in Figure 4.3(b), which is

the length from the origin until the autocorrelation function stop decreasing. The autocorrelation of this sample is about 20 grid units. The porosity measured from the binary image is 0.29. We can see that Equations (4.4) and (4.5) are successfully verified in Figure 4.3(b).

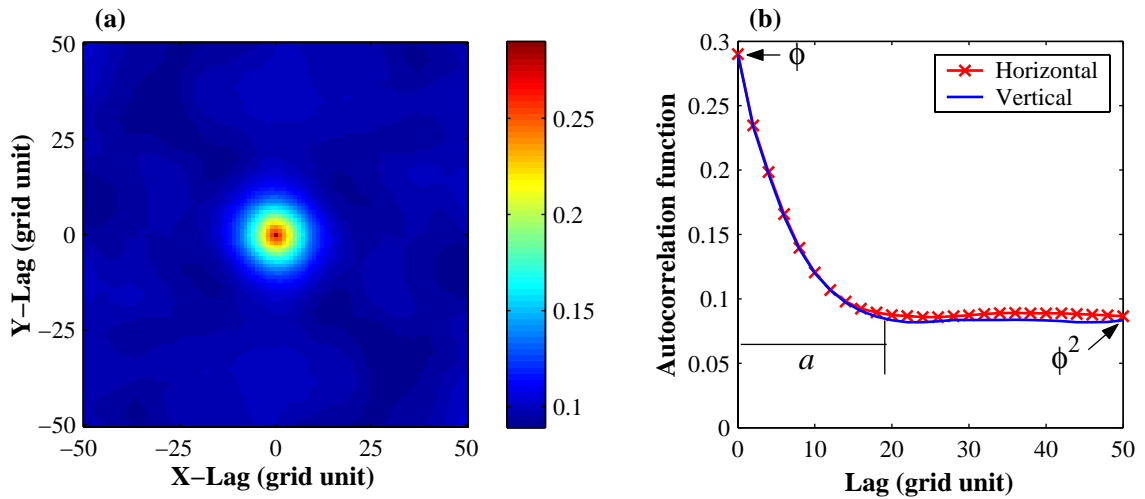


Figure 4.3: (a) 2D autocorrelation function from the binary image. (b) Horizontal and vertical autocorrelation functions (1D). The autocorrelation length (a) of this sample is about 20 grid units.

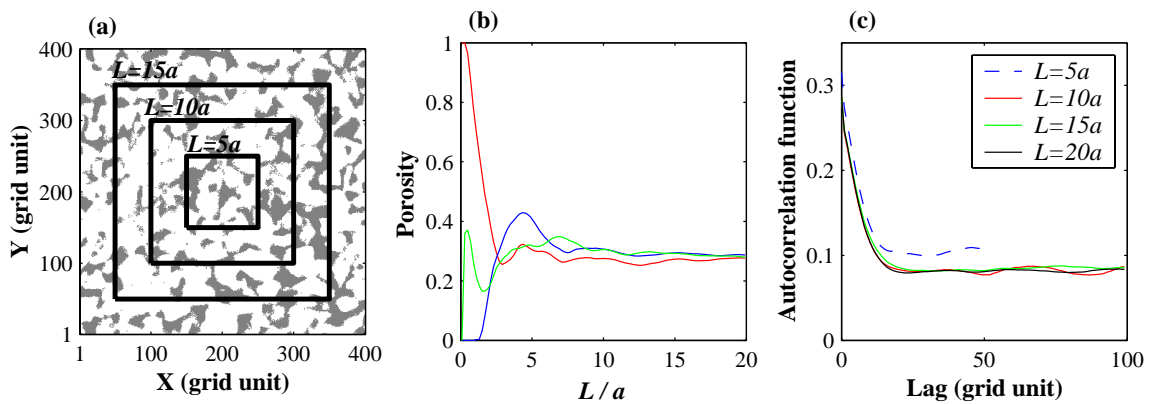


Figure 4.4: (a) Different square areas from a thin section image for estimating statistical parameters. a is the autocorrelation length. (b) Porosity given by different lengths of squares. Different lines denote different locations of the squares. (c) Autocorrelation functions from squares of different areas. When L is greater than $10a$, autocorrelation functions from different sized squares are very close to each other.

When a rock is homogeneous, the statistical averages can be replaced by volume averages. If it is also isotropic, the volume (3D) averages can be replaced by surface (2D) averages. Under these conditions, the use of thin sections can be justified. Estimation of statistical parameters from images requires the choice of image lengths much larger than the correlation length. Otherwise the estimates are not statistically representative and are affected by local fluctuations. This is shown in Figure 4.4. Squares of different sizes were chosen from the binary image for the calculation of the statistical parameters. Porosity is shown in Figure 4.4(b). Lengths of the squares (L) are normalized by the autocorrelation length (a). Different lines denote different choices of the origin point of the square. When L/a is very small, porosity fluctuates. However, it converges to a constant value that is very close to laboratory measurement when L/a is greater than 10. The autocorrelation function in Figure 4.4(c) also shows behavior very similar to that of porosity. When the size of the binary image (L) is greater than $10a$, we can safely assume it is a representative elementary area. We also tested the isotropy of rock samples by comparing these statistical parameters of two orthogonal thin sections from one core sample. Both porosities and autocorrelation functions from two orthogonal thin sections are very close to each other. Thus the assumptions of isotropy and homogeneity are valid with our samples. We used thin section images whose sizes are no less than $20a$ for accurate estimation of statistical parameters: porosity and autocorrelation function. All porosities from thin sections show good agreement with laboratory measurements and the differences are less than 2%. This porosity estimation should be carefully performed since all following procedures are strongly dependent on the estimated porosity.

4.3.3 2D to 3D Porous Media by Stochastic Simulation

From image processing of thin section data, we obtained spatial statistical parameters of the binary image – porosity and autocorrelation function. To simulate stochastic realizations of 3D porous media, we used a geostatistical approach (Deutsch and Journel, 1998). We first performed variogram modeling. The variogram has nearly the same meaning as the autocorrelation function, except that it reflects the dissimilarity of data in spatial distribution instead of similarity.

The variogram is given by (Deutsch and Journel, 1998),

$$\gamma(\mathbf{h}) = \frac{1}{2N(\mathbf{h})} \sum_{i=1}^{N(\mathbf{h})} [f(r_i) - f(r_i + \mathbf{h})]^2 \quad (4.7)$$

where $N(\mathbf{h})$ is the number of pairs of data locations with a vector \mathbf{h} apart. The variogram of the thin section image is related to the autocorrelation function as follows:

$$\gamma(\mathbf{h}) = A(0) - A(\mathbf{h}). \quad (4.8)$$

The raw experimental variogram computed from the images using Equation 4.7 is then modeled by an exponential function to ensure positive-definiteness of the variogram model. The exponential variogram is given as follows:

$$\gamma(\mathbf{h}) = c \left[1 - \exp\left(-\frac{3\mathbf{h}}{a}\right) \right] \quad (4.9)$$

where a is called as “practical range” in geostatistics. Figure 4.5 shows variograms along different directions and the variogram model. The variogram model is the best fit for both X- and Y-variograms with a nonlinear least-squares data fitting by the Gauss-Newton method.

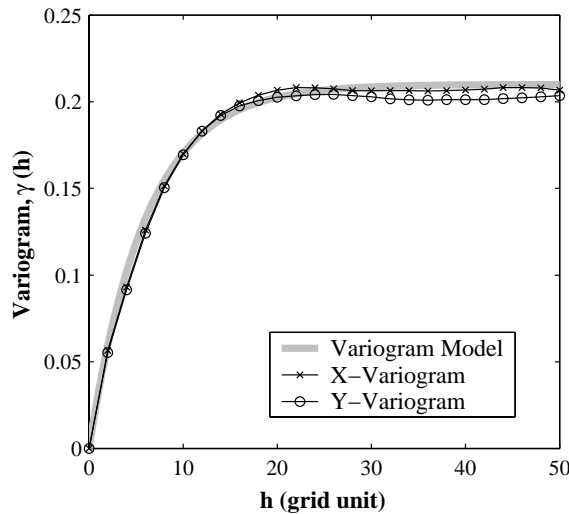


Figure 4.5: Variogram modeling. Circles denote X-variogram (horizontal) and squares Y-variogram (vertical). Gray solid line is a variogram model using an exponential function.

Using the variogram model, multiple realizations of 3D porous media were simulated conditioned to the thin section. Since our image is represented by an indicator random function, $f(r)$, indicator-based simulation algorithms are most appropriate and straightforward. The algorithm was the sequential indicator simulation (SISIM) from the geostatistics software library (GSLIB) by Deutsch and Journel (1998). In this algorithm, all the nodes in the 3D cube are visited along a random path. At each node, a local conditional cumulative distribution function (*ccdf*) for $f(r)$ is estimated. This *ccdf* is conditioned to the 2D image as well as all previously simulated nodes along the random path. The estimation of the *ccdf* is done by indicator kriging. A value for $f(r)$ is drawn from the local *ccdf*. This value is retained as conditioning data and we proceed to the next node along the random path. When all the nodes are visited, we have one realization of the 3D binary field with the correct spatial statistics.

Figure 4.6 shows a simulated 3-D porous medium. Since our implementation requires a periodic boundary condition along the flow direction, the same conditional data from the thin section image were included at both ends of the cube. Then we do not need to mirror the 3D cube. It saves memory and calculation time. We generated no less than 10 different realizations of simulated 3-D cubes for each thin section. We then calculated statistical parameters of the simulated 3D porous media to verify that the simulated realization has the same statistical properties. Three 2D cross sectional areas were chosen from a simulated realization (Figure 4.7). The variograms of three cross sections are very close to the variogram model as seen in Figure 4.7(b). Hence, as expected, the simulated 3D realization has the same statistical properties as the thin section.

4.3.4 Permeability Estimation by Lattice-Boltzmann Flow Simulation

Since the stochastically simulated 3D porous media from thin section have quite complex geometry and also may contain statistical noise from the simulation, a robust flow simulation technique, which can not only handle this complexity, but also can give reliable calculation results, is essential. The Lattice-Boltzmann method for numerical flow simulation can be an excellent choice, since it can handle complex pore geometry without any modifications or simplifications (Ladd, 1994b; Spaid and Phelan, 1997; Bosl et al., 1998; Keehm et al., 2001).

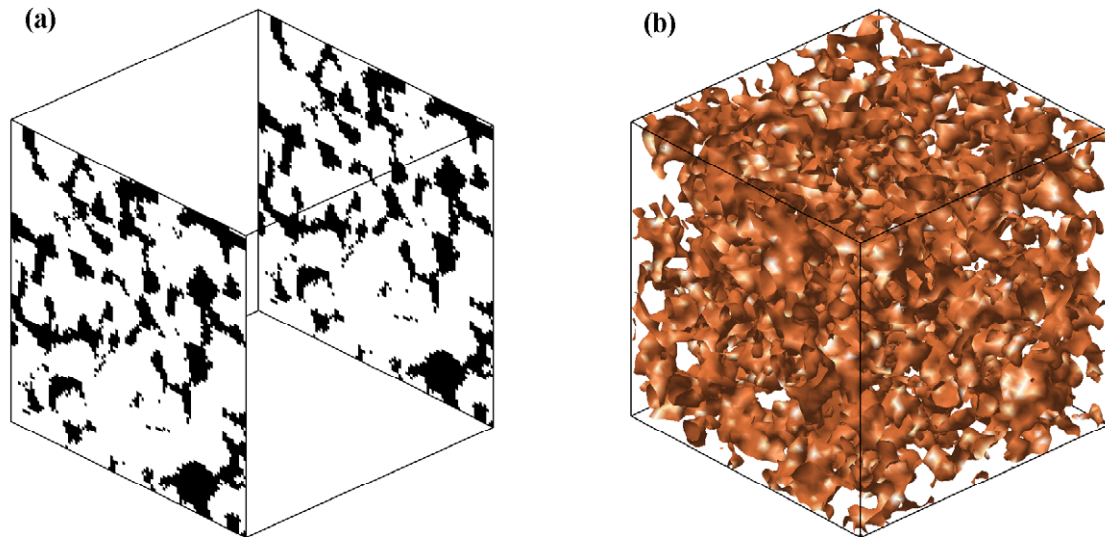


Figure 4.6: (a) Conditional data for the sequential indicator simulation. The slices on both ends of the cube have 100×100 pixels and are obtained from the binary image of a thin section. (b) Isosurface plot of a 3D realization by the sequential indicator simulation.

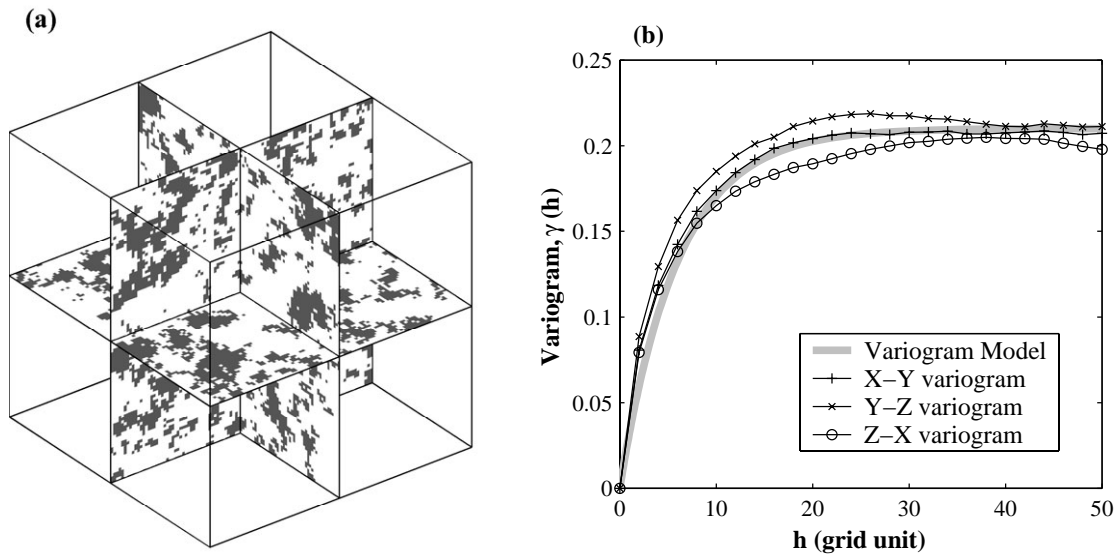


Figure 4.7: (a) Cross-sectional areas of a simulated 3D realization. Three areas (XY, YZ and ZX planes) were chosen to verify that the simulated 3D porous media have the same statistical properties. (b) Variograms of those three areas. The variograms of simulated areas of 3D porous media are close to the variogram model.

Figure 4.8 shows the distribution of local fluid flux after the flow simulation. Local fluxes are calculated by Equation 3.7, and normalized by the maximum value. The effective flow paths and their complexity can be clearly seen in cross-sectional plots.

The permeability, which is a macroscopic property, can be calculated by Darcy's law (Equation 2.3). Figure 4.9 shows estimated permeability for Sample #1. The laboratory measurement is shown as a square symbol. Cross symbols are permeability estimations from stochastic realizations, and a circle is the averaged value. In this sample, the porosity difference between the thin section and the laboratory measurement is about 1.3%. The average permeability is almost identical to measurement value.

Results from all seven thin sections can be found in Table 4.1. Porosity and permeability estimations are reasonably consistent with laboratory measurements. Figure 4.10 shows calculated permeability for all seven thin sections used in this study. They are also compared to the laboratory measurements. All estimated values reflect very good agreement with laboratory measurements. The correlation coefficient between estimated permeability and laboratory measurement is 0.96, which implies excellent estimations.

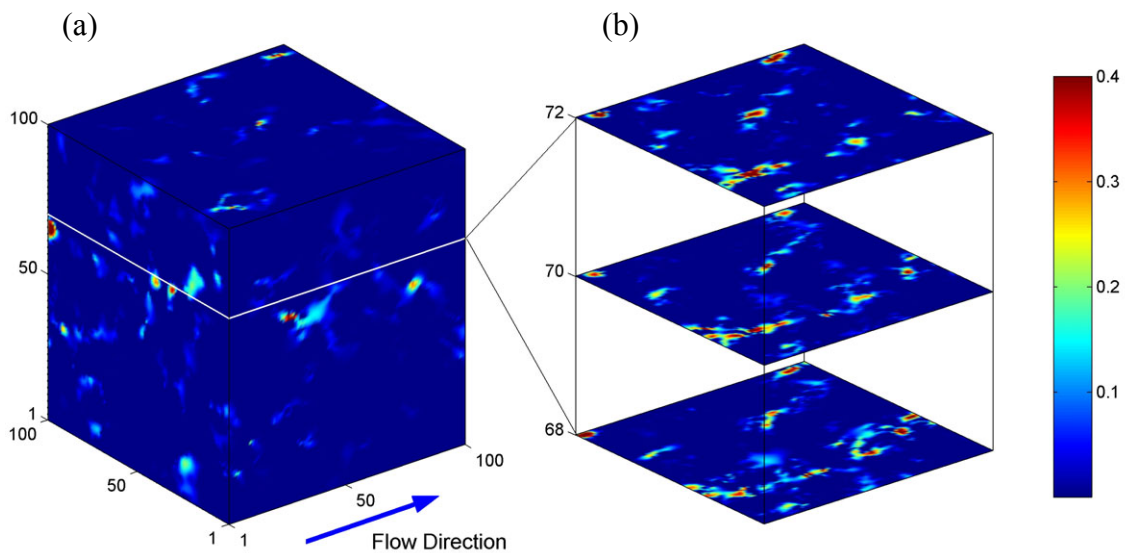


Figure 4.8: (a) Local mass flux by the Lattice-Boltzmann flow simulation. Flux values are normalized by the maximal value. (b) Cross-sectional plots of the high flux area.

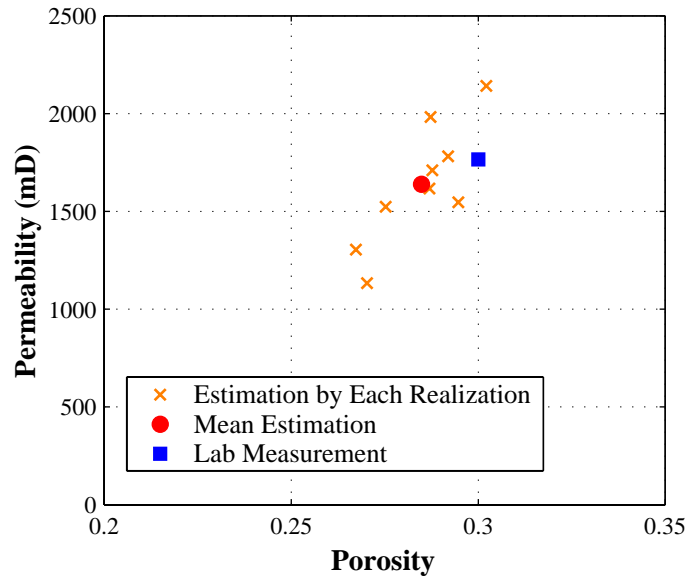


Figure 4.9: Permeability estimation by the Lattice-Boltzmann flow simulation on stochastically generated 3D porous media. Laboratory measurement is shown as a square. Estimated permeabilities from different 3D realizations are in cross symbols and a circle is an averaged permeability estimation.

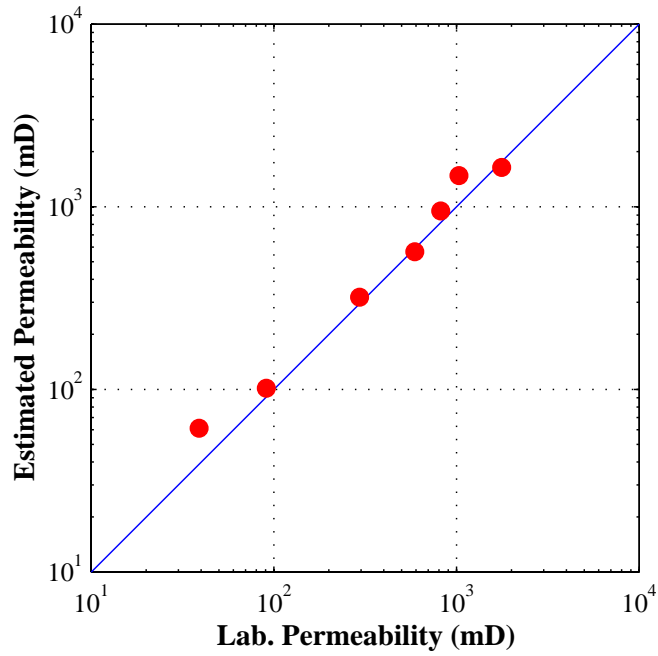


Figure 4.10: Calculated permeabilities versus laboratory measurements for all seven samples used in this study.

4.4 Comparison to the Specific Surface Area Method

Kozeny and Carman (Carman, 1961) provided a theoretical relation between fluid permeability and other measurable properties of porous media (Equation 2.7). Walsh and Brace (1984) derived a similar formula relating permeability to porosity, formation factor (F) and the specific surface area:

$$\kappa = \frac{\phi^2}{c F^2 S^2}, \quad (4.10)$$

where c is another geometric factor, with $c = 2$ for circular pores and $c = 3$ for flat cracks. This is a different version of Kozeny-Carman relation, where the formation factor replaces the tortuosity, which is hardly measurable.

Blair et al. (1993) suggested that Equation 4.10 would be useful for estimating permeability from thin sections. They extracted statistical parameters from thin section image analysis to obtain porosity and the specific surface area. They then estimated permeability using Equation 4.10,

To determine the specific surface area, Blair et al. (1993) used a relationship between the specific surface area and the spatial correlation functions (Berryman, 1987; Berryman et al., 1987). Under the assumption of homogeneous and isotropic media, the angular-averaged correlation function is given by

$$A(h) = \langle f(r)f(r+h) \rangle, \quad (4.11)$$

where $h = \|\mathbf{h}\|$. Berryman showed that the specific surface area S , can be expressed as (Berryman, 1987),

$$S = -4 \times A'(0), \quad (4.12)$$

where $A'(0)$ is the slope of the two-point correlation function at the origin. Although Equation 4.12 is an exact analytical derivation, the subtlety comes when dealing with real discretized image data. Since we have discrete binary images from thin sections, the first derivative becomes a finite difference. In order to have a good approximation, we need to have very high-resolution images. This will usually involve a more sophisticated device, such as SEM, and more computing time for image processing. As the resolution increases,

the whole scanning area becomes smaller and the porosity prediction becomes more difficult, since the small scanning area may not have the representative elementary area, as discussed in the previous section (see also Figure 4.4). In addition, the estimation of the specific surface area is sensitive to the image resolution. If the resolution is too low, the specific surface area is underestimated and the permeability will be overestimated, and vice versa. Blair et al. (1993) suggested that the optimal resolution for obtaining the specific surface area is such that the autocorrelation length (a) is two orders of magnitude larger than the discrete grid size (dx), *i.e.*, $a \cong 100 \times dx$. However, even a slight change in resolution will significantly change the permeability estimation. Other parameters, the formation factor (F) and the geometric factor (c), need to be obtained from sources other than the 2D images, such as laboratory measurements or empirical relations.

We applied Blair et al.'s (1993) method to our data set for comparing our method to theirs. Figure 4.11 shows the schematic procedures of their method. Since this method requires much higher resolution for determining the specific surface area, the total size of the binary image (Figure 4.11) was smaller than the one we used (Figure 4.2). For porosity calculation, we used lower resolution images with larger total areas for better statistics. Figure 4.11(d) shows the angular averaged autocorrelation function and estimation of $A'(0)$. We also use the empirical relation $F = \phi^{-2}$ for the formation factor, and the geometrical factor (c) is assumed to be 2, as Blair et al. (1993) suggested for sandstone samples. After obtaining all parameters, the permeability can be calculated by Equation 4.10. Predicted values are shown in Figure 4.12 as squares. Red dots denote estimated permeability by our method for comparison. Both show a reasonably good match, but our method gives better agreement. More importantly, our method does not require any assumptions regarding the formation factor and the geometrical factor.

Another interesting observation in Figure 4.12 is that two methods show very similar trends in permeability estimations. When one overestimates permeability, the other also overestimates. This implies that the error trend must have come from a common source: the statistical parameters from the thin section image. Since the thin section cannot represent the whole rock exactly, there are always small fluctuations between the statistical parameters of the image and those from the whole rock. Although both

methods use the autocorrelation function as one of statistical parameters, the specific surface area method uses the first derivative of the two-point correlation function, which is more sensitive to the statistical errors from thin section images. On the other hand, our method uses the autocorrelation function itself and may be less sensitive to the statistical errors, because the derivative is not involved. Thus, our method is more robust against the errors from thin sections.

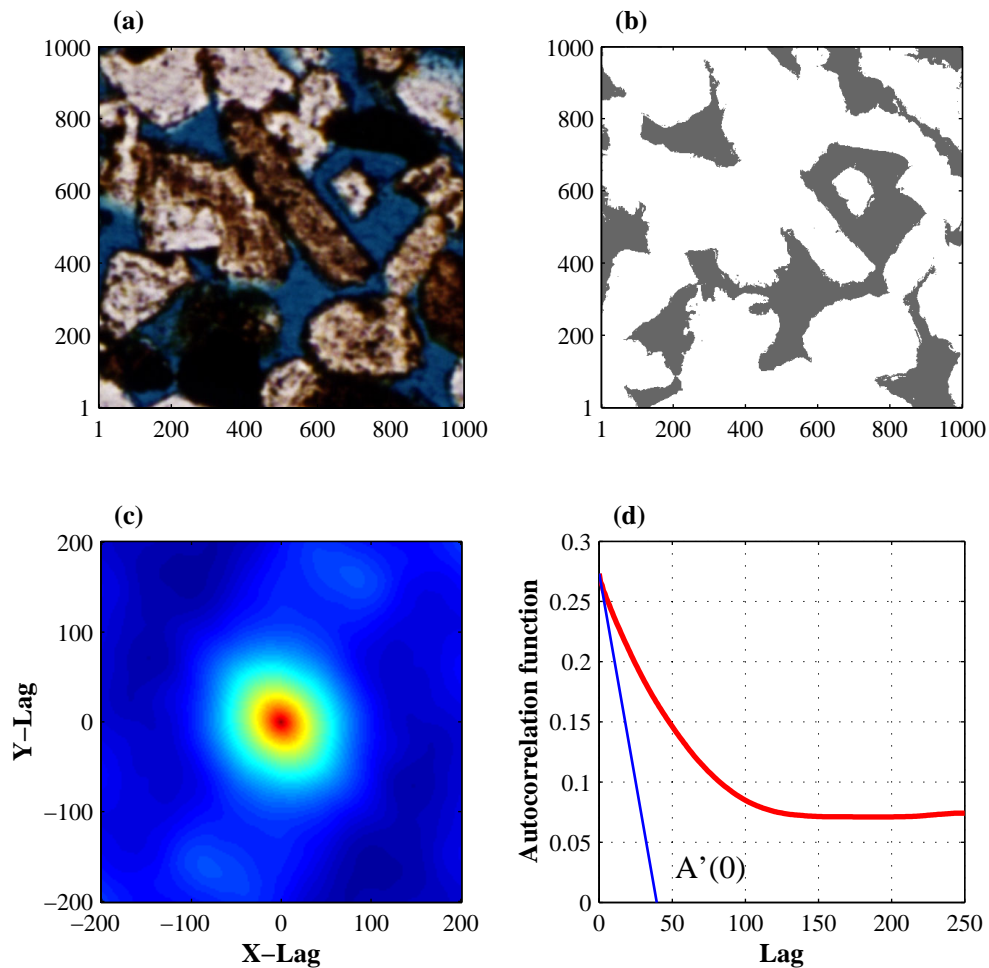


Figure 4.11: Steps of estimating the specific surface area as suggested by Blair et al. (1993). (a) High resolution scanned image. (b) Binary image. (c) Autocorrelation function. (d) Angular averaged autocorrelation function and estimation of the slope near the origin.

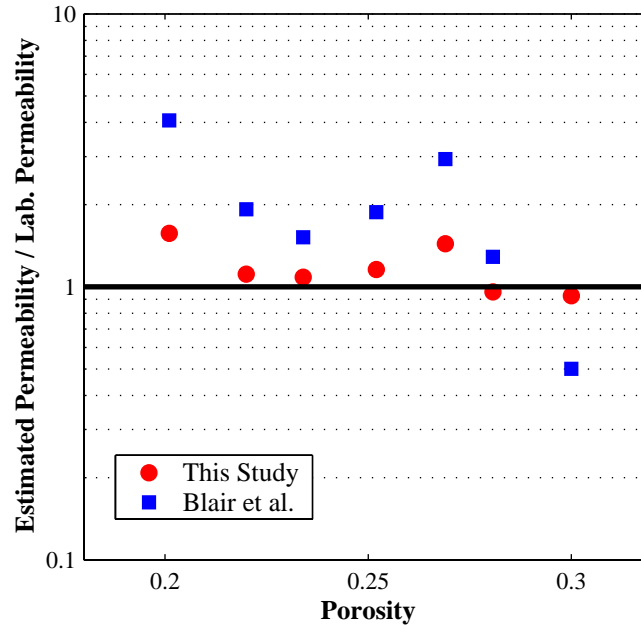


Figure 4.12: Permeability estimations from the seven thin sections by Blair et al.'s (1993) method (squares) and ours (dots).

4.5 Discussion

Although techniques and devices have been improved and permeability measurement has become a standard procedure in the laboratory, it is still a time-consuming task. Therefore there have been many attempts to relate permeability to other easily measurable parameters. The well-known Kozeny-Carman relation is one of these attempts. Since permeability is a function of only pore geometry, finding parameters characterizing the pore geometry is the main task for the permeability estimation. Porosity, tortuosity, specific surface area, and characteristic pore length are a few suggested parameters. Some parameters are very difficult to measure and are sometimes replaced by other parameters, such as formation factor.

Obtaining these characterizing parameters from thin sections is of great value, since thin sections are relatively easy to obtain and are widely available. In homogenous rocks,

the porosity and autocorrelation function (the two-point correlation function) are the most widely used parameters to characterize the pore geometry. Permeability estimation from these statistical parameters can be categorized into two groups; using theoretical relations usually based on the Kozeny-Carman relation (Group I), and performing flow simulation on stochastically generated 3D porous media (Group II). Both groups have shown relatively good results on permeability estimation, especially for sandstone samples (Adler et al., 1990; Blair et al., 1993; Liang et al., 1999; Hilfer and Manwart, 2001).

Group I usually requires parameters that cannot be easily obtained from thin section images. The estimation of the specific surface area based on first derivative of the two-point correlation function is sensitive to the statistical errors. Group II usually uses the truncated Gaussian method, which has difficulty determining the right threshold. Indicator-based simulation techniques may be more natural and straightforward for stochastic simulation of 3D porous media reconstruction. Moreover it is almost impossible to incorporate multiphase and anisotropic media in the truncated Gaussian method, while simulations based on indicator algorithms can handle easily. Flow simulation techniques are also important for permeability estimation. The Lattice-Boltzmann algorithm handles complex pore geometry very well, even with statistical noise from stochastic simulation.

Although we have very good agreement between our permeability estimation and laboratory measurements, our data are quite limited: thin sections from fairly clean and well sorted sandstone samples. If a porous medium has two distinct pore scales (e.g., a rock with microporosity), the autocorrelation function may not generate 3D porous media close enough to the real rock. We need more complicated variogram models that can capture these different length scales in pore geometry. Even if we can have good 3D realizations of porous media, flow simulation will be also very difficult with multiple, highly disparate scales. We need to have a very fine grid structure to resolve the micropores, and the size may be too big to fit on even modern computers. Multiscale analyses or parallel computing can be future solutions for this problem. More statistics from thin sections will help construct better 3D realizations. In addition to porosity and the two-point correlation function, many studies try to incorporate more statistical

parameters in 3D reconstruction, such as the lineal-path function (Yeong and Torquato, 1998a and 1998b; Manwart et al., 2000; Torquato, 2001), the chord-distribution function (Roberts and Torquato, 1999), and multipoint correlation functions (Torquato and Stell, 1982; Caers and Xianlin, 2002). Reconstruction of textures from multi-level wavelet coefficients and steerable directional filters is another promising approach (Simoncelli and Portilla, 1998; Heeger and Bergen, 1995).

4.6 Conclusions

The Lattice-Boltzmann flow simulation on reconstructed 3D porous media by stochastic simulation is a good combination for permeability estimation from thin sections. It seems quite remarkable that direct flow simulations on stochastically generated 3D porous media provide very good agreement with laboratory measurement over such a wide range of permeability.

We found that determining the porosity and the autocorrelation function from thin sections should be carefully performed, since all following procedures are strongly dependent on these statistical parameters. For better statistics of these parameters, the image size should be greater than 10 times the autocorrelation length ($L \geq 10a$). Another important scale factor is the grid spacing (dx). To accurately estimate permeability, the Lattice-Boltzmann flow simulation requires at least 10 grid points within a characteristic pore length ($dx \leq a/10$). The sequential indicator simulation successfully generates multiple realistic 3D porous media and does not seem to be very sensitive to the statistical error. The Lattice-Boltzmann flow simulation shows its strength in handling these complicated 3D pore geometries. The estimated permeability shows very good agreement with laboratory measurements.

Permeability prediction by the specific surface area from two-point correlation (Blair et al., 1993) gives good estimates of permeability. However, estimation of the specific surface area is dependent on image resolution and is very sensitive to changes in resolution. The error in estimation is about 2-3 times higher than with our proposed method. Both methods show similar trends of the deviation from laboratory measurements. This implies that the statistical parameters from the thin section do not

fully represent the whole rock. Practically this cannot be avoided. The results indicate that the proposed method is less sensitive to the small discrepancy between thin sections and rock samples. In addition, our method does not require any other empirical or free parameters.

The most important part of this method is constructing 3D porous media from thin sections, because it is a difficult and non-unique problem. However, predicting permeability from thin sections has great importance for oil reservoirs and underground systems since thin sections are widely available and are very easy to obtain. Moreover, constructed 3D porous media, if reasonably representative for the whole rock, will be sharing objects for other physical property simulations. Relative permeability estimation from thin sections will have much bigger impact on many fields of science and engineering. We already showed several recent attempts to use higher statistics in the previous section, such as multipoint correlation functions. Stochastic reconstruction of 3D porous media from thin sections using higher statistics is a very challenging but essential module for physical process simulations of rocks.

Chapter 5

Diagenesis Modeling and Transport Properties of Rocks

5.1 Abstract

This chapter presents a model to simulate diagenesis in porous media and its impact on transport properties. The model combines the Lattice-Boltzmann flow simulation with pore filling mechanisms. Eight different mechanisms related to diagenesis controlled by fluid flow, were tested. The transport properties (permeability and electrical conductivity) were calculated during the evolution of pore geometry. After investigating the transport properties in each mechanism, four distinct groups were categorized: (1) grain boundary group, (2) low-flux group, (3) high-flux group and (4) random filling group. In the grain boundary group, the diagenesis or pore alteration happens first at pore nodes with the neighboring grain nodes. In low- and high-flux groups, the alteration occurs first at pore nodes with the lowest and the highest fluxes, respectively. The random filling group randomly fills up the pore node with grains.

The boundary group showed very high correlation with empirical relations such as the Kozeny-Carman relation and Archie's law. Since the empirical relations are based on statistics derived from real data, the grain boundary can be regarded as a realistic

approximation to the diagenetic process of sedimentary rocks. The low-flux group showed very small changes in permeability even in the latter stages of the diagenetic process. For this case, electrical conductivity decreases faster than permeability as porosity decreases. The high-flux group is the most efficient in reducing permeability, since pore filling starts at the region with high flux. With a few percent of porosity reduction, the effective fluid flow can be practically negligible. Random filling showed an intermediate behavior between high-flux and boundary groups. At early pore filling stages, the permeability decreases very rapidly, followed by a trend very similar to the boundary group.

The main bottleneck of simulating evolving pore geometry lies in the slow convergence of the Lattice-Boltzmann flow simulation when the pore geometry is close to percolation. One of the innovations in this work was to change the iterative Lattice-Boltzmann algorithm of the flow simulation to a sparse matrix form. The matrix method shows better convergence when the pore geometry is close to percolation. Another benefit of the matrix version is predicting convergence before the actual flow simulation, by calculating a condition number of the kernel matrix. This modeling enables us to efficiently explore and delineate diagenesis mechanisms and their effects on transport properties of porous media. Even though this model is too simple to represent a realistic diagenesis in sedimentary rocks, the same framework can be used with more complex and realistic diagenetic processes.

5.2 Introduction

Diagenesis is defined as chemical, physical, and biological changes that affect sediments after its initial deposition, including turning the sediments into rocks. During diagenesis, the porosity of the sediment typically decreases. Other physical properties, such as seismic velocity, permeability and electrical conductivity, are also changed. One main focus of rock physics is to find quantitative relations between porosity and other physical properties during diagenesis.

Although some recent works (Rimstidt and Barnes, 1980; Nagy and Lasaga, 1991) suggest that reaction-dependent kinetics control the diagenesis of most sedimentary rocks,

a real diagenetic process would involve both reaction-dependent and flow-dependent mechanisms. An important motivation for flow-dependent diagenesis is to understand the formation of gas hydrates. Xu and Ruppel (1999) suggest that all natural gas hydrates systems are advection dominant, which means they are in the flow-dependent diagenesis regime. Results of these studies imply that flow-dependent diagenetic mechanisms would be more appropriate to explain the formation of gas hydrates. Xu and Ruppel (1999) also provide results of numerical simulations using mass and energy balances of the hydrate layers in the ocean, treated as an effective medium. They do not take into account the changes in pore geometry. Aharonov and Rothman (1996) present a dynamic model for growth of pore-scale structures in sedimentary rocks. They focus on a reaction-dependent mechanism to relate the growth of pore structures to fractal dimensions (Aharonov and Rothman, 1996; Aharonov et al., 1997).

This chapter will present procedures and results of flow-dependent diagenetic modeling. Our model is a very efficient way to investigate the relation between evolving pore-geometry and transport properties. Implementation of a sparse-matrix version of the Lattice-Boltzmann flow simulation method will follow. This variant avoids the convergence problems of standard iterative version of the Lattice-Boltzmann flow simulation in low porosity and low permeability rocks.

5.3 Diagenesis Modeling

The modeling of diagenesis consists of flow simulation and pore-filling processes iterated in tandem. We start with a digital structure of pore geometry defined on a 3D grid. Then a numerical flow simulation using the Lattice-Boltzmann method (LBM) is performed on this digital pore geometry. Diagenesis or cementation is simulated by changing selected pore nodes into grain nodes. We will call this “pore filling”. The criteria for pore filling are based on the fluid flux distribution obtained from the numerical flow simulation step. We then obtain a new pore geometry, which will be used as an input for the next flow simulation. This procedure is repeated until porosity or effective permeability reaches zero. The LBM is used in this modeling because it can handle very complicated pore geometry, and is also sensitive enough to detect the change

of fluid flow due to small changes in pore structures (Chopard and Droz, 1998). The LBM is a very appropriate tool for modeling flow-dependent diagenesis. To explore and simulate a wide range of diagenetic kinetics, eight different pore-filling mechanisms were chosen. The pore nodes to be filled were determined by [1] grain boundary (filling pore nodes near grain boundary), [2] low flux regions (filling pore nodes with low flux), [3] high flux regions (filling pore nodes with high flux), [4] low flux regions among boundary nodes (filling nodes near grain boundary, but one with low flux first), [5] high flux areas among boundary nodes (filling nodes near grain boundary, but one with high first), [6] low flux regions weighted by neighboring grain nodes (filling nodes with low weighted flux), [7] high flux regions weighted by neighboring grain nodes (filling nodes with high weighted flux), and lastly [8] random pore filling. The weighted flux means the flux multiplied by a weighting factor, which is related to the effect of neighboring grain nodes. In this study, the weighting factor is simply the sum of the number of nearest neighboring grain nodes. This is an attempt to incorporate both the grain boundary and the fluid flux effects into pore-filling decision, although it is not very realistic. In the case [7], for example, the pore filling occurs at the pore nodes with high flux area, but preferably near the grain boundary. In the end, we found no difference in permeability and electrical conductivity of the cases [2], [4], and [6]. The cases [3], [5] and [7] also show almost no difference.

For each pore-filling scheme, two transport properties of rocks, permeability and electrical conductivity, were calculated at every incremental stage of the evolving pore geometry. The eight different diagenetic simulation schemes were applied to synthetic and real digital pore geometry including a sinusoidal tube model, Finney's random sphere-pack model (Finney, 1970) and a digital Fontainebleau sandstone obtained by X-ray microtomography. The three digital rocks are shown in Figure 5.1. Interestingly, the three different pore structures showed very similar behaviors in permeability and electrical conductivity. Moreover, the eight different mechanisms of pore filling can be classified into four groups: grain boundary (Group I), low-flux (Group II), high-flux (Group III) and random filling (Group IV). Figure 5.2 shows a cross-sectional view of the flux distribution of fluid flow and filled pore nodes for the four groups of pore filling

mechanisms at an intermediate stage of the process. The arrows are local fluid velocities calculated at the initial stage, and the red pixels denote the pore filling areas according to each group.

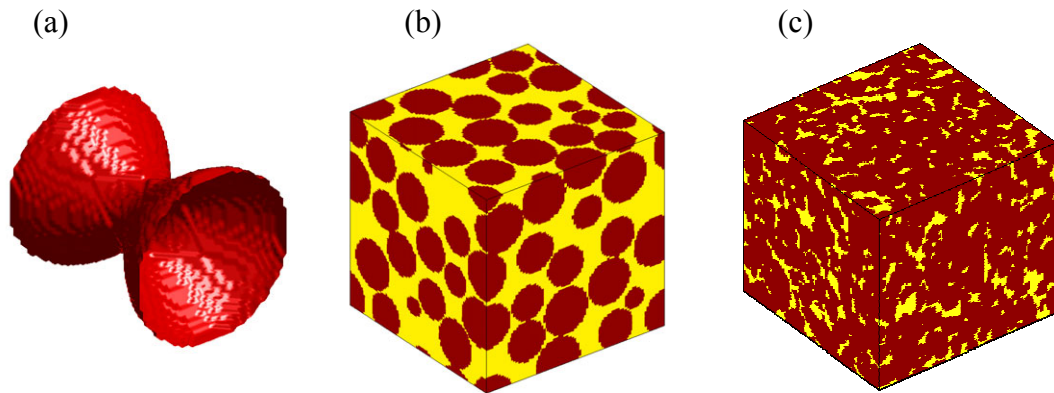


Figure 5.1: Three digital pore structures used in the study. (a) A sinusoidal tube, (b) a random dense pack of identical spheres (Finney's pack) and (c) a digital Fontainebleau sandstone sample.

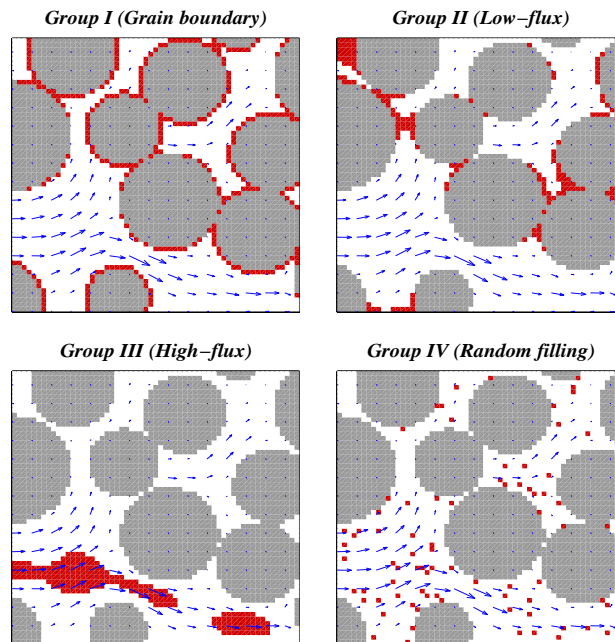


Figure 5.2: Flux distribution of fluid mass (arrows) in a sphere-pack model and filled pore nodes (red area) according to the different pore-filling mechanisms. Gray area indicate the initial grains.

Figure 5.3 shows the computed permeability as a function of porosity for each pore-filling mechanism. The black line with dots denotes an empirical relation (Mavko et al., 1998), $\kappa/d^2 \propto \phi^3$, where κ denotes permeability, ϕ porosity, and d is a characteristic grain or pore dimension. The power of 3 appears to be appropriate for very clean materials. Group I shows a trend very close to the empirical relation. Since the relation was derived from theoretical models and real data, it represents the average behavior of most sedimentary rocks. This implies that the grain boundary may be the most important factor in diagenesis of many sedimentary rocks. As we might expect, Group II shows the least decrease in permeability as porosity decreases, since low-flux areas do not affect permeability whether they are filled or empty. Group III shows the most efficient mechanism for reducing permeability. After only a few percent of pore nodes were filled, the permeability practically dropped to zero. Group IV shows intermediate behavior between Group I and III. Permeability rapidly decreases during early pore filling stages, but after that it slows down and exhibits a curve similar to that of Group I.

Figure 5.4 shows the electrical conductivity vs. porosity of the same pore structures used in Figure 5.3. The low and high flux-related mechanisms show very similar results to those of permeability. However, Group I and IV show reversed trends in permeability and electrical conductivity, *i.e.* as porosity decreases, the electrical conductivity of Group I decreases faster than that of Group IV. Since the random filling mechanism scatters the filled pore nodes, it effectively increases the surface area with very little change in porosity. Thus, the permeability decreases much faster than the electrical conductivity. We can observe this effect by the fact that the electrical conductivity of Group IV decreases almost linearly as porosity decreases. The thin black line denotes Archie's law, $F = b\phi^m$, where $\sigma_r = \sigma_w/F$, b is a proportionality coefficient, F is the formation factor and σ_r , σ_w are the electrical conductivity of the saturated rock and brine, respectively. The power m usually varies in the range of 1.5 to 2.5 in most sedimentary rocks. In Figure 5.4 the value of 2 was chosen, which is known to be appropriate for very clean sandstone rocks (Guéguen and Palciauskas, 1994). As was the case for permeability, Archie's law is very close to the electrical conductivity of Group I. This also supports the idea that the grain boundary is a very important controlling factor in the diagenetic process in most

sedimentary rocks.

Because electrical conductivity measurements in the laboratory or *in situ* are relatively easy, the ability to deduce permeability from electrical conductivity is a major goal in many applications (Guéguen and Palciauskas, 1994). Therefore, it is important to compare both transport properties in each diagenetic mechanism. Wong et al. (1984) suggested the relation, $\kappa \propto F^{-n}$, and showed that there was good agreement with sintered glass bead data when $n=2$. Figure 5.5 shows the cross plot of permeability and electrical conductivity. Group I appears to be matched with a line of $n=1.5$. Group II has a smaller exponent than Group I. Group III and IV may have higher exponents than 3, which means that in these cases, the permeability is a lot more sensitive to changes in pore geometry than the electrical conductivity is.

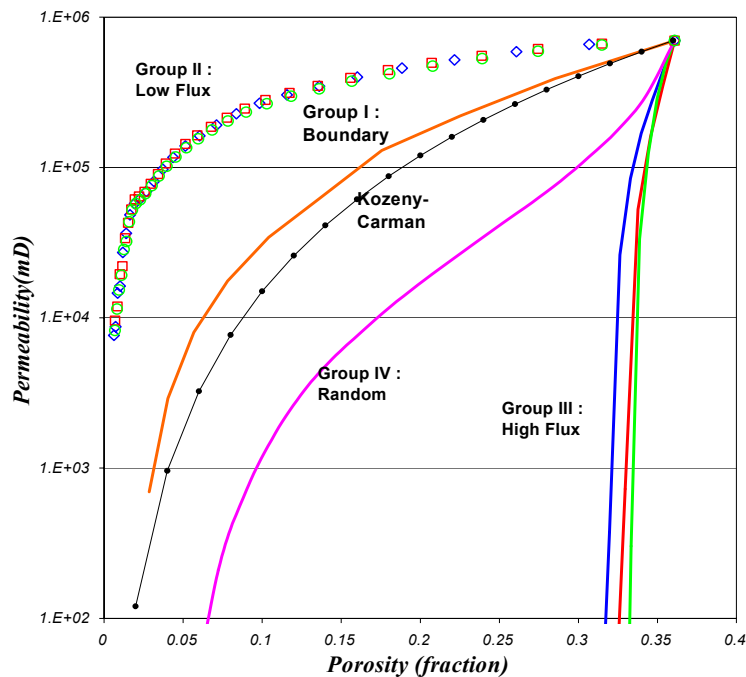


Figure 5.3: Permeability in different pore filling mechanisms.

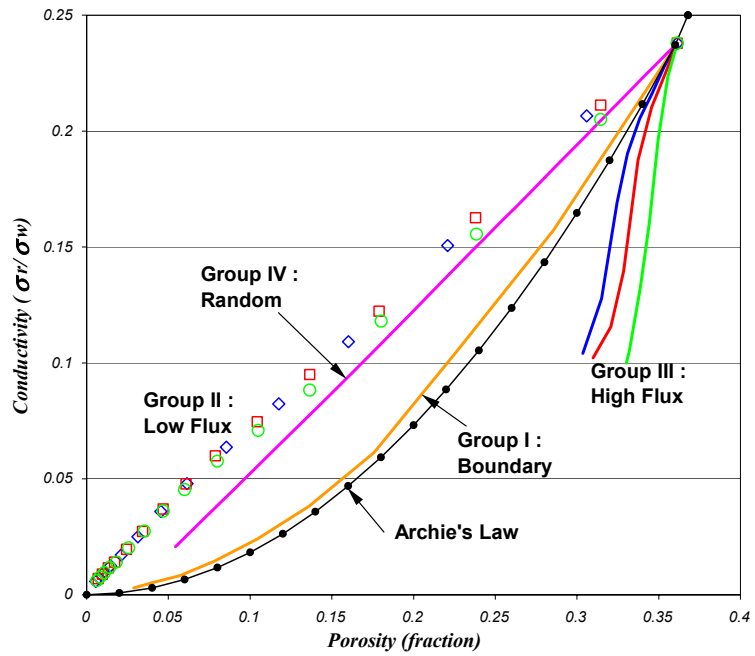


Figure 5.4: Electrical conductivity of different groups. Notations are same as in Figure 5.2.

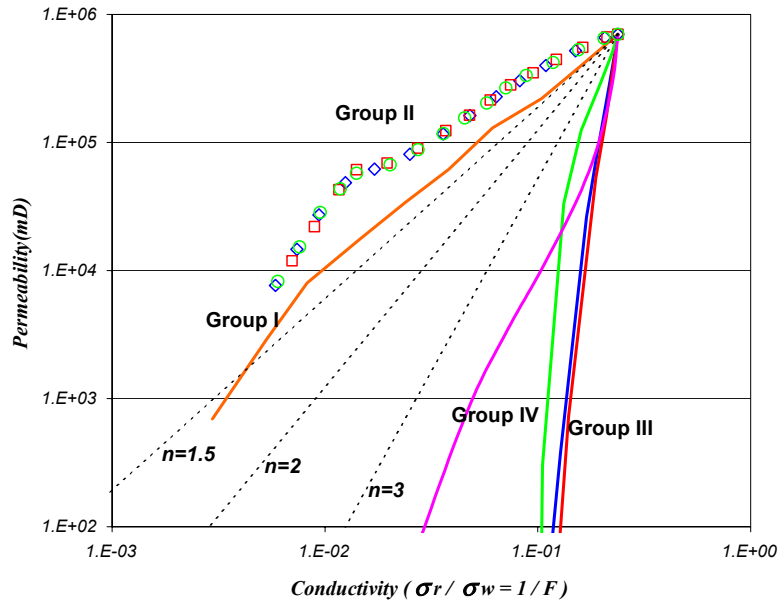


Figure 5.5: Cross plot of permeability and electrical conductivity. Dotted lines show the power laws ($\kappa \propto F^{-n}$) with $n=1.5, 2$ and 3 , respectively.

Figure 5.6 shows permeability and electrical conductivity as functions of porosity. In this figure, each group is plotted in a separated sub-figure. To compare the changes in permeability and electrical conductivity, we used the relation, $\kappa \propto F^{-2}$. Both values were normalized by their maxima; the results clearly show which decreases faster as porosity decreases. Permeability decreases faster with high flux and random filling groups. Electrical conductivity (F^{-2}) decreases faster with low flux group as porosity decreases. In boundary group, both have the same trends. These results can be very useful to determine the main diagenetic cause of a specific formation of rocks, if measurements of permeability and electrical conductivity are available. On the other hand, it will allow us to determine preliminary trends of permeability as a function of porosity, if we know the main controlling diagenetic mechanism and electrical conductivity.

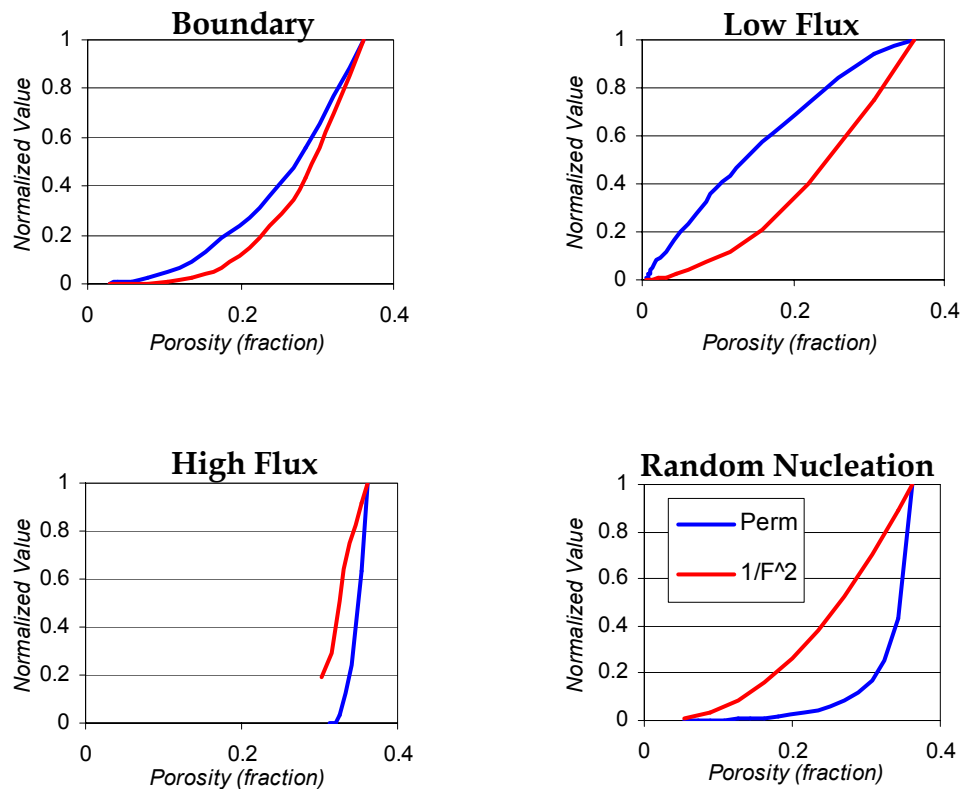


Figure 5.6: Permeability and electrical conductivity curves for each group. For comparison, $1/F^2$ is used for electrical conductivity values. Legend in the last subplot is the same for all subplots.

5.4 Lattice-Boltzmann Method with Matrix Inversion

During the diagenetic modeling, the pore space eventually loses the connectivity, *i.e.* percolation. When the evolving pore geometry is close to percolation, it is very difficult for fluid to flow through the pore geometry. As a consequence, the LBM flow simulation shows very slow convergence. This happens especially in the high-flux diagenesis mechanisms. Figure 5.7 shows an example of this problem. The calculated permeability oscillates because the pore geometry is difficult to flow through. There are two typical problems: (1) the simulation takes very long time, and (b) the simulation may converge to local minima. We found that the convergence problem can be bypassed by reformulating the iterative scheme into a matrix formulation as described below.

In order to form a matrix version of the Lattice-Boltzmann method, we need to recall the algorithm (see Chapter 3 for details).

- (1) Calculate mass and momentum.
- (2) Collision step: redistribute the mass.
- (3) Apply a force according to the pressure gradient (∇P).
- (4) Propagation step: the mass is shifted around the lattice.
- (5) Repeat (1)-(4), until steady state is reached.

Let n_i^k be the particle density in the i -direction at the k -th node, and \mathbf{c}_i is the velocity vector in this direction (Ladd, 1994a). Then the mass and momentum at node k , are

$$\rho^k = \sum_i n_i^k, \quad \mathbf{j}^k = \sum_i n_i^k \mathbf{c}_i. \quad (5.1)$$

The post-collision particle density distribution (Equation 3.6) can be expressed with a linear combination of the pre-collision particle density distribution,

$$\left[n_j^k \right]_{(t+1)} = \sum_j C_{ij} \left[n_j^k \right]_{(t)} \quad (5.2)$$

or in the matrix form,

$$\left[\mathbf{n}^k \right]_{(t+1)} = \mathbf{C} \left[\mathbf{n}^k \right]_{(t)} \quad (5.3)$$

where C_{ij} is a collision matrix and (t) denotes the time step. Now, we can combine all nodes into one column vector (\mathbf{N}) and express the equation as,

$$\begin{bmatrix} \mathbf{n}^1 \\ \mathbf{n}^2 \\ \vdots \\ \mathbf{n}^m \end{bmatrix}_{(t+1)} = \begin{bmatrix} [C_{ij}] & 0 & \cdots & 0 \\ 0 & [C_{ij}] & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & [C_{ij}] \end{bmatrix} \begin{bmatrix} \mathbf{n}^1 \\ \mathbf{n}^2 \\ \vdots \\ \mathbf{n}^m \end{bmatrix}_{(t)} \quad (5.4)$$

or

$$\mathbf{N}_{(t+1)} = \mathbf{B} \mathbf{N}_{(t)} \quad (5.5)$$

where \mathbf{B} is a block-diagonal matrix. At step (4) of the algorithm, we need to propagate this new mass distribution to neighboring nodes, *i.e.*, to permute the column vector \mathbf{N} . The permutation is easily implemented, simply by permuting the rows of the matrix. Let the permuted matrix be \mathbf{B}' . Then the whole algorithm turns into a simple matrix equation,

$$\mathbf{N}_{(t+1)} = \mathbf{B}' (\mathbf{N}_{(t)} + \mathbf{f}) \quad (5.6)$$

after including the force term (\mathbf{f}). To have the accurate non-slip boundary condition, the mass distribution should be evaluated after applying half of the force (\mathbf{f}_h), since the boundary is located in the middle of the line between the grain node and the pore node in a discrete grid. Details can be found in Ladd (1994a). Lastly, we can write the algorithm as the following matrix equation:

$$\mathbf{N}_{(t+1)} = \mathbf{B}' (\mathbf{N}_{(t)} + \mathbf{f}_h) + \mathbf{f}_h. \quad (5.7)$$

At steady state, $\mathbf{N}_{(t)} = \mathbf{N}_{(t+1)}$, giving

$$(\mathbf{B}' - \mathbf{I})\mathbf{N} = -(\mathbf{B}' + \mathbf{I})\mathbf{f}_h, \quad (5.8)$$

or equivalently,

$$\mathbf{A} \mathbf{x} = \mathbf{b}, \quad (5.9)$$

where $\mathbf{A} = (\mathbf{B}' - \mathbf{I})$, $\mathbf{x} = \mathbf{N}$, and $\mathbf{b} = -(\mathbf{B}' + \mathbf{I})\mathbf{f}_h$. In solving this system of linear equations, the bi-conjugate-gradient stabilized (BiCGS) method was used. We tested the new algorithm and the results were identical to those of the iterative method. The new method was

applied to the same model that exhibited a problem in Figure 5.7. The calculation time of the matrix method is equivalent to 1000 iterations of the iterative method.

We used the matrix method for the diagenetic modeling with high flux related mechanisms. In that case, the matrix version converged to the real flow distribution faster than the iterative method. However, there are some drawbacks in the new method. First, it requires much more memory (about ten times more) than the iterative method. Secondly, the calculation times of the two methods were not much different in most cases. In high porosity and high permeability rocks, the iterative method converges faster than the matrix version, since the model is easy to converge and the matrix version requires the extra calculation cost of setting up a large sparse matrix. For better performance of the matrix version, we may use a carefully designed pre-conditioner for the matrix inversion to achieve faster convergence. For the high-flux related diagenesis schemes, the matrix method is more efficient and converges faster than the iterative one.

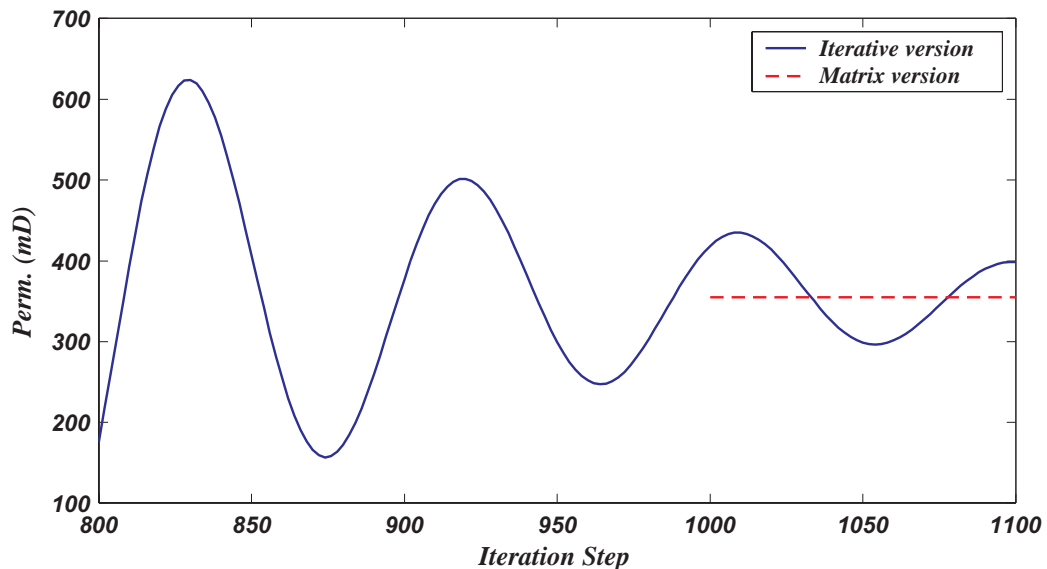


Figure 5.7: Comparison between iterative version and matrix version of Lattice-Boltzmann method. Matrix version converges to the right value. The calculation time of matrix version is about same as 1000 iterations of iterative version.

After finishing this study, we found that Verberg and Ladd (1999) introduced a similar matrix formulation of the Lattice-Boltzmann algorithm, but utilized a direct solution of time-independent equations, rather than usual temporal evolution to steady state. The key idea was to solve for the steady-state mass and momentum density directly, rather than allow them to diffuse in time. They reported the computational efficiency is 1-2 order of magnitude greater than the conventional Lattice-Boltzmann method. A drawback of this method is that it requires approximately 50% more memory than the conventional one.

5.5 Conclusions

The flow-dependent diagenesis simulation provides a simple but fundamental model to explore the relation between transport properties and diagenesis. The empirical relations (the Kozeny-Carman relation and Archie's law) show good correlation with boundary-related mechanisms (Group I), and this implies that grain boundary may be a very important factor in diagenesis of most sedimentary rocks. Low-flux related diagenetic mechanisms (Group II) gives almost no change in permeability until considerable decrease of porosity, while the electrical conductivity is more sensitive to the diagenetic processes than permeability. In high-flux related diagenetic mechanisms (Group III), permeability decreases very rapidly and becomes negligible with a few percent reduction of porosity. The permeability decreases much faster than the electrical conductivity as porosity decreases. The random pore filling (Group IV) shows a linear decrease in electrical conductivity. The permeability decreases quickly at early stages of diagenesis, and then has a similar trend to the boundary related diagenesis mechanisms. A matrix version of the LBM was implemented and applied to the pore geometries with high-flux related diagenetic mechanisms and showed better convergence. The flow-dependent diagenetic modeling in this study, combined with the new LBM implementation using a sparse matrix inversion, is a basic framework for delineating trends of transport properties with a given diagenetic process. The immediate implication of this method will be incorporating more realistic diagenetic mechanisms, such as chemical deposition and dissolution.

Chapter 6

Two-phase Lattice-Boltzmann Flow Simulation

6.1 Abstract

This chapter presents implementation and simulation results of the two-phase Lattice-Boltzmann method (LBM). In the previous chapters, we have seen that the single-phase LBM estimates absolute permeability of porous media very accurately and efficiently. The method is extended to two-phase flow simulation by introducing surface tension and wettability. We will show some characteristics of the two-phase LBM method and results of numerical simulations. The implementation was verified by various numerical experiments. Our implementation honors physical two-phase flow phenomena, such as Laplace's law and contact angles. The successful replication of a drainage-type snap-off indicates that our method properly describes capillary pressure. We applied our method to realistic pore geometry, a microtomographic Fontainebleau sandstone. Both steady state and unsteady state simulations were performed, and we could successfully obtain relative permeability and effective flux as a function of time and location. We observed the nonlinearity between pressure gradient and effective flux through the steady-state simulation. This nonlinearity is easily predictable, since two-phase fluid flow is mainly

controlled by capillary pressure, which is not linearly related to the pressure gradient. The unsteady-state simulations showed very realistic replacement of one phase by the other, for drainage as well as imbibition. They, however, share limitations of laboratory experiments, such as the outlet end effect.

After verifying that the two-phase LBM can simulate two-phase fluid flow through porous media, we investigated the effect of initial distribution of two fluids. In the steady state simulations, the non-wetting phase relative permeability is more sensitive to initial distributions of two fluids than is the wetting phase relative permeability, which is almost unaffected by initial distributions. With the unsteady-state simulations, we investigated the change in sweep efficiency due to different initial distributions. When the non-wetting phase is discontinuous and isolated in the pore space (patchy saturation), the imbibition efficiency is poor and the residual oil saturation is high.

The results show that two-phase Lattice-Boltzmann implementation can simulate realistic two-phase fluid flow in complex pore geometries, which could complement or replace difficult and time-consuming laboratory measurements. In addition, it can help us understand the pore-scale physics and determine appropriate macroscopic relations in two-fluid systems. Lastly, the two-phase simulator can be a module for more complex applications, such as capillary pressure curve simulation, relative permeability estimation from thin sections and diagenesis modeling with two-phase fluid flow.

6.2 Introduction

Two-phase flow in porous rock has considerable economic and scientific importance in typical oil reservoirs. However, the description of two-phase flow phenomena is far more complicated than single-phase fluid flow, because it involves more physical forces, surface tension and wettability. Laboratory measurements of relative permeability are very difficult to perform and are also very time-consuming and costly. An alternative way is to obtain relative permeability through numerical flow simulations. We have already seen the advantages and disadvantages of various numerical flow simulation techniques in Chapter 2. The Lattice-Boltzmann flow simulation technique is chosen in this study for the same reasons as in the single-phase flow simulations. Briefly, the Lattice-Boltzmann

method is comparatively simple to implement and very robust for complicated rock geometries. It has been successfully applied for two-phase flow simulations (Gunstensen and Rothman, 1993; Olson and Rothman, 1997). In this work, we will focus on following objectives.

- (1) To implement two-phase Lattice-Boltzmann flow simulators. This is done by extending the single-phase algorithm by incorporating surface tension and wettability to the discrete Lattice-Boltzmann equation.
- (2) To verify the implementation through several idealized numerical experiments, the results of which have been shown by laboratory measurements or are known from theory.
- (3) To perform feasibility and applicability tests for the simulations with realistic and complex pore geometries.
- (4) To investigate the effect of initial distributions of two fluids both in steady state and unsteady state simulations as one of applications using this implementation.

6.3 Algorithm and Implementation

The Lattice-Boltzmann method for single-phase flow describes fluid flow as collisions of mass particles in a lattice, as we have already seen in Chapter 3. In two-phase flow implementations, we follow almost the same procedures as in the single-phase simulation, except that we have two different types of particles representing two immiscible fluids, and we need to calculate surface tension and wettability, as well as pressure gradient. We will briefly see the algorithm and implementation of two-phase Lattice-Boltzmann method. Extensive details can be found in Rothman and Zaleski (1997), and Chopard and Droz (1998).

The two immiscible fluids – non-wetting and wetting fluids, are commonly called red and blue particles respectively, as they were introduced by Rothman and Keller (1988). This concept is roughly equivalent to have color-dyed fluids, which allows us to see the fluid motion but does not affect the fluids themselves. Interactions between these two

colors (species) are the key ingredient of two-phase fluid flow. For instance, an immiscible two-phase flow can be simulated with a local interaction between particles with the same type and a local repulsion between particles of different types. For this reason, a *color gradient* is computed using the color densities of nearest neighboring nodes. The direction of this gradient determines the local interface between two fluids. Then the post collision density distribution of each fluid is chosen so that a particle of a given color preferentially moves in the direction where the same color species reside. Mass and momentum of each type of fluid should be conserved during these procedures.

Let R_i and B_i be the non-wetting (red) and wetting (blue) phase density distributions. Then the total density distribution, n_i is expressed as follows:

$$n_i = R_i + B_i . \quad (6.1)$$

The local color gradient (\mathbf{f}) is the direction-weighted sum of difference between red and blue density distributions of neighboring nodes, and can be expressed as follows:

$$\mathbf{f} = \sum_i \mathbf{c}_i \sum_j R_j(\mathbf{r} + \mathbf{c}_i, t) - B_j(\mathbf{r} + \mathbf{c}_j, t) . \quad (6.2)$$

The perturbation of density distribution due to surface tension can be express as (Rothman and Zaleski, 1997),

$$C_i(\mathbf{r}, t) = A |\mathbf{f}| \left[\frac{(\mathbf{c}_i \cdot \mathbf{f})^2}{\mathbf{f} \cdot \mathbf{f}} - \frac{1}{2} \right] . \quad (6.3)$$

Equation 6.3 will be more apparent when we consider a 2D hexagonal case. Then the perturbation becomes,

$$C_i(\mathbf{r}, t) = \frac{A}{2} |\mathbf{f}(\mathbf{r}, t)| \cos 2\theta , \quad (6.4)$$

where θ is the angle between \mathbf{f} and \mathbf{c}_i . Essentially, the perturbation is chosen to remove mass from the density distribution moving parallel to the interface and add mass to the density distribution moving in the normal direction while conserving mass and momentum. Equation 6.3 has a scaling parameter A , to adjust the surface tension. This will be covered in detail later in this section. Now, the discretized Boltzmann equation (Equation 3.5) becomes,

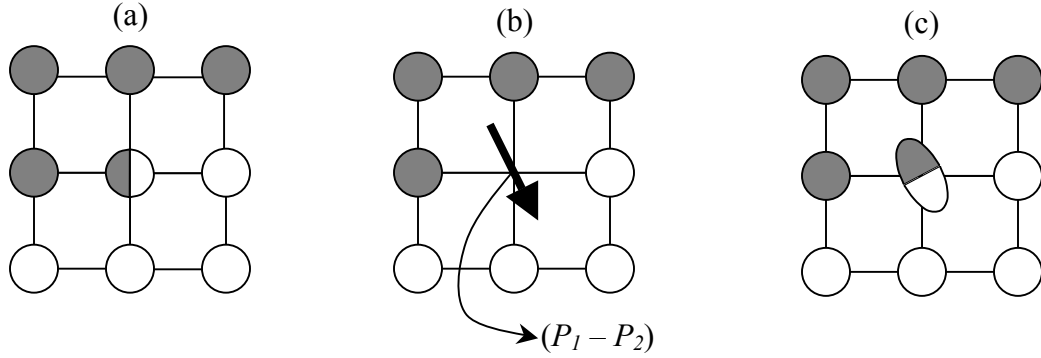


Figure 6.1: Schematic diagram of the surface tension calculation. (a) Initial state. Gray color denotes non-wetting fluid and white denotes wetting fluid. The target node is at the center, which is half non-wetting fluid and half wetting fluid. (b) Calculation of the local gradient from the distribution of neighboring nodes. Arrow denotes the direction of local surface tension. (c) Final state. The two fluids are redistributed by the local surface tension. Note that the total mass distribution of the node is deformed by the surface tension force.

$$n_i(\mathbf{r} + \mathbf{c}_i, t + 1) = n_i(\mathbf{r}, t) + \sum_j n_j^{neq}(\mathbf{r}, t) \Omega_{ij} + C_i(\mathbf{r}, t). \quad (6.5)$$

We have a new density distribution, and the next step is to redistribute the density distribution to minimize the diffusion of one type of fluid into the other. This involves finding R_i' and B_i' that satisfies the following equation:

$$\max \left\{ \sum_i (R_i' - B_i') \mathbf{c}_i \cdot \mathbf{f} \right\},$$

subject to,

$$\sum_i R_i' = \sum_i R_i, \quad \sum_i B_i' = \sum_i B_i \quad \text{and} \quad n_i = R_i' + B_i'. \quad (6.6)$$

The new collision rule for two-phase immiscible fluid flow is shown in Figure 6.1 and is summarized as follows:

- (1) Perform a single-phase collision (Equation 3.4).
- (2) Calculate the local color gradient using the color densities of neighboring nodes (Equation 6.2).

(3) Perturb the density distribution (Equation 6.5).

(4) Separate the two fluids (Equation 6.6)

The whole algorithm of the two-phase LBM can be obtained by simply replacing the collision step in Chapter 3 with this new collision rule.

To find the relation between the parameter A and macroscopic surface tension, we use the mechanical description of surface tension (Rothman and Zaleski, 1994) with a flat interface between wetting and non-wetting phases.

$$\gamma = \int_{-\infty}^{\infty} (P_N - P_T) dn \quad (6.7)$$

where P_N and P_T are the normal and tangential components of the momentum flux tensor, n is the normal to the interface, and γ is the surface tension. Detailed derivation in a FCHC (Face-Centered Hyper Cube) lattice can be found in Gunstensen et al. (1991). The surface tension in the Lattice-Boltzmann method is

$$\gamma = -\frac{8A}{\lambda} n_c \rho, \quad (6.8)$$

where λ is an eigenvalue of the collision operator, n_c is the number of velocity vectors ($n_c=24$ for FCHC) and ρ is density.

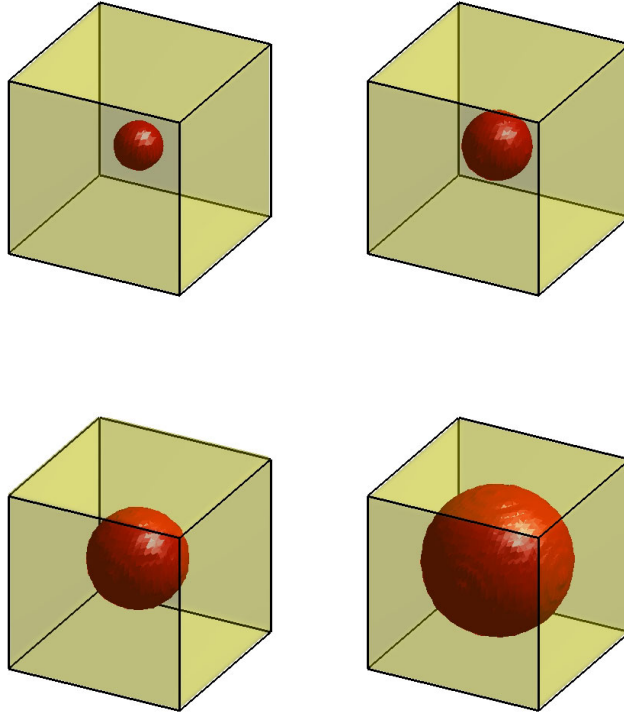
6.4 Verifications

In this section, the implementation of the two-phase Lattice-Boltzmann method is applied to many different idealized situations in which the results are already known from theory or laboratory measurements. This will show that the implementation is valid and applicable for real physical two-phase flow in porous media.

6.4.1 Capillary Pressure

A common experiment for capillary pressure between two immiscible fluids is to use spherical bubbles of non-wetting fluid (Gunstensen and Rothman, 1992). In this case, the capillary pressure is given by Laplace's equation (Dullien, 1992),

(a)



(b)

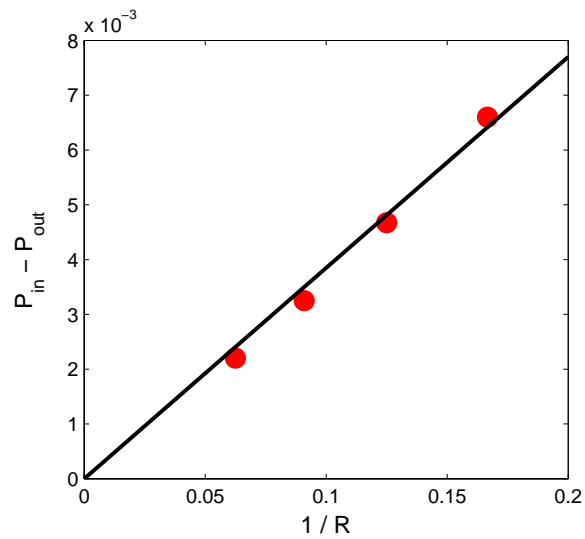


Figure 6.2: (a) Four different sizes of non-wetting bubbles in a sea of wetting-phase fluid. (b) Capillary pressure vs. reciprocal of bubble radius. Simulated values (dots) agree well with the theoretical prediction (solid line).

$$P_{in} - P_{out} = \frac{2\gamma}{R} \quad (6.9)$$

where P_{in} and P_{out} are pressures inside and outside of the bubble and R is the radius of the bubble. Four different sizes of bubbles (Figure 6.2a) are used for the numerical experiments. Figure 6.2(b) shows the capillary pressures for four different bubbles. The theoretical prediction of the capillary pressure is calculated by Equation 6.8 and is shown as a solid line. The simulated values (dots) are obtained by simply calculating pressures inside and outside after the numerical simulations. The numerical flow simulation results show very good agreement with the theoretical prediction. Small discrepancies come from the discretization effect of the bubble. This numerical experiment shows that our implementation honors Laplace's law for capillary pressure.

6.4.2 Wettability and Contact Angle

Two-phase fluid systems in porous media always involve a solid phase; hence three interfacial tensions are involved (Figure 6.3). In equilibrium, the horizontal forces are balanced as follows:

$$\begin{aligned} \gamma_{so} &= \gamma_{sw} + \gamma_{ow} \cos \theta, \\ \text{or, } \cos \theta &= \frac{\gamma_{so} - \gamma_{sw}}{\gamma_{ow}}, \end{aligned} \quad (6.10)$$

where γ_{sw} is interfacial tension between the solid and water, γ_{so} is interfacial tension between the solid and oil, γ_{ow} is interfacial tension between oil and water, and θ is the contact angle.

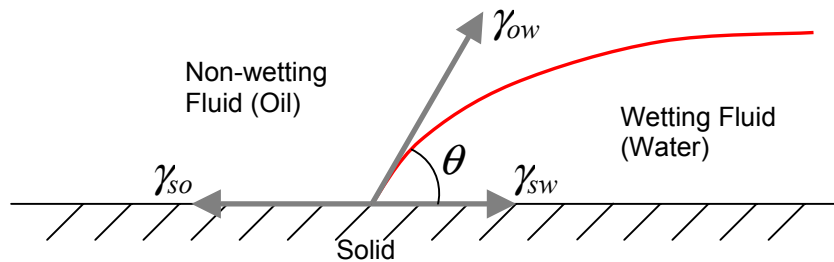


Figure 6.3: Equilibrium at a line of contact (Craig, 1971).

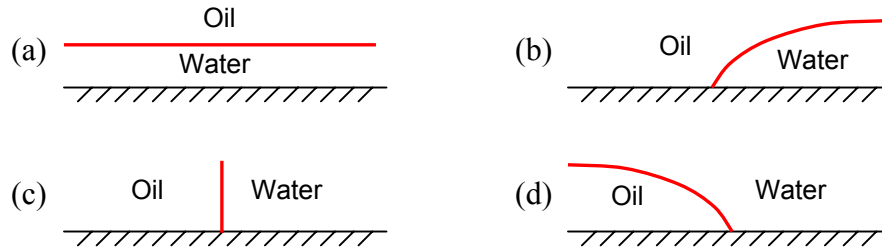


Figure 6.4: Different wettability situations in porous media: (a) completely water-wet, (b) water-wet, (c) neutral-wet, and (d) oil-wet systems.

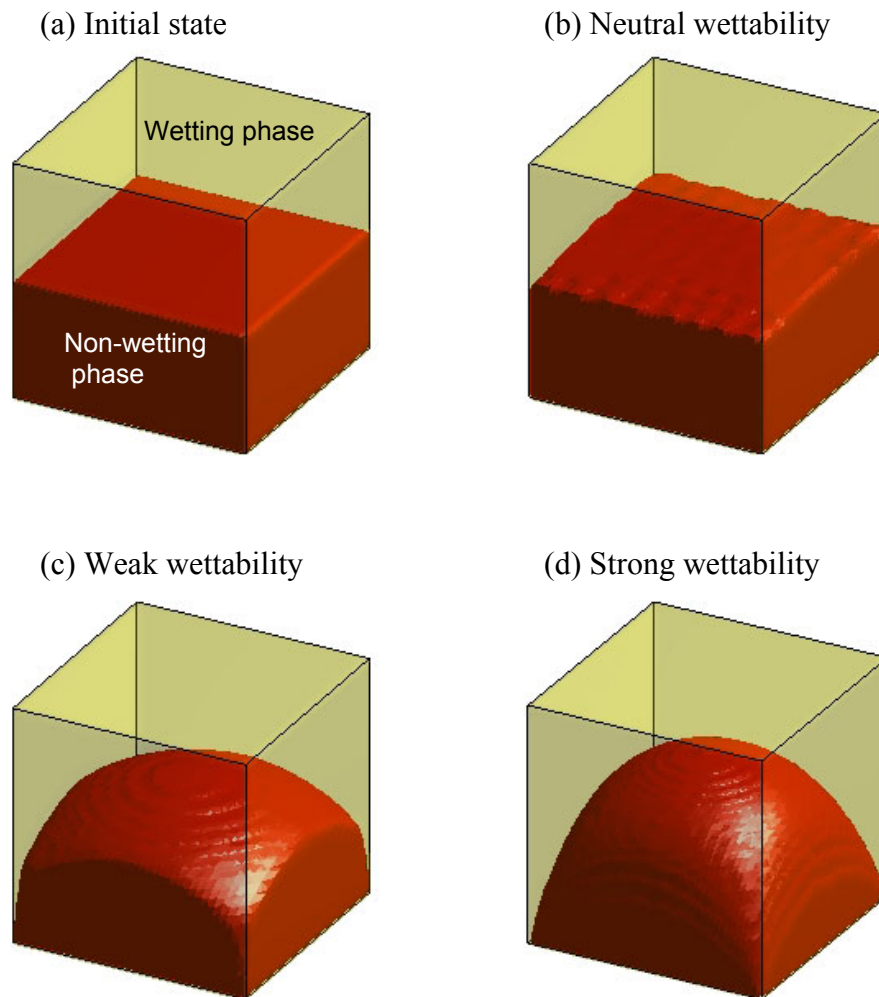


Figure 6.5: Wettability test with a simple box model. (a)Initial model has two layers of immiscible fluids separated horizontally. (b), (c) and (d) show simulated results for neutral wettability, weak wettability and strong wettability, respectively.

Figure 6.4 shows different wettability situations in porous media. Clean sandstones are usually water-wet, however, most of reservoir rocks are not completely water-wet. To confirm that the implementation of two-phase flow simulations by the Lattice-Boltzmann method can represent a wide range of wettability situations, a simple model of a box containing two immiscible fluids is used (Figure 6.5). The wetting phase is located in the upper half and the non-wetting phase is in the lower half of the box. As the wettability increases, the meniscus is deformed and successfully shows different wettability situations. In the neutral wettability case, it shows the right contact angle of 90° .

Figure 6.6 shows a very similar situation to Figure 6.5. In this case, the non-wetting phase is initially located at the bottom-center of the box. With neutral wettability, the contact angle shows 90° as theory predicts. As the wettability increases, the contact angle decreases accordingly, and eventually the non-wetting phase is separated from the bottom and forms a spherical bubble. Therefore the implementation not only replicates the right contact angles but also handles a wide range of wettabilities.

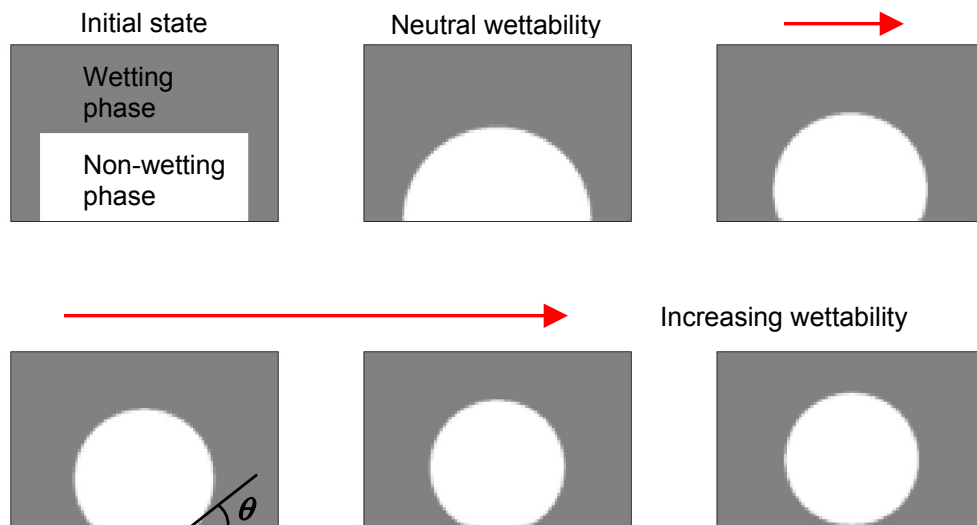


Figure 6.6: Contact angles by the two-phase Lattice-Boltzmann simulation. Each plot is shown as a cross-sectional view. Initially the non-wetting fluid is located at the bottom center of the box. As wettability increases, the contact angle decreases accordingly. Eventually the non-wetting phase is detached from the bottom and forms a spherical bubble.

6.4.3 Two-phase Flow through Pipes

In the previous sections, we have seen that the two-phase LBM can represent interfacial tensions properly. Now we will include an external driving force, a pressure gradient, which causes the two fluids to flow through porous media. Figure 6.7 shows simple pipe models with three different diameters. We used the same values of pressure gradient and wettability for all models. The left column shows imbibition in which the wetting fluid replaces the non-wetting fluid and the right column shows drainage where the non-wetting fluid is pumped into the wetting-fluid-saturated system. As the diameter of the pipe increases, we can observe that the contact angles decrease accordingly because the capillary pressure decreases (Equation 6.9). It is also observed that the contact angle at imbibition (θ_1) is greater than that at drainage (θ_2). This is the well-known contact angle hysteresis (Dullien, 1992). We can also observe the capillary pressure in this simple model. Figure 6.8 shows the pressure profiles along the samples. The profile lines, $A-A'$ and $B-B'$ are shown in Figure 6.7. The pressure in the non-wetting phase is always higher than that in the wetting phase because of the capillary pressure near the interface. The simulation successfully shows the capillary hysteresis.

6.4.4 Simple Mixed-wet System

The algorithm of the Lattice-Boltzmann method also enables us to have a very complex mixed-wet system. This would be very useful for simulating two-phase fluid flow through mixed-wet porous media. The model (Figure 6.9) is a simple pipe with different wettabilities of the solid phase. The first half of the solid is water-wet, while the other is oil-wet. The pore is initially saturated with oil and water is pumped from the left-hand side. Until the meniscus between oil and water meets the boundary between water-wet and oil-wet solids, it is convex to the water phase. However, as soon as the meniscus reaches the boundary, it becomes convex to the oil phase. This is a very simple mixed-wet system. However, the two-phase LBM can handle more complicated mixed-wet situations. Actually we can assign a different wettability to each node.

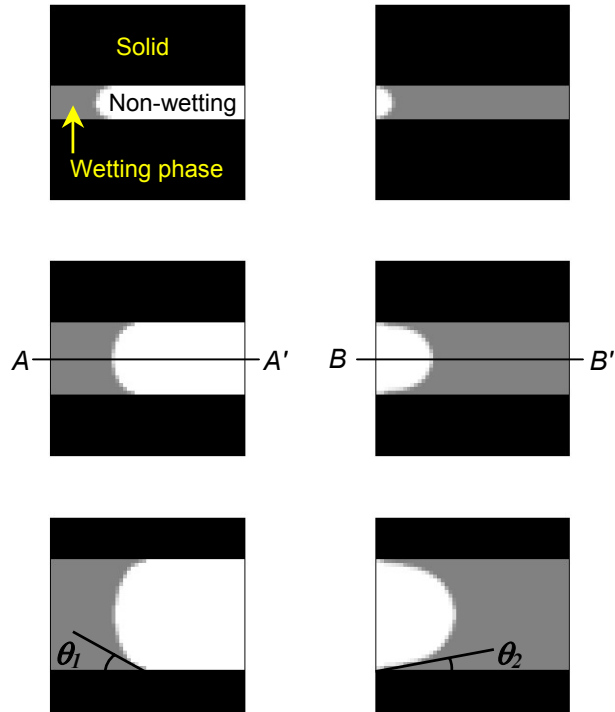


Figure 6.7: Imbibition (left column) and drainage (right column) through pipes. Fluid is pumping from the left-hand side of the pipe. Each row has a pipe of a different radius. Note that the contact angle at imbibition is different from that at drainage. The two lines ($A-A'$ and $B-B'$) in the middle row are for the pressure profiles in Figure 6.8.

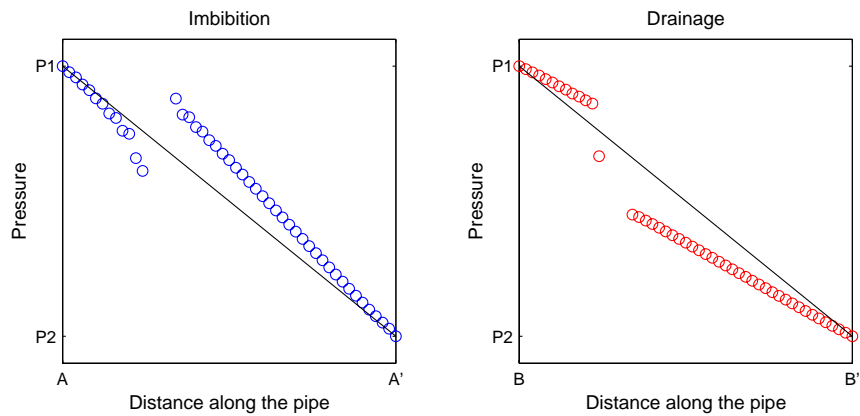


Figure 6.8: Pressure profiles along the pipe model for imbibition and drainage. Dots are pressure values from the simulation. The $A-A'$ and $B-B'$ lines are from Figure 6.7. Black solid line denotes a constant pressure drop.

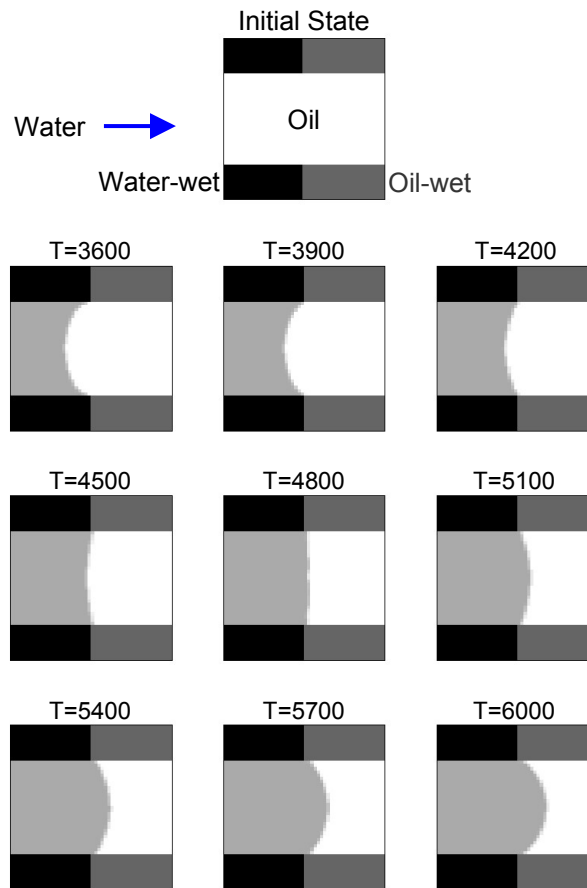


Figure 6.9: Two-phase flow in a mixed-wet system. Plots are shown in cross-sectional views of a pipe model. The solid part has two regions of different wettability. The left half is water-wet and the right half is oil-wet. Initially the pore is filled with oil, and water is pumped from the left-hand side. Time in the title of each plot is in the simulation unit. The meniscus between oil and water changes when it meets the boundary between the water-wet and oil-wet solid.

6.4.5 Drainage-type Snap-off

Another well-studied model of immiscible displacement, the so-called pore doublet model, is a little more complicated. A typical pore doublet consists of two tubes with different diameters, joined at both ends (Figure 6.10). Since the capillary pressure is inversely proportional to the radius of the tube, the capillary pressure of the smaller tube is greater than that of the bigger tube. Drainage-type snap-off occurs when the external

pressure gradient is big enough to overwhelm the capillary pressure of the bigger tube, but is not big enough for the smaller tube. Theory and laboratory experiments show that under this condition the wetting phase in the smaller tube is trapped while that in the bigger tube is replaced by the non-wetting phase (Lenormand et al., 1983; Chatzis and Dullien, 1983). Figure 6.10 shows that the two-phase Lattice-Boltzmann method successfully replicates the drainage-type snap-off, which tells us that the method describes capillary pressure phenomena of porous media accurately.

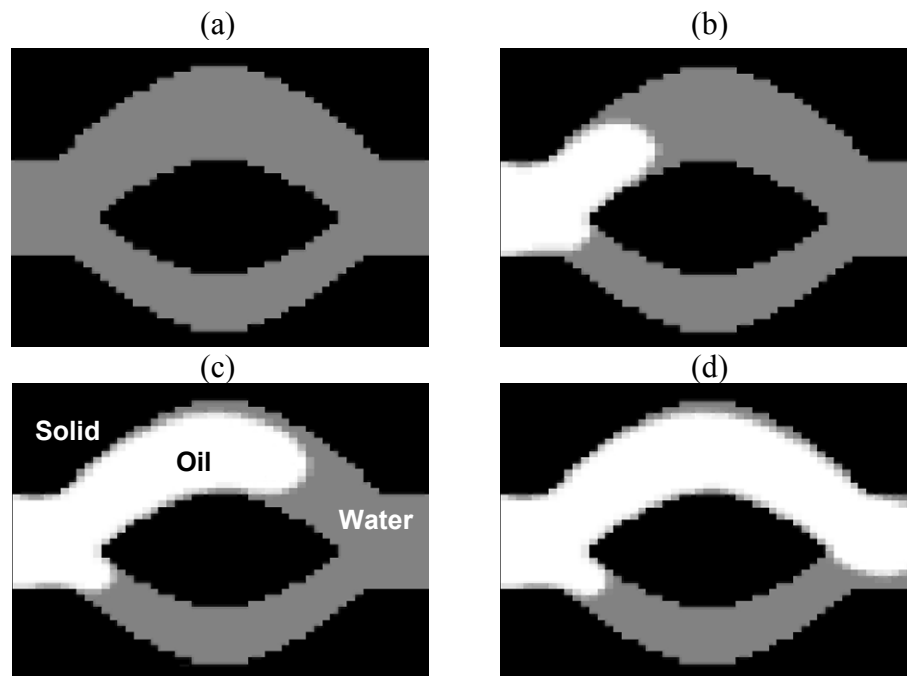


Figure 6.10: Drainage-type snap-off in a doublet. (a) Initial stage. Pore space is completely saturated with water (wetting phase). (b) After 5000 iterations. Oil is replacing water in the system. (c) After 10000 iterations. (d) Final stage. There is a continuous oil phase through the bigger pore, while the water phase is trapped in the smaller pore.

6.5 Two-phase Flow Simulation in Porous Media

This section presents two-phase flow simulations in realistic and complex porous media. The microtomographic Fontainebleau sandstone sample was used in the study. Details of the sample have been covered in Chapters 3 and 4.

6.5.1 Relative Permeability and Two-phase Flow Simulation

There is no widely accepted macroscopic relation in two-fluid systems. However, if two fluids were macroscopically separated in porous media, each phase would satisfy Darcy's equation:

$$\begin{aligned} q_w &= -\left(\frac{\kappa_{rw}\kappa}{\mu_w}\right) \frac{dP_w}{dx} \\ q_n &= -\left(\frac{\kappa_{rn}\kappa}{\mu_n}\right) \frac{dP_n}{dx} = -\left(\frac{\kappa_{rn}\kappa}{\mu_n}\right) \frac{d(P_n - P_c)}{dx} \end{aligned} \quad (6.11)$$

where the subscripts w and n denote wetting and non-wetting fluids and P_c is the capillary pressure. The new parameters, κ_{rw} and κ_{rn} are called relative permeabilities for the wetting and non-wetting phases, respectively. Although Equation 6.11 is used for most practical modeling, there are many issues regarding its applicability. First of all, the equations implicitly assume that the two fluids flow independently each other through the porous media. Coupling between two fluids or any flow-driven changes in the interface geometry are limited in this approach. The coupling between two fluids has been studied and the results are quite controversial – from negligible to significant (Olson, 1995). A modification of Darcy's law with coupling terms has been suggested (Gunstensen and Rothman, 1993). Secondly and more fundamentally, it is unclear why the two-fluid system with non-linear capillary force would obey a linear relation between the flux and pressure gradient (Adler and Brenner, 1988). Although there are many issues with Equation 6.11, we will use the equation to address calculated relative permeability from the two-phase flow simulation. However, in most parts of this chapter, we will use a normalized flux of each phase, since the normalized flux is straightforward to measure and it is approximately the same as relative permeability. The definition of effective flux can be found in the next section (Equation 6.12).

There are a couple of assumptions for non-wetting and wetting fluids in this study. The viscosities and densities of two fluids are assumed to be the same. Therefore, the non-wetting and wetting phases are not oil and water, since we used the same viscosity and density for both fluids. However, they are called oil and water for more clear discussion. The assumption for using the same viscosity for two fluids is not due to

limitations of the Lattice-Boltzmann algorithm. It is done to simplify the model for faster calculations, since two very distinct viscosities of fluids will require almost an order of magnitude larger calculation time. There are a few recent efforts incorporating different viscosities of fluids in the Lattice-Boltzmann method (Langaas and Yeomans, 2000; Orlandini et al., 1995).

There are typically two kinds of laboratory measurement techniques for two-phase fluid flow and relative permeability estimation – steady state and unsteady state methods. Both techniques were implemented and simulated, although the numerical implementations are a little different from the laboratory measurement techniques. Details on each simulation technique are covered in the following sections.

6.5.2 Steady State Simulation

Figure 6.11 shows a schematic diagram of the steady-state simulation technique. In the steady-state laboratory measurement (Tiab and Donaldson, 1996), two fluids are premixed with a pre-defined ratio and are pumped into the core sample. The simulation methods in this study are very similar to the laboratory experiment, although there are a few differences. In the numerical simulation, we simply assign pre-determined volumes of two fluids in the pore nodes and let them flow under a given pressure gradient with periodic boundary condition. There can be an issue regarding the initial distribution of two fluids, since different initial distributions may lead to different final status. This issue will be covered in the later sections.

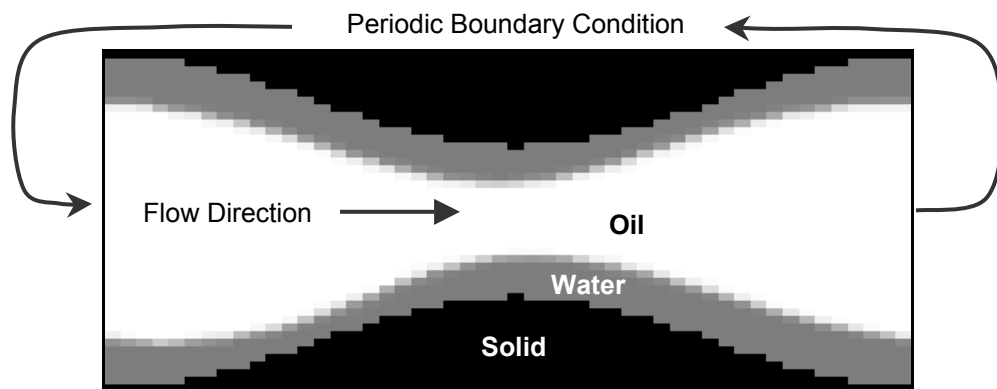


Figure 6.11: Schematic diagram of the steady-state simulation of two-phase fluid flow.

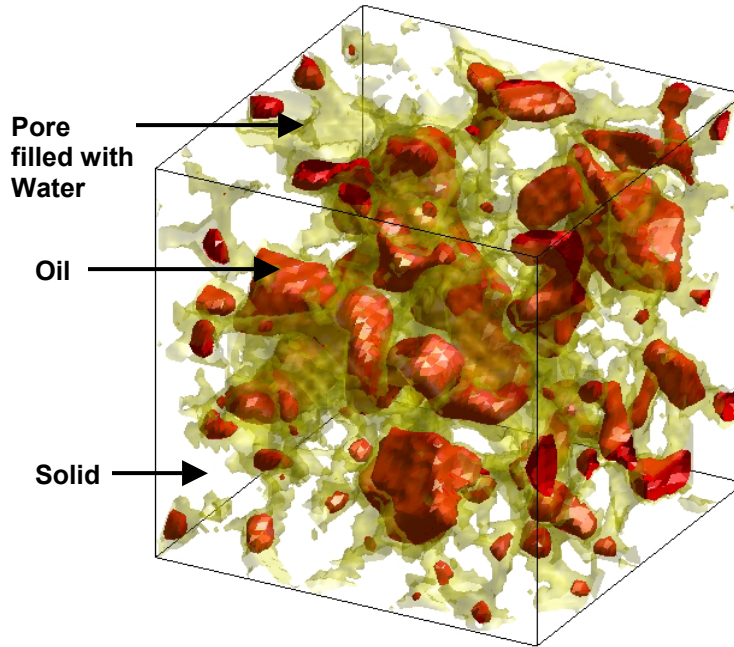


Figure 6.12: Distribution of two fluids after the steady-state simulation. White parts denote solid grains and greenish yellow denotes pore structure filled with water. The oil phase is shown in red.

Figure 6.12 shows the final distribution of two fluids in the Fontainebleau sandstone sample after a two-phase flow simulation. The solid grains are shown as white. Green geometry is the pore structure filled with water, and red denotes the oil phase. We can see how complexly the two fluids are distributed. The calculation time for a typical pore geometry of $128 \times 128 \times 128$ nodes is from a few hours to a day on a PC workstation with a 1.7GHz Pentium4 processor. The relative permeability curves are obtained from the steady state flow simulation as functions of water (wetting phase) saturation (S_w). Figure 6.13 shows series of normalized flux curves, which are given by

$$q_{nw} = \frac{q_w}{q_{sw}}, \quad q_{nn} = \frac{q_n}{q_{sn}} \quad (6.12)$$

where q_{sw} and q_{sn} are fluxes from single phase flow simulations with wetting and non-wetting fluids, respectively. The normalized flux is the horizontal flux along the pressure gradient direction, divided by the flux from the single-phase flow simulation under the

same pressure gradient. This is approximately the same as the relative permeability, if the pressure gradient of the wetting phase (Equation 6.11) is not much different from that of the non-wetting phase.

As seen in Figure 6.13, we observe that relative permeability curves change with pressure gradient. This implies that the flux (flow rate) of each fluid is not linear with the pressure gradient, which shows that the extended Darcy's law for two-phase flow (Equation 6.10) is not completely valid. This nonlinearity is quite obvious because the controlling force for two-phase fluid flow, capillary pressure, is not linear. Another observation is that the oil phase has bigger changes in relative permeability, while the water phase does not show a significant difference in relative permeability due to the change in pressure gradient. Furthermore, the oil relative permeability increases as the pressure gradient increases, while the water relative permeability seems to decrease, though the amount is very small. This nonlinearity between the flux and pressure gradient is greater at intermediate to high water saturation.

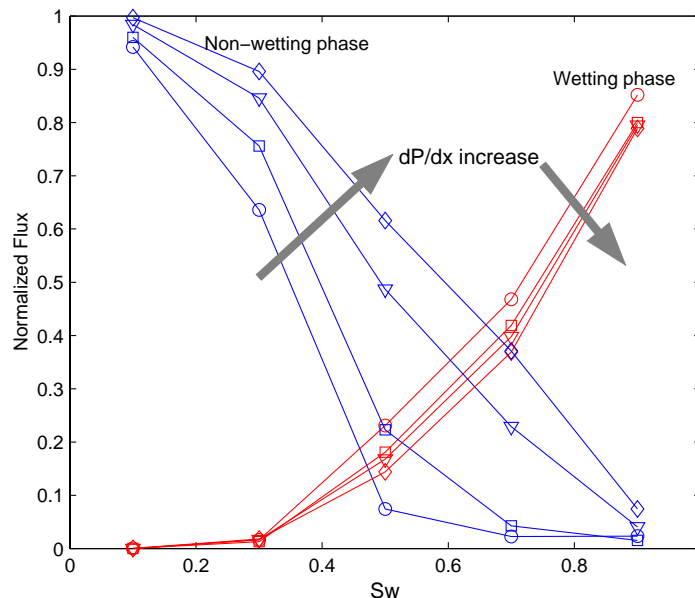


Figure 6.13: Normalized flux for non-wetting and wetting phases. Different symbols denote normalized flux values under different pressure gradients. The non-wetting phase shows higher sensitivity to the change in pressure gradient than the wetting phase.

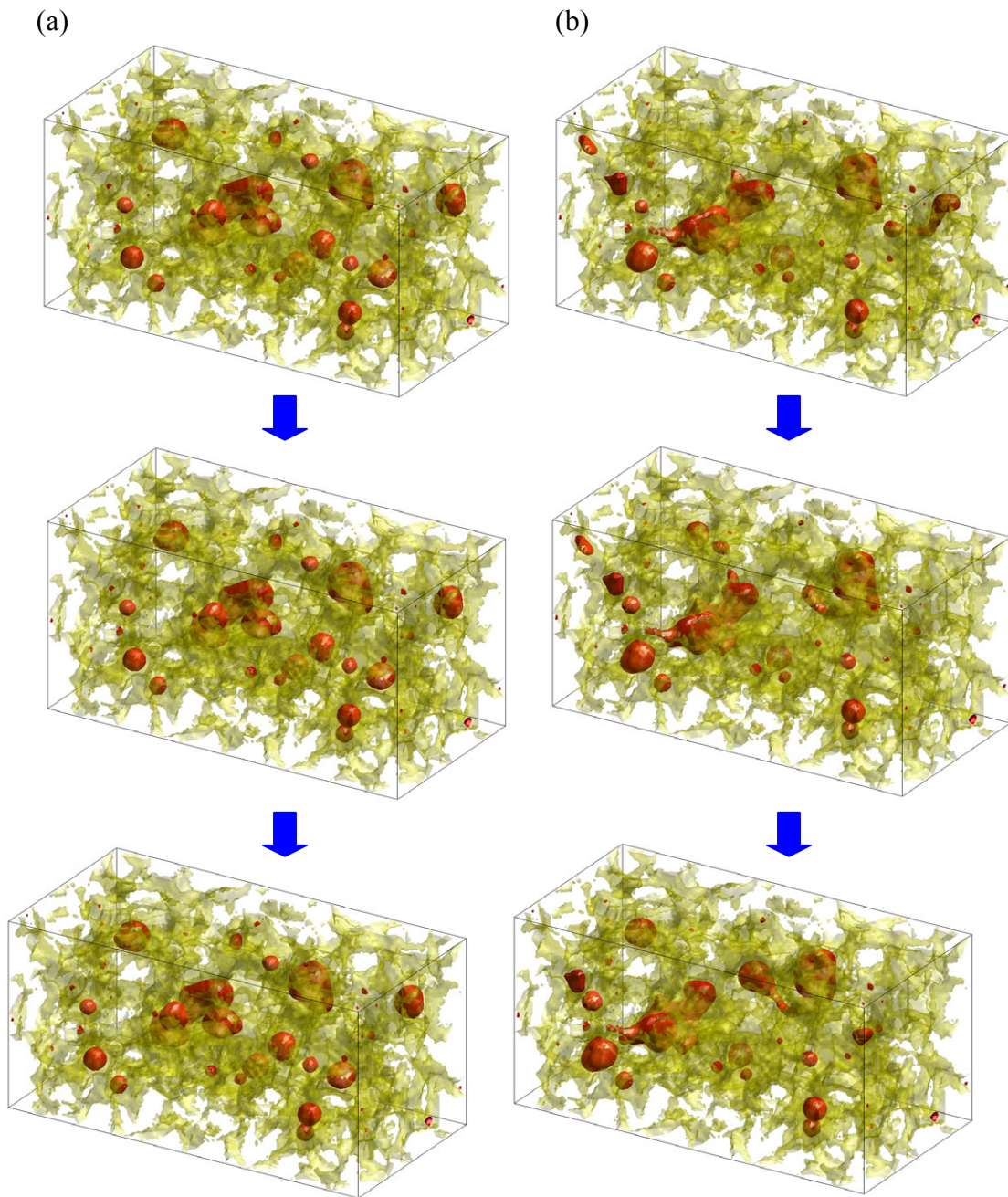


Figure 6.14: Two-phase flow in a Fontainebleau sandstone (a) under low pressure gradient, and (b) under high pressure gradient. Fluid flows from right to left. Color notations are the same as in Figure 6.12. Each gray arrow represents a time lapse of 5000 iterations.

Figure 6.14 shows the nonlinearity more clearly. Two series of snapshots from a steady-state flow simulation are shown. The water saturation is about 90%; the oil phase loses connectivity and is trapped in the big pore space. Column (a) is under a low pressure gradient and Column (b) is under a pressure gradient high enough to drive the oil phase against the capillary pressure. Three snapshots at each column were taken 5000 iterations apart. Column (a) shows that nothing happens during the simulation. The oil phase is trapped and stays in the big pores. However, with the high pressure gradient, the oil blobs can go through pore throats and flow with water. Due to the capillary pressure, which is a function of pore geometry and types of fluids, the relative permeability is not linear to pressure gradient. The non-linearity effect is bigger at high water saturation, when the oil phase loses continuous connectivity.

6.5.3 Unsteady State Simulation

The unsteady state technique in the laboratory (Tiab and Donaldson, 1996) mimics oil migration and production in oil reservoirs. Initially the rock sample is fully saturated with water. Oil is pumped into the sample and water is drained until no more water is produced (drainage). The saturation at this point is often called irreducible water saturation. Then the sample is jacketed and allowed to adjust to capillary equilibrium. After achieving equilibrium, water is pumped into the sample (imbibition), and the volumes of the two fluids at the inlet and outlet are measured. The unsteady-state numerical simulation of this study is almost the same as the laboratory measurement technique. The simulation also consists of two stages, drainage and imbibition. Figure 6.15 shows a schematic diagram for the unsteady-state simulation.

Again, we use the same rock sample, and the rock is assumed to be completely water-wet. Figure 6.16 shows snapshots during the simulation. A small portion of the rock sample (lower right portion of Figure 6.12) is chosen for a more clear view of two-fluid flow. The color notations are the same as in previous figures. The drainage simulation is shown from (a) to (d). Initially the sample is completely water-saturated, and then the oil phase replaces water in big and well-connected pores first, and forms finger-type shapes. Finally no more water is produced and the drainage simulation is finished.

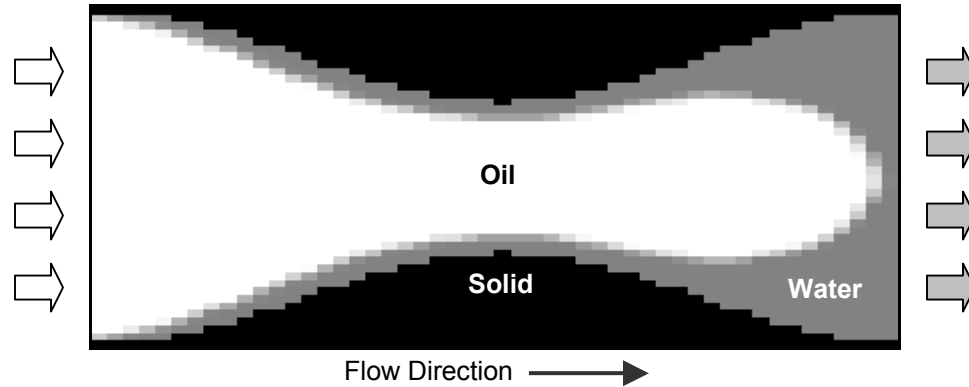


Figure 6.15: Schematic diagram of the unsteady state simulation of two-phase fluid flow.

Figure 6.16(e) shows the irreducible water saturation, which is about 35%. The imbibition simulation is presented from (f) to (g). Water, which is the wetting phase in this system, replaces oil in a different way. It pushes the oil phase in a so-called piston-like way. Since the capillary pressure helps water replace oil, imbibition is a more efficient mechanism than drainage. Figure 6.16(h) shows the end of the imbibition simulation. In the upper part of the figure, we can see the residual oil saturation, which is about 10%. These snapshots show realistic two-phase flow through complex pore geometries.

The relative permeability (\approx normalized flux) is shown in Figure 6.17. The oil flux in the drainage case shows an interesting result; it stays almost the same for water saturations of 70% to 50%. Figure 6.18 shows the reason for this phenomenon. Within this range of saturations, a significant portion of the oil phase moves perpendicular to the pressure gradient direction. Therefore the effective flow along the pressure-gradient direction does not increase and stays almost the same. The normalized flux curves from the imbibition in Figure 6.17(b) show the very linear trends for both oil and water phases. Commonly, the relative permeability curves are much smaller than these linear trends (see Figure 6.13).

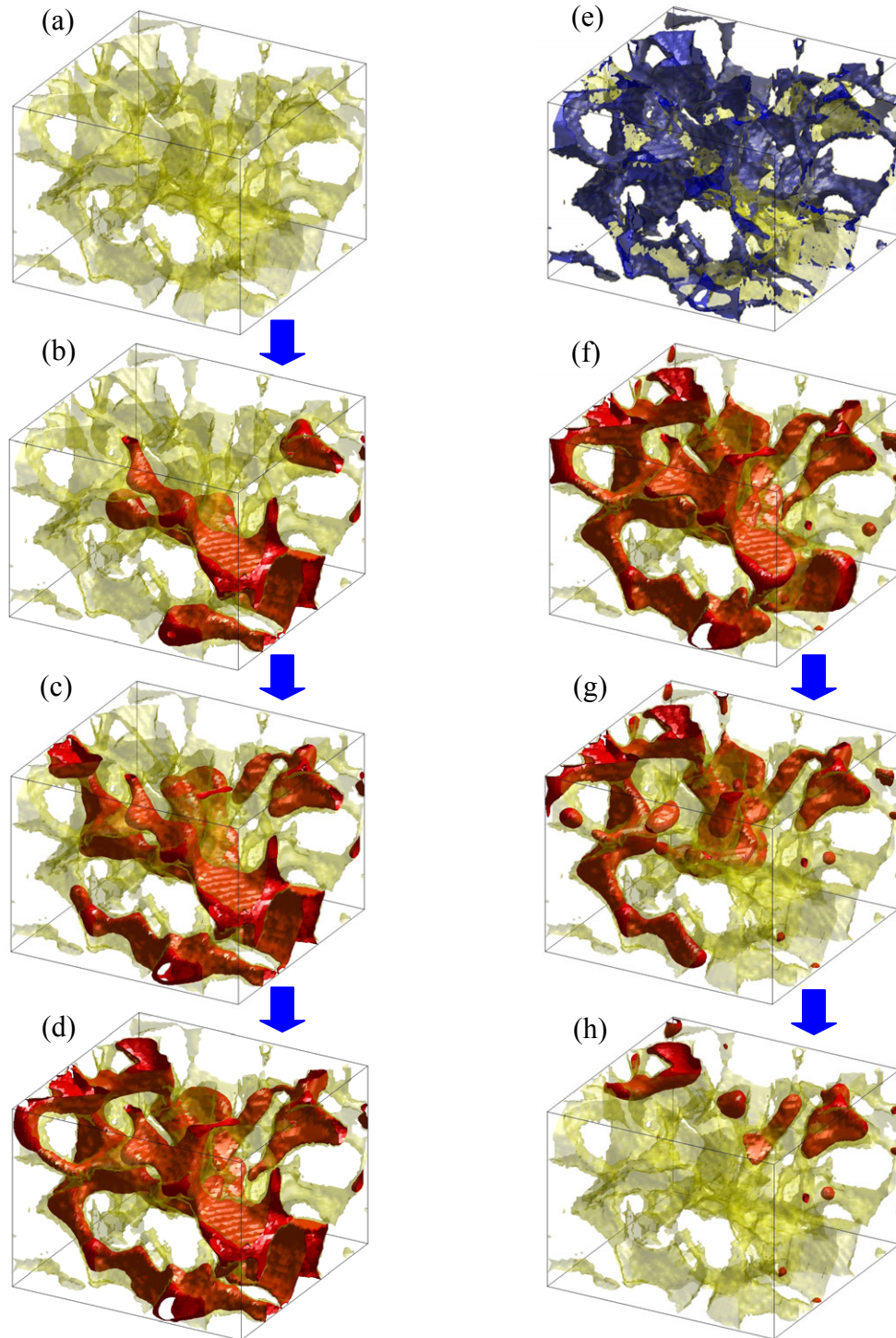


Figure 6.16: Snapshots during an unsteady-state simulation; drainage (a-d), irreducible water (e) and imbibition (f-h). Fluid flows from right to left. The blue in (e) denotes water. The other color notations are the same as in Figure 6.12.

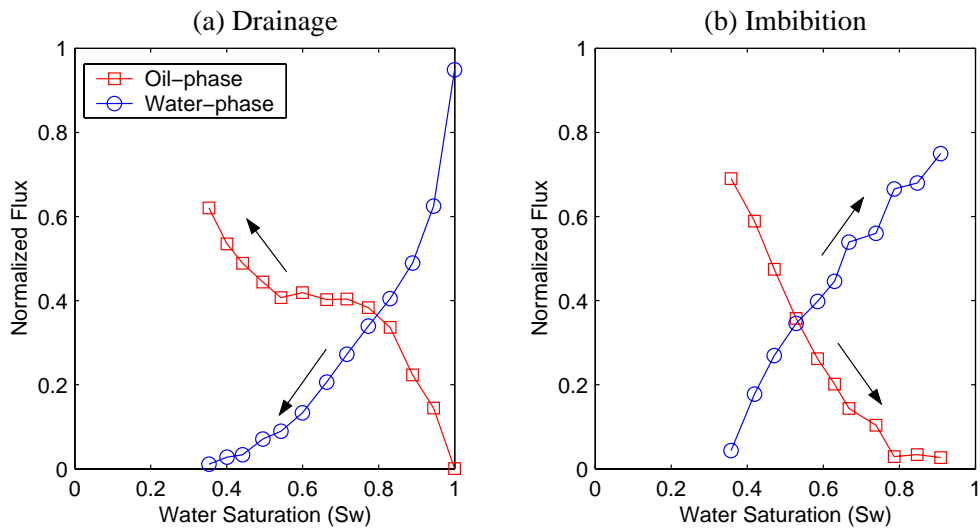


Figure 6.17: Normalized flux curves from the unsteady-state simulation.

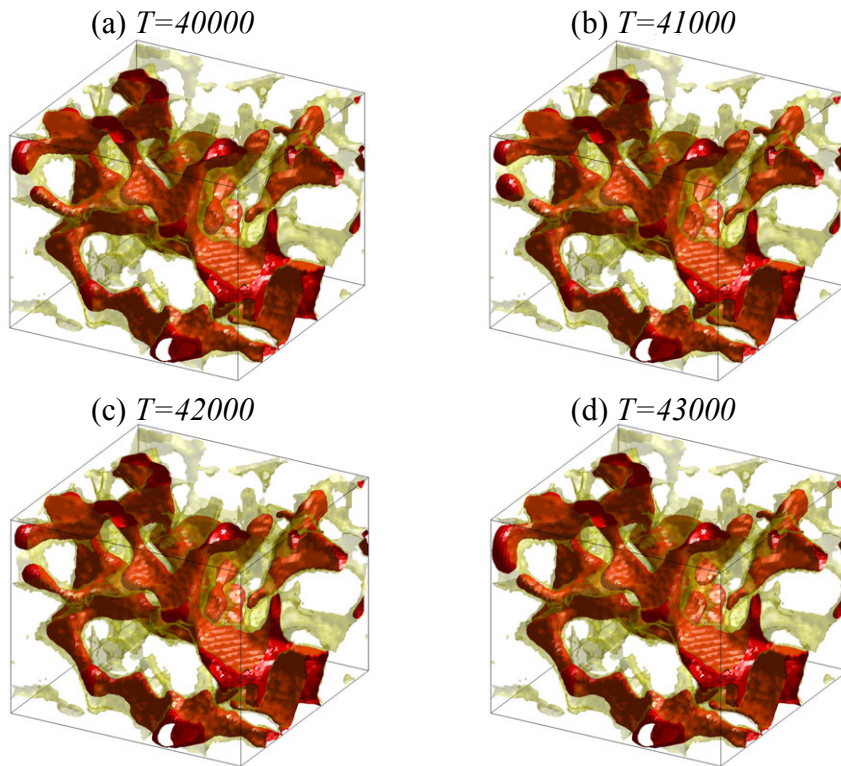


Figure 6.18: Detailed snapshots during drainage. Note that the movement of the oil phase is perpendicular to the direction of the pressure gradient.

According to Craig (1971), the unsteady-state method has two serious limitations: (1) The pressure gradient or is flow rate has to sufficiently high to eliminate outlet end effects; and (2) viscous oils are generally used to obtain relative permeabilities over a wide saturation range. Otherwise, the imbibition displacement would be piston-like with high flow rates. The linear trend of normalized flux curves comes partially from these limitations. However, since we can control the full range of parameters, such as interfacial tensions and pressure gradient, we did change these sets of parameters to see the differences. Though there were small differences in normalized flux curves upon changing the parameters within reasonable ranges; the changes were not significant. Another cause of the linear trend is that the rock sample is not big enough to capture all possible two-phase distributions in pore geometry. In other words, the size of the digital rock sample is not big enough to represent the two-phase fluid flow of the whole rock. This issue will be covered in detail in Chapter 7.

6.5.4 Representative Elementary Volume and Grid Spacing

In Chapters 3 and 4, we discussed the representative elementary volume (REV) for porosity and absolute permeability. To deal with the length scale consistently, the autocorrelation length or two-point correlation length (a) has been used, which is an approximate measure of mean pore size. When a rock is isotropic and homogenous, a small portion of a rock, greater than $5a$ in length, accurately represents the whole rock well in terms of porosity. Absolute permeability, on the other hand, requires a bigger REV to represent fluid flow through the rock. Although the REV strongly depends on the homogeneity and isotropy of the rock, we found that a sample size with a length of $10a$ or greater represents the permeability of the whole rock quite well.

However, two-phase flow through porous media is much more complicated, since we have two immiscible fluids and they are controlled by the nonlinear force, capillary pressure. The force balance is a function of not just pore geometry; it depends also on local fluid saturation. The continuity of a fluid, especially for the non-wetting phase, also has a significant effect on the flow rate. Thus, two fluids can have very different distributions depending on the details of pore geometry. In other words, the two-phase fluid flow is affected strongly by the arrangement of pore size distribution and

connectivity. Even if a digital rock sample is big enough to represent porosity, the autocorrelation function and single-phase fluid properties, the sample does not necessarily represent the two-phase fluid properly. It is almost impossible to quantify how big the sample should be. However, it is obvious that we need a bigger sample for the two-phase flow simulation to capture all possible two-fluid situations in the pore geometry, such as trapped oil, irreducible water saturation, and continuity of old phase.

On the other hand, the two-phase flow simulation requires finer grid structures (higher resolution of pore geometry). There are two interfaces in two-phase flow simulations – the interface between two immiscible fluids and pore/grain boundary – while the single-phase flow simulation requires only one interface – the pore-grain boundary. To describe the two interfaces properly, we need to use a finer grid structure. We learned that a grid spacing of $1/5$ the mean pore length ($dx = a/5$) is not fine enough to describe two-phase flow. The grid spacing should be $dx = a/10$ or less. If the pore size distribution is very broad, smaller grid spacing should be used, since small pores or pore throats play a significant role in two-phase flow. Thus we need to run much bigger problems in two-phase flow simulations than in single-phase flow simulations. If we double the size of a digital rock and reduce the grid spacing by half, the total size increases by four times in 1D and by 64 times in 3D. Even with rapidly increasing modern computing power, this size is almost prohibitive. One approach to this dilemma would be parallel implementations, which will be described in Chapter 7. The grid spacing and size of digital rock samples used in this chapter is very close to the minimum to satisfy these conditions. More details will be covered in Chapter 7.

6.5.4 Summary

This section has shown that the two-phase LBM represents realistic two-fluid flow in complex pore geometry reasonably well. Both the steady state and unsteady state simulations can replicate laboratory measurements. Moreover the LBM can handle two-phase flow more easily, since we can obtain all parameters locally in a rock, such as pressure, saturation and flux as a function of location and time. Through the numerical experiments, we found that the implementation of the two-phase Lattice-Boltzmann method can be a good candidate for calculating relative permeability and also for

exploring the pore-scale physics of two-phase fluid flow.

6.6 Effect of Initial Distribution of Two Fluids

This section will show the potential of two-phase Lattice-Boltzmann implementation for exploring the pore-scale physics of two-phase fluid flow in real rock

The spatial distribution of pore fluids can have a significant impact on physical properties of rocks. For example, elastic properties of a rock with uniform saturation of fluids may be very different from those with patchy saturation (Sengupta, 2000). Similarly, the original distribution of two fluids in a pore structure will have a significant effect on how the two fluids flow and eventually what the relative permeabilities will be. We will investigate the change in relative permeability (normalized flux) due to the different initial geometry of two fluids.

In the steady-state simulation, we assigned wetting and non-wetting fluids in the pore geometry according to the initial saturation. To investigate changes in relative permeability due to different initial distributions of two fluids, three different kinds of initial distributions are used; (1) boundary coating with wetting fluid, (2) random distribution, and (3) patchy saturation. The boundary scheme assigns the wetting phase into nodes near the grain boundary. This scheme mimics the affinity of the wetting phase for the grains (wettability), and the affinity is weighted by an inverse of the distance of the nearest grain node. As a result, the wetting phase is located on small pores or pore throats and the non-wetting phase is at the center of big pores. The random scheme simply assigns the wetting and non-wetting phase randomly. The patchy scheme would put the non-wetting phase at the center of the rock sample. Figure 6.19 shows initial and final fluid distributions of these three schemes with a wetting phase saturation of 50%. The left column of Figure 6.19 shows initial distributions, and final distributions are shown in the right column. Although the initial distributions of two fluids are very different, the final distributions are quite similar for the boundary and the random schemes, while the patchy scheme shows a very different final distribution.

Three sets of relative permeability curves from different initial distributions of two fluids are shown in Figure 6.20. The first thing we notice is that there are bigger

differences in the non-wetting phase than in the wetting phase. The second observation is that the differences in relative permeability among different initial distributions get larger as the wetting-phase saturation increases. The difference in relative permeability of the non-wetting phase becomes really significant when the wetting phase saturation is greater than 50%. The difference increases until $S_w = 70\%$, then decreases and converges toward zero. The main reason for this difference in oil relative permeability is the connectivity of the oil phase.

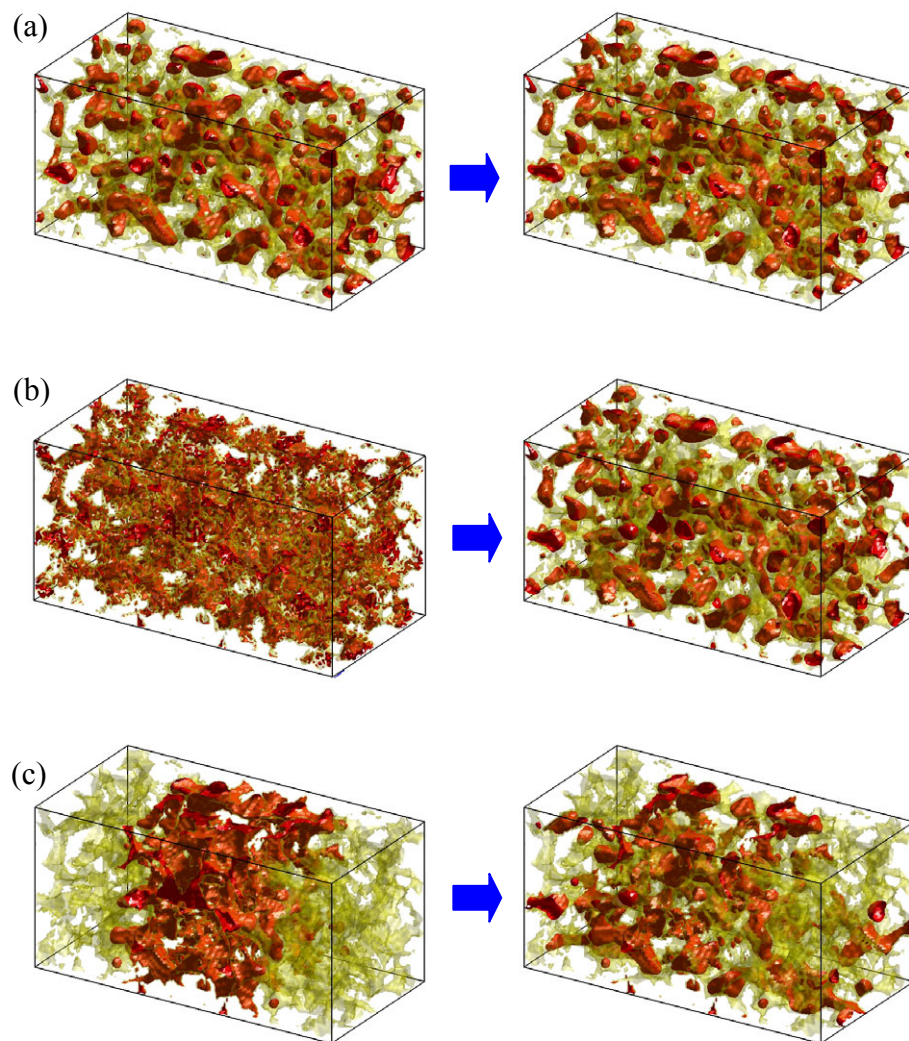


Figure 6.19: Initial (left column) and final (right column) distributions of two fluids in Fontainebleau sandstone by (a) boundary, (b) random, and (c) patchy schemes. The wetting fluid saturation is 50% for all cases. Color notations are the same as in previous figures.

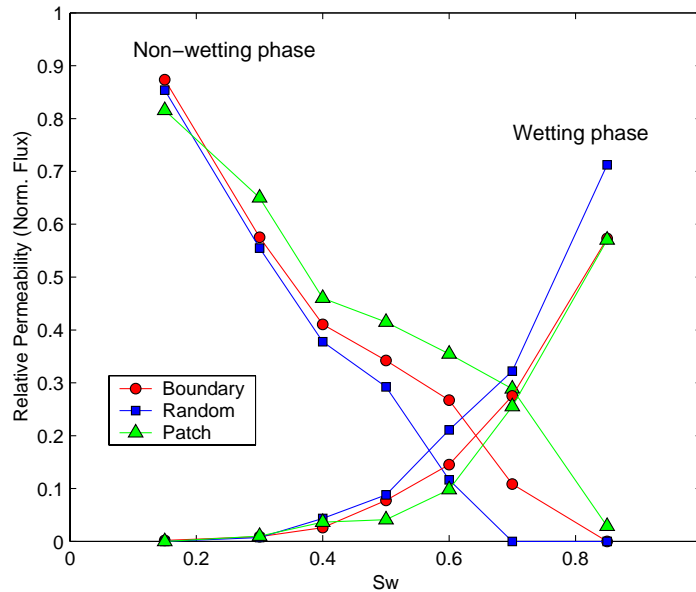


Figure 6.20: Relative permeability curves from different initial distributions of two fluids. Different initial distributions of fluids are denoted by different symbols. The non-wetting phase permeability is more sensitive to the change of initial distribution of two fluids.

As shown in Figure 6.19, the final distribution of the patchy case shows the best connectivity of the oil phase, since it is initially continuous. The boundary scheme and random scheme show very similar final distributions overall. However, in detail, the amount of oil phase that was initially located in small pores in the random scheme is higher than in the boundary scheme. Once the non-wetting phase is disconnected, there can be two consequences. If enough of the non-wetting phase is supplied, the non-wetting phase can move to next pore. Otherwise, it will be trapped in the pore and cannot contribute to the flux.

Figure 6.21 shows the discontinuity of the non-wetting phase in different initial distributions of two fluids with a high water saturation (S_w) of 70%. The first plot, (a)

shows the high flux area from a single-phase flow simulation. The high flux area represents the main path of fluid flow. The next three plots, (b) (c) and (d), are the final distributions of two fluids from the boundary, random and patchy cases, respectively. Obviously, in the random initial distribution, most non-wetting phase is trapped in pores and the continuity of the non-wetting phase is very poor. The boundary case shows better continuity. Much of the non-wetting phase is located along the main pore paths. However, the non-wetting phase is still discontinuous. On the other hand, the patchy case shows excellent continuity of the non-wetting phase through the sample. This good continuity helps the non-wetting phase have a significant relative permeability even in very low non-wetting phase saturation.

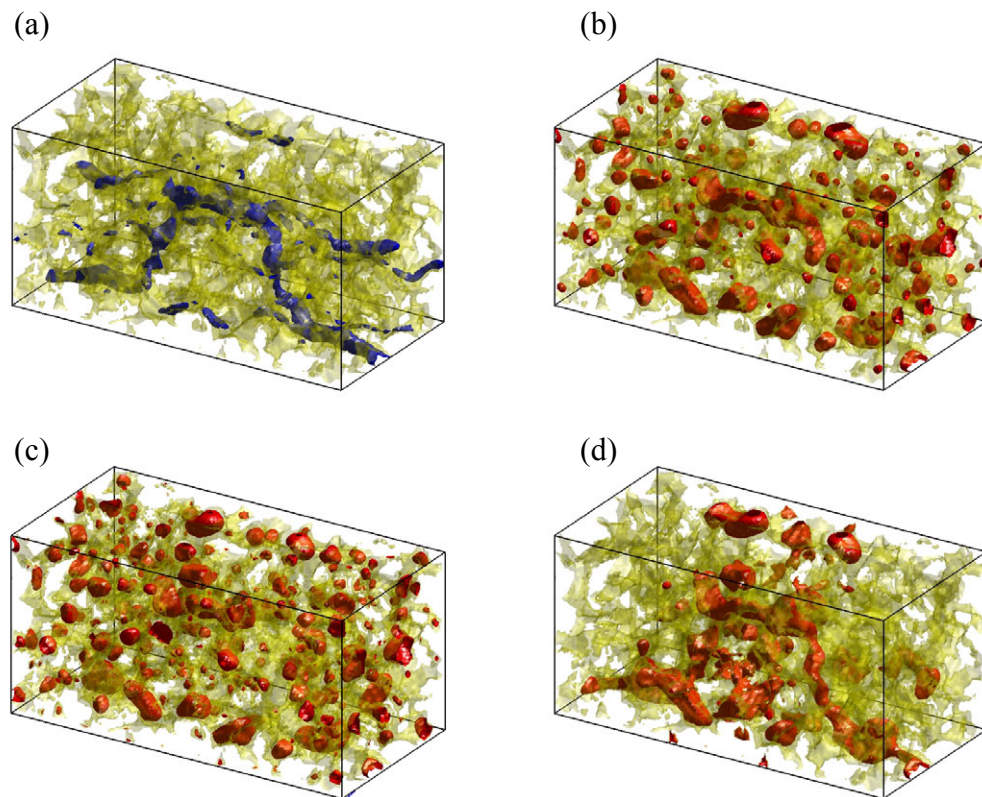


Figure 6.21: Final distributions of two fluids with $S_w=70\%$. (a) High flux pore path from a single-phase flow simulation. (b) Boundary case. (c) Random case. (d) Patchy case.

The continuity of the non-wetting phase plays a significant role in the magnitude of relative permeability for the non-wetting phase. The next question is what a physically reasonable initial distribution of two fluids would be. It should be closely related to the condition of previous oil migration. The random distribution would not be a naturally occurring scenario. The patchy and boundary schemes would be more realistic scenarios, though the pressure and temperature conditions would have a considerable effect on the distributions of two fluids. The change in wettability over time would also be an important factor. Boundary and patchy types can probably be found in the same rock and at the same time. The main conclusion of this section is that the initial distribution of two fluids will have a significant impact on relative permeability. However, considering that our rock model has just a couple of effective paths for fluid flow, we need to investigate whether the current model is representative for the whole rock. This will be discussed in the next section.

The next example uses the unsteady-state simulation. We investigate the effect of initial two-fluid distribution on the sweep efficiency. Two different initial distributions of fluids are shown in the top row of Figure 6.22. The left-hand side is patchy-type saturation, where the oil phase is located at the center of the sample, while the right-hand side shows a final distribution from the imbibition simulation. Both have initially about 65% water saturation. This is one simple description of patchy saturation. First of all, the sample size is too small to represent a reasonable situation of patchy saturation. Secondly the patch would not have a sharp boundary as shown in Figure 6.22. However, we would like to get a rough idea of how this subtle difference in initial distributions will affect the sweep efficiency. As the simulation proceeds, the oil phase is replaced by the water phase. Final distributions of two fluids from both cases show noticeable differences. The residual oil saturation from patchy saturation is slightly larger than from drainage simulation, and the locations of residual oil are also very different. The sweep efficiency also shows a slight difference.

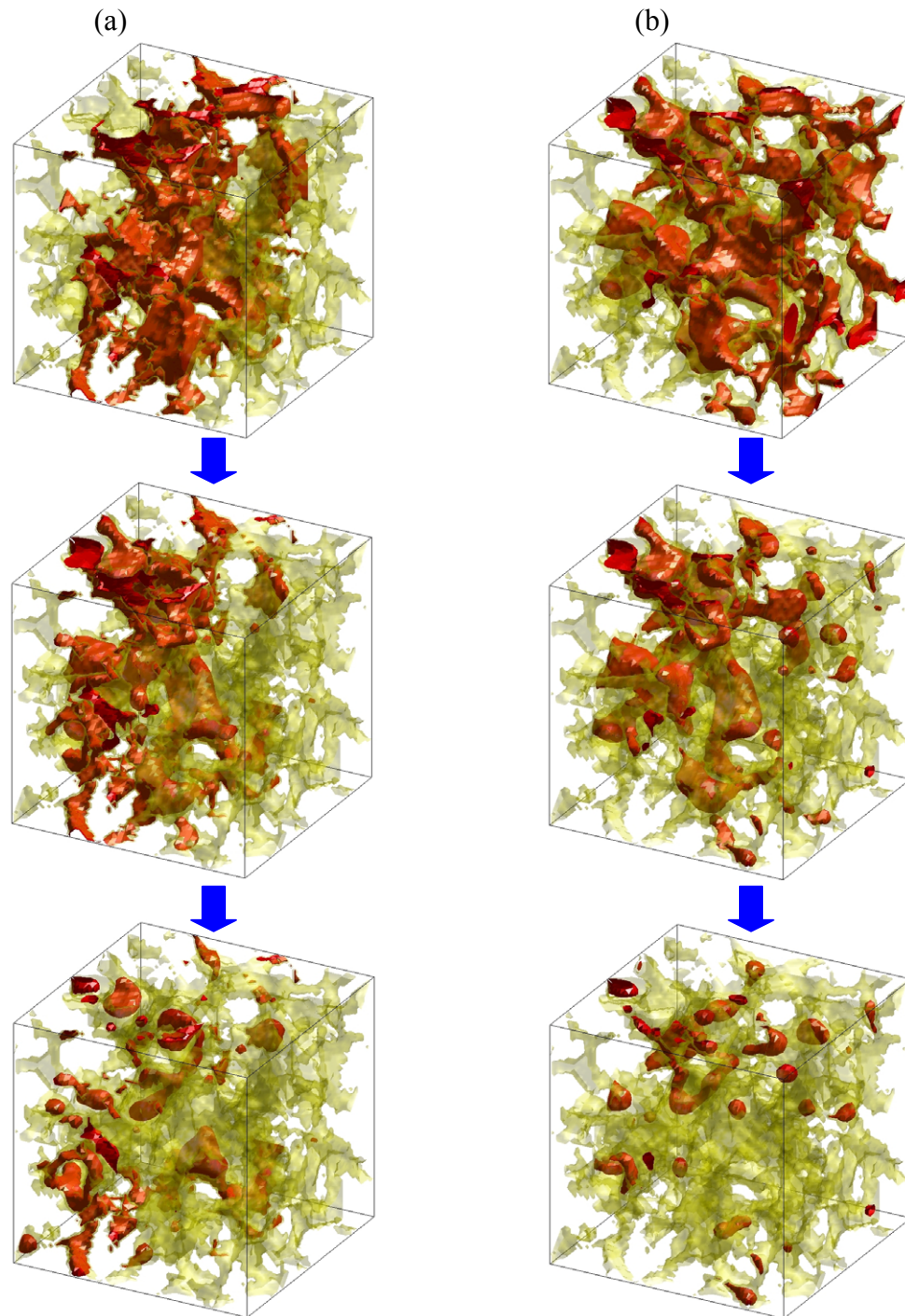


Figure 6.22: Imbibition simulations on (1) a sample with patchy saturation, and (2) a sample from a drainage simulation.

Figure 6.23 shows the saturation change during the drainage simulation. The water replaces oil more efficiently in the sample from the drainage simulation. The figure also shows more clearly about the difference in residual oil saturation. From this simple example, we observe that the computational code can simulate the subtle details of two-fluid distributions in pore geometry realistically and that two-phase flow phenomena can be more complicated with all these minor details. At macroscopic scale, the details may contribute in a predictable way; however, we really need to understand the dynamics at pore scales to understand two-phase flow more thoroughly.

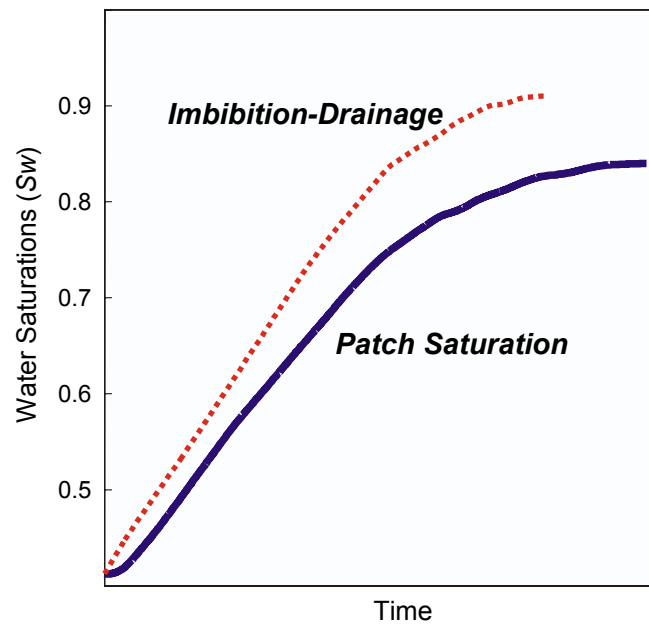


Figure 6.23: Saturations curves for sweep efficiency. The two initial conditions come from imbibition simulation and from patchy saturation.

6.7 Future Applications and Implications

We have seen that the two-phase Lattice-Boltzmann implementation can simulate two-phase fluid flow in real pore geometries. We also showed a simple application: the

effect of initial distribution of two fluids on relative permeability and sweep efficiency.

The implementation is now ready for more complicated applications of two-phase fluid flow in porous media as a module. In Chapters 4 and 5, we could use our methods to calculate absolute permeability, estimate permeability from thin sections, and model diagenesis modeling with single-phase fluid flow. These applications are possible because the Lattice-Boltzmann method can handle very complex pore geometries without any modification or simplification. By the same token, the two-phase Lattice-Boltzmann flow simulator can be applied to the same applications: prediction of relative permeability from thin section and diagenesis modeling with two-phase fluid flow.

The prediction of relative permeability has a big impact on industrial applications, since relative permeability is probably the most important parameter in oil reservoirs, and the measurements are very difficult and time-consuming. Prediction of relative permeability from thin sections will be a big innovation in the oil industry if it is reasonably accurate. Diagenesis modeling with two-phase fluid flow will be more appropriate than with single-phase fluid flow, since underground situations involve multiphase fluid conditions, and diagenesis mechanisms are strongly influenced by two-phase fluids in porous media.

Another important application is capillary pressure curve simulation. The capillary pressure curve is a more direct and standard measurement in oil reservoirs for quantitative determination of reservoir rocks. The numerical simulation of the capillary pressure curve can be easily done with the two-phase Lattice-Boltzmann flow simulator by simply repeating the simulation with different macroscopic pressure gradients.

6.8 Conclusions

We developed and implemented two-phase Lattice-Boltzmann flow simulators. We then performed two-phase flow simulations on digital rock samples to understand the links between pore geometry and relative permeability. The main results are as follows:

- (1) The implementations simulated the steady-state measurement of relative permeability reasonably well and are still much more cost-effective than actual

laboratory measurements, even though the calculation time is longer than for the single-phase flow simulation.

- (2) Two-phase LBM can simulate realistic two-phase flow in porous media and replicate many two-phase flow phenomena, such as Laplace's law, contact angle, capillary pressure, and snap-off in a pore doublet.
- (3) Nonlinearity between flux and pressure gradient was observed through steady-state simulation and can be explained by the fact that the controlling force, capillary pressure, is not linear.
- (4) The effect of initial distributions of two fluids on relative permeability is bigger for oil than water, especially when the oil saturation is low *i.e.*, the oil phase is discontinuous. Patchy saturation would have a considerable impact on oil recovery by affecting both sweep efficiency and residual oil saturation.
- (5) The two-phase LBM can be a tool to explore two-phase flow at pore scales in complex, realistic pore geometry, and can compliment many laboratory measurements that are very difficult or time-consuming.
- (6) A few immediate applications using the two-phase LBM include relative permeability estimation from thin sections, diagenesis modeling with two-phase fluid flow, and capillary pressure curve simulation.

Chapter 7

Parallel Lattice-Boltzmann Flow Simulations and Its Implications

7.1 Abstract

Numerical flow simulations using the Lattice-Boltzmann method have drawn great attention, as they can simulate various fluid flow situations that are difficult in the laboratory. However, these simulations are very computation-intensive and time-consuming even with modern computing power, especially for two-phase fluid flow in porous media. Since parallel machines are now widely available and most parts of the Lattice-Boltzmann algorithm are local operations, a parallel implementation is a good solution for more efficient simulations. The main bottleneck in parallel algorithms is the communication overhead. To minimize the communication overhead, we developed several optimization schemes, which are customized for the Lattice-Boltzmann algorithm. The most important one is a *non-overlapping boundary scheme*. In a typical parallel implementation, each partition shares boundary nodes, which need to communicate with other partitions. However, each node in the Lattice-Boltzmann method has a density distribution containing several floating-point elements, but only a few elements of the distribution need to communicate. The non-overlapping communication scheme does not

require sharing nodes: it simply sends and receives only the necessary elements out of the total distribution. We can save approximately 80% communication cost with this non-overlapping boundary scheme. Secondly, a simple bisectional partitioning is not appropriate, as our pore structure is a very complex unstructured mesh. We used a *graph partitioning method*, which enables us to minimize the partition boundaries. The graph partition method gives us a small boundary area between partitions, which is usually located at pore throats. This reduces the communication cost by more than half. Finally, we implemented *optimized communication ordering* and *weighted graph partitioning*. One partition generally needs to communicate with many neighboring partitions, and each pair has a different size of data to communicate. Since the next iteration is only possible when the communication finishes, a poor ordering of communication will cause many processors to be idle. Communication ordering minimizes the idle time by finding an optimized order of communication pairs. The weighted graph partitioning also minimizes the idle time by improved communication balancing. Each of these optimizations gives 20-30% reduction in total communication time. With all the optimized schemes, we can achieve very good performance on both a shared-memory machine and a distributed-memory machine. The optimized two-phase simulation code could achieve a speedup of 12 with 14 processors, while non-optimized code gave only 4 times faster code with 14 processors.

With efficient parallel implementation, we can perform two-phase flow simulations on bigger pore geometries to investigate the representative elementary volume (REV) for two-phase flow. We found that the REV for two-phase fluid flow is larger than that for single-phase fluid flow. The reasonable size for two-phase flow simulation would be larger than 20 mean pore size ($L=20a$). In addition, we can attack more complex problems with the efficient parallel implementations, such as capillary pressure curve simulations, diagenesis modeling with two-phase fluid flow, and estimation of relative permeability from thin sections. These applications require a set of two-phase flow simulations and can be simulated efficiently with these parallel implementations of the two-phase Lattice-Boltzmann method.

7.2 Introduction

The development of a computationally efficient two-phase Lattice-Boltzmann simulator for relative permeability prediction is an important technical contribution, as it is based on fundamental physics and it can handle complex pore geometry. The capability to use the real pore structure geometry is a particular strength of the model, compared to the more widely used pore network simulators. In addition, it can complement many difficult and time-consuming laboratory measurements of two-phase flow. We have already seen in the previous chapter that two-phase flow simulation can replicate physical two-phase fluid flow in realistic and complex pore structures. However, the computation time of the two-phase flow simulation is still quite large even with modern computers. In addition, we need a finer grid structure of our digital rock in two-phase flow simulations than in single-phase flow simulations, because we need to resolve two interfaces – the grain/pore boundary and the boundary between wetting and non-wetting fluids, while the single-phase flow simulation needs to resolve just the grain/pore boundary. This requires a more robust and efficient implementation that can handle the bigger problem and can finish the calculation in a reasonable amount of time. With the recent and wide availability of parallel machines and standardized libraries (Geist et al., 1994; Karypis and Kumar, 1998a; Gropp et al., 1999; Chandra et al., 2000; Balay, et al., 2002), the parallel implementation of two-phase flow simulation could be a good solution. In addition, the algorithm of the Lattice-Boltzmann flow simulation has many local operations, which do not require communication, and are well suited for parallelization.

There have been a few studies on parallel Lattice-Boltzmann flow simulations. Skordos (1995) implemented two different parallel codes for single-phase flow simulation – one with the Lattice-Boltzmann method and the other with an explicit finite difference method. He used a common partitioning method, simple bisectional decomposition. He investigated how the total problem size, the size of each partition and number of processors affect performance. He had about 50% of efficiency with more than 15 processors. In other words, the parallel code with 20 processors ran about 10 times faster than its serial counterpart. Amati et al. (1997) used a parallel Lattice-Boltzmann code for turbulent channel flow. They used up to 512 processors on Quadrics, a family of

parallel Single Instruction-Multiple Data (SIMD) machines. They could achieve performance close to ideal linear speedup due to the efficient communication between processor nodes. Martys et al. (1999) performed multiphase flow simulations on an SGI Onyx with 12 R10000 processors and an IBM SP2 with 37 RS/6000 processors. They could also achieve good speedups, slightly below the ideal linear speedup.

Most previous studies followed the general parallelization approach and special classes of parallel machines. In this work, we will focus on optimizations customized for the Lattice-Boltzmann algorithm, not just parallelizing a pre-existing serial code. In addition, our implementation focuses on good performance with distributed-memory machines, such as cheaper and standard PC clusters.

- (1) Minimization of data to be communicated: *non-overlapping partition*.
- (2) Minimization of the boundary between partitions: *graph-partitioning method*.
- (3) Minimization of the waiting (idle) time: *ordering of communication pairs and weighted graph-partitioning*.

Each optimization scheme will be covered in detail and the performance increase at each optimization will follow later in this chapter.

7.3 Motivations

The direct motivation of the parallel implementation came from the simulated relative permeability curves. Figure 7.1 shows a digital Fontainebleau sandstone sample that is used for two-phase flow simulations. The actual sample size used for the two-phase flow simulation is represented by colored boxes, the length scale (L) of which is about 10 times the autocorrelation length or characteristic pore size (a). From many single-phase simulation experiments, we are confident that this size of digital rock can represent the single-phase fluid flow of the whole rock. However, the relative permeability curves in Figure 7.1 tell us a different story. The choice of the digital rock sample location can give quite different relative permeability results. Since the Fontainebleau sandstone is very isotropic, homogeneous and well sorted, we have a good reason to believe that the

relative permeability of the sample should be more or less homogeneous as well. One possible explanation of the discrepancy is that the size of the digital rock sample may not be sufficient to represent the two-phase fluid flow of the whole rock

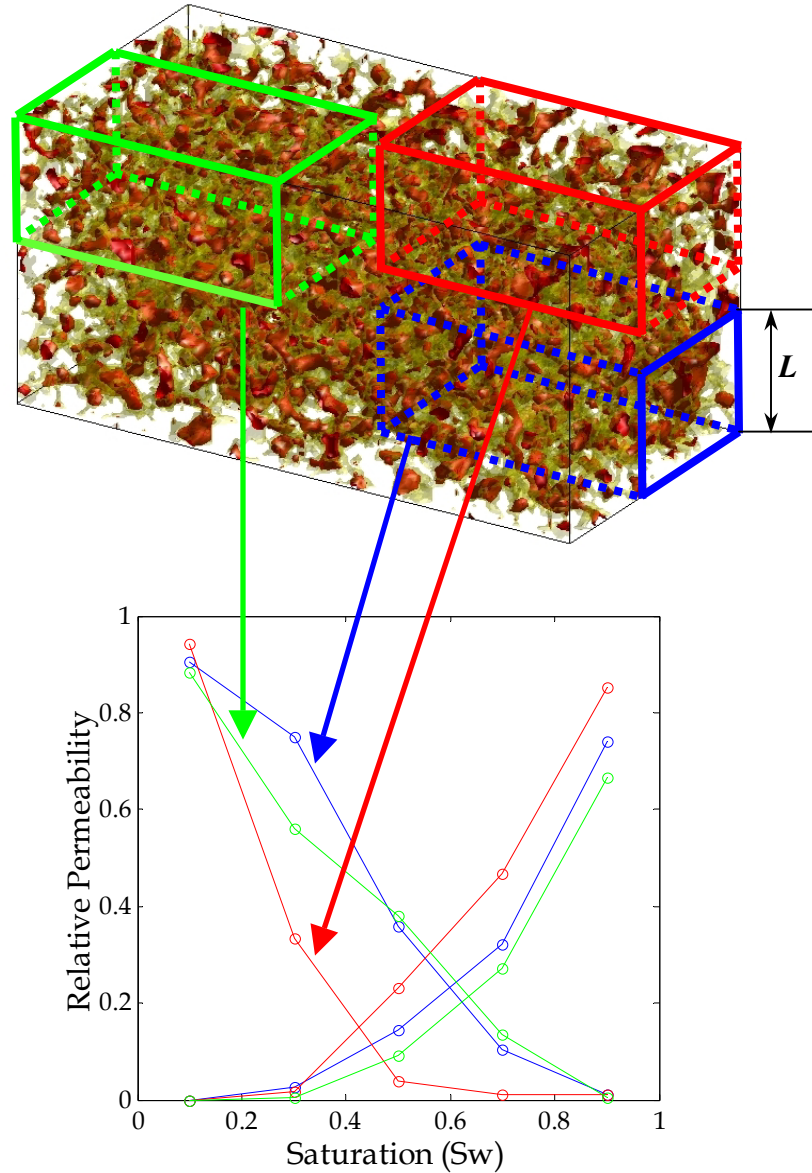


Figure 7.1: Pore geometry of a Fontainebleau sandstone (top). The green, blue and red boxes denote the size of the digital samples for two-phase flow simulations. The length scale of the sample (L) is about $10a$ (a : autocorrelation length). The relative permeability curves from different digital rock samples (bottom). The non-wetting relative permeability is sensitive to the choice of the rock sample location.

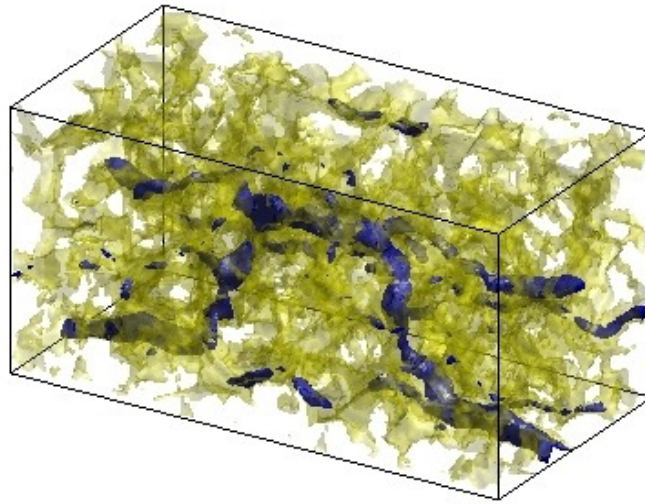


Figure 7.2: The effective fluid path in a Fontainebleau sandstone sample. Green color denotes pores, and blue areas are effective flow paths where the local flux is not negligible.

The representative elementary volume (REV) is the minimal volume of pore geometry, the statistical average of which represents a macroscopic property. The basic idea of REV was briefly covered in Chapter 4. We have seen that the size of $L > 7a$ is representative for porosity and the size of $L > 10a$ is usually representative for absolute permeability in Chapter 4. Figure 7.2 suggests that the size of $L \approx 10a$ may not represent the whole rock in terms of relative permeability. The blue areas in the figure are regions with non-negligible flux from a single-phase flow simulation. We can see this sample has two effective fluid paths. Since two-phase fluid flow is controlled by a nonlinear force, the capillary pressure, this size of sample probably is not big enough to represent two-phase fluid flow of the whole rock. In other words, these small samples cannot represent all possible situations of two-phase fluid of the whole rock, such as trapped oil, irreducible water or snap-offs. In order to check the REV, we need to perform two-phase flow simulations on larger samples, which are almost prohibitive to run on a single computer. This is one of the main reasons to implement efficient parallel Lattice-Boltzmann flow simulators.

7.4 Overview of Parallel Implementation with MPI

Before the details of the parallel implementation, general parallel implementation techniques and libraries are briefly covered in this section. More detailed information about parallel methods and interfaces can be found in Golub and Ortega (1993), Culler et al. (1998), Dongarra et al. (1999), and Fosdick et al. (1999).

7.4.1 Parallel Machine Architecture

We intend to achieve two objectives with parallel computing: faster calculation speed and ability to run larger problems, which are very difficult or impossible to achieve on single-processor machines. Although there are many kinds of parallel machines, we can categorize them into two types: shared-memory machines and distributed-memory machines. A shared-memory machine contains multiple processors, and all processors share one large main memory (Figure 7.3). Since the processors are connected into a BUS or a switch, the latency and bandwidth are good. However, the cost of building a shared memory machine would be very high and the architecture is complex. On the other hand, a distributed-memory machine consists of multiple computers connected in a network, which will cause poor latency and bandwidth of memory. However, it is very cost-effective, and we can build large clusters more easily.

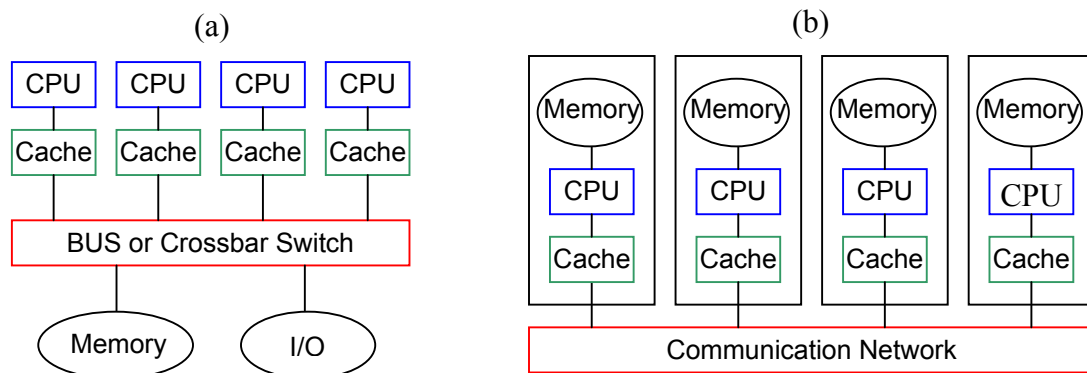


Figure 7.3: Two typical parallel machines: (a) a shared-memory machine and (b) a distributed-memory machine.

We used both types of machines to test the parallel implementations. The shared memory machines are a Sun Enterprise 6500 with sixteen 400MHz UltraSparc II Processors and 16 Gigabytes of memory, and a Dell Precision Workstation with dual 1.7GHz Xeon processors and 1 Gigabyte of memory. The distributed memory machine consists of sixteen identical PC-clones, each of which has one 1.4 GHz Athlon processor and 1 Gigabytes of memory. The distributed memory machine has a Linux operating system, with each processor connected to a 100Mbit network switch.

7.4.2 MPI (Message Passing Interface)

There are several interfaces for the parallel implementation, including MPI (Message Passing Interface), PVM (Parallel Virtual Machine), HPF (High Performance Fortran), OpenMP and Pthread (Geist et al., 1994; Gropp et al., 1999; Chandra et al., 2000). The first three are message-passing interfaces and the last two belong to shared-memory programming. Shared memory programming interfaces are exclusively for shared memory machines and cannot be used on distributed memory machines, while message-passing interface can be used on both types of machine. More detailed information about each interface can be found in Culler et al. (1998). We used MPI over other interfaces for several reasons (MPI Forum, 1994; Pacheco, 1997). (1) It is the only message-passing library that can be considered a standard. (2) There is no need to modify the source code when porting codes from one platform to another. (3) Vendor implementations should be able to exploit native hardware to optimize performance. (4) A variety of implementations are available, both by vendors and in public domain. Although MPI provides more than 200 functions for parallelization, normal codes are likely to use only a few functions, which gives us an easy way to have parallel implementations.

7.4.3 General Parallel Implementation

There are several approaches to make a problem parallel. SIMD (Single Instruction-Multiple Data) and MIMD (Multiple Instruction-Multiple Data) are among these. The SIMD approach is a more straightforward and better-performing scheme, and was chosen in this study. With the SIMD scheme, we partition a whole data structure into several sub-blocks (or partitions) and assign each sub-block to a different processor. Each

processor handles its own dataset with the same algorithm. When a processor needs data from other partition(s), we use a point-to-point communication provided by MPI.

We start with a simple example of digital data structures in a 2D grid (Figure 7.4(a)). We will call a grid point a *node*. The brief flowchart of a general parallel implementation begins with partitioning the whole domain into smaller partitions. We have four partitions in the example. Each partition will be assigned to a different processor to apply an algorithm. Now we can see the partitions should ideally be as close to the same size as possible (load balancing). If the sizes of partitions are quite heterogeneous, a processor with less data will finish its calculation earlier and will wait until other processors finish their calculations. Thus we are wasting processor time with idle processors.

When an operation is local, each processor will take care of its own data. However, each node will need neighboring nodes at a certain point. The most common example is a 2nd-order finite-difference operator on a discrete grid. The finite-difference operator requires the information of neighboring nodes. If the neighboring node is located on another partition, this will require *communication* between the partitions. A common approach for the communication is to introduce shared nodes or so-called ghost nodes at the boundary (Figure 7.4(b)). Now each partition has two kinds of nodes: local nodes that were originally assigned to the partition and shared nodes from neighboring partitions. When the value of a shared node is modified, other partitions that share the node should update their value with the modified one. Many numerical simulations including the Lattice-Boltzmann method involve iterative algorithms to solve physical equations. Thus data communication is required at every iteration. Since the communication is a very expensive operation, minimization of communication is the main task for parallel programming. The bottom line for better performing parallel codes is to have a good partitioning method that gives a perfect load balance with minimal communication cost. In most cases with regular grid structures, bisectional partitioning gives good performance.

Even after we have the optimal partitioning, there is another important optimization – the order of communication. As seen in Figure 7.4, Partition I has shared nodes with all other partitions. Assume that Partition II, III and IV are trying to communicate with

Partition I at the same time. If Partition II happens to communicate with Partition I first, Partition III and IV simply will be idle and wait for their turns. It would be much better if Partition III communicates with IV during this time. This simple example shows how the ordering of communications will impact the performance of a parallel implementation. In our example in Figure 7.4, it is relatively easy to achieve optimal ordering. However, the optimization of ordering is more complicated with unstructured grids, which is the case with the Lattice-Boltzmann algorithm

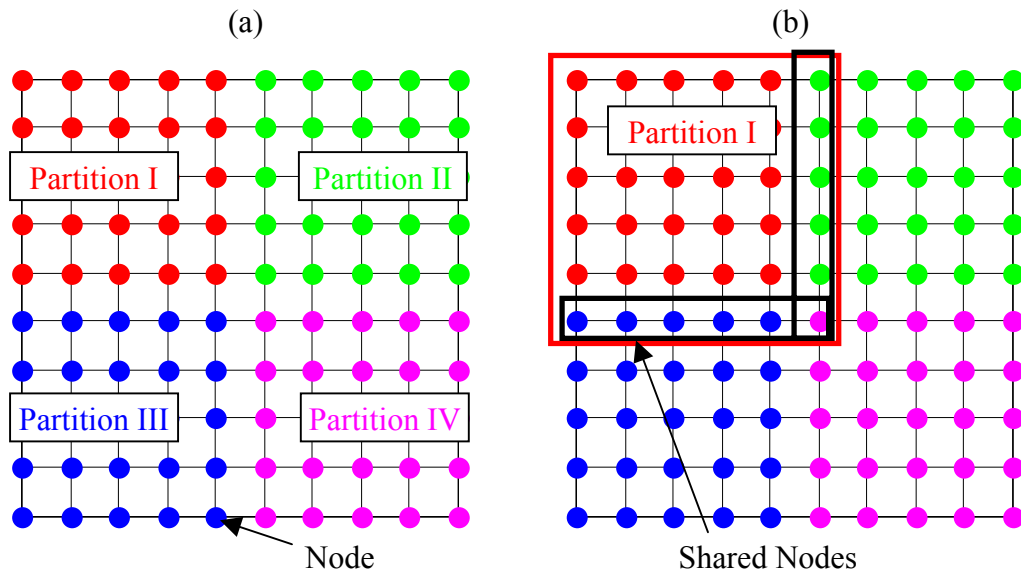


Figure 7.4: (a) Four partitions for 2D digital data. (b) Partitions with sharing layers. Red square denotes the area of Partition 1.

7.5 Parallel Lattice-Boltzmann Method

In the previous section, we mentioned general parallel procedures with a simple example. There are several differences between the general parallel implementation and the implementation of the parallel Lattice-Boltzmann method.

First of all, each node in Lattice-Boltzmann simulation has a distribution of values instead of one value. With a 3D FCHC (face-centered hypercube) grid, there are 18 floating-point elements at each node. At the communication step, however, we do not

need all the elements. Instead, each neighboring node needs only a few elements depending on the node geometry. If we use the sharing-node partition, all elements including the unnecessary ones will be sent and received. To avoid this unnecessary communication, we developed a *non-overlapping partition* scheme, in which only the necessary elements of a boundary node are communicated. This reduces the total communication to about 1/5 of the original size.

Secondly, the grid structure of our pore geometry is not regular. Although we use a regular grid (FCHC) in describing pore geometry, we use only the pore nodes not the grain nodes. Figure 7.5 shows a typical digital rock, and only the pore nodes, shown in red, will be in the grid structure. Thus the grid for the Lattice-Boltzmann flow simulation is unstructured, or irregular. Due to the complex grid geometry, the role of partitioning becomes much more important. We used a *graph-partitioning* algorithm (Karypis and Kumar, 1998a; 1998b; 1998c) that enables us not only to minimize the communication but also to have a good load balance among partitions.

Finally, since the grid is unstructured, the size of the communication between each pair will be different. We developed a method to find an optimized *ordering of communications* and used a *weighted graph partitioning* method (Karypis and Kumar, 1998a) to minimize the idle (waiting) time during communication.

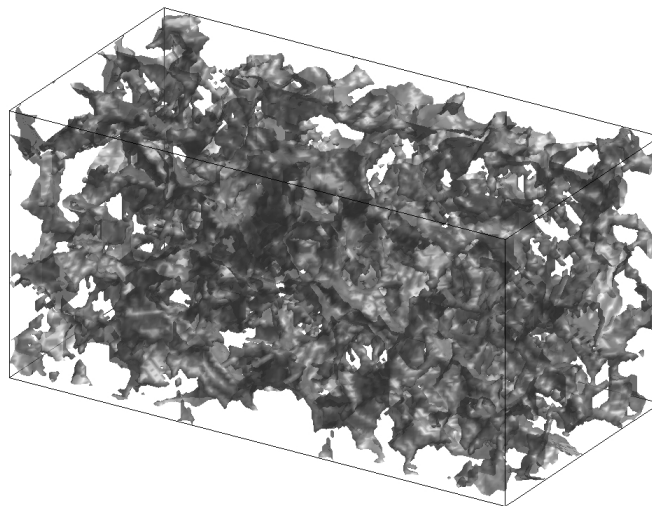


Figure 7.5: Pore geometry of a Fontainebleau sandstone. Gray denotes pore.

7.5.1 Non-overlapping Boundary

There are two typical steps in the Lattice-Boltzmann method – collision and propagation (see Chapter 3). In the collision step, we create a new density distribution by the discrete Boltzmann equation. The actual communication occurs in the propagation step, where each element of the distribution moves to a neighboring node. Figure 7.6 describes the procedures schematically in a 2D FCHC grid, each node of which has eight elements. Figure 7.6a shows the initial density distribution of the center node. The center node has a new distribution at the collision step, and each element of the distribution propagates to the corresponding neighboring nodes at the propagation step.

Figure 7.7 shows how non-overlapping partition can be achieved. We simply move necessary elements from one partition to the other, in this case, only six elements from Partition I to Partition II. If we choose to use the sharing-nodes scheme, we need to send 24 elements to Partition II, since each node contains eight elements in 2D. In this simple 2D example, the non-overlapping partition scheme reduces the size of the communication to 1/4 of the original size. This reduction is even better with a 3D grid. The size of communication is reduced to 1/5~1/6 in a 3D FCHC grid, which saves more than 80% of the communication bandwidth.

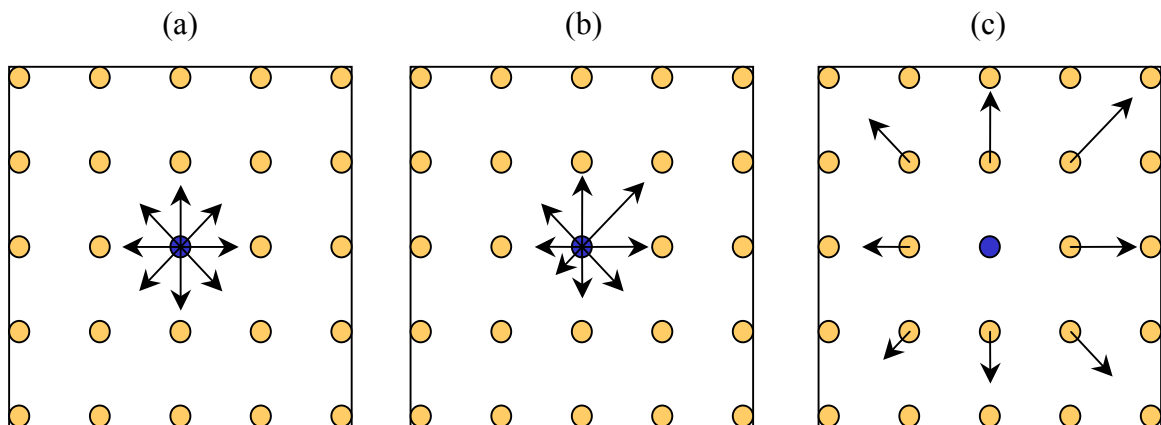


Figure 7.6: Procedures of the Lattice-Boltzmann algorithm in a 2D FCHC grid. (a) Initial state of the density distribution. (b) Collision step. (c) Propagation step.

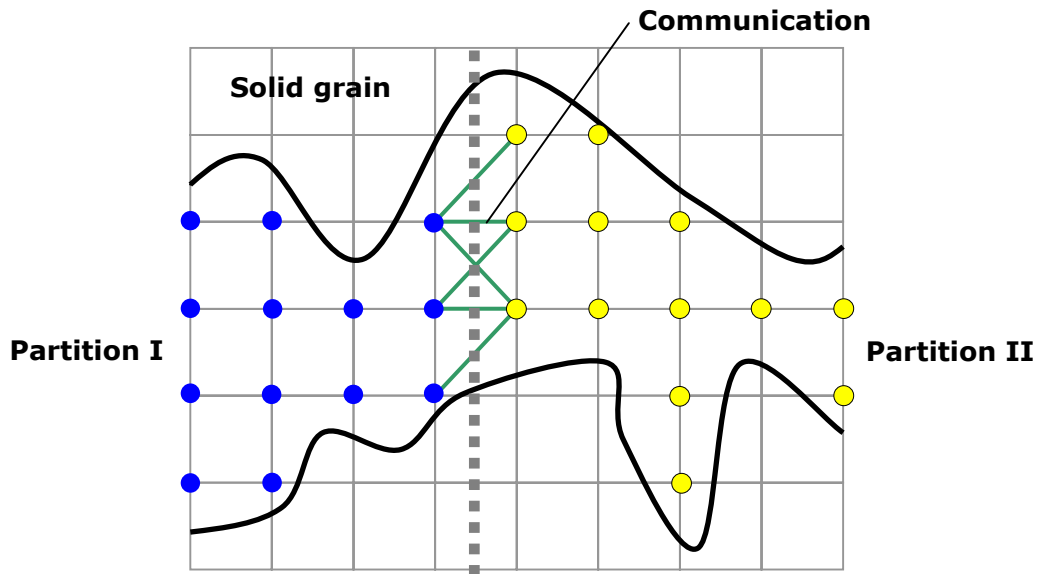


Figure 7.7: Schematic diagram of the non-overlapping partition geometry.

We have an extra cost for the non-overlapping boundary scheme. We need to calculate the mapping indices for the elements at boundary nodes, since this communication is now element-based, not node-based. However, this is a one-time calculation for the entire simulation, while the communication occurs in every iteration. Thus the extra cost is generally negligible compared to the benefit in communication.

7.5.2 Graph Partitioning

A good partitioning algorithm is necessary to achieve good load balancing while minimizing communications. However, finding an optimal partitioning is a very difficult problem (Smith et al., 1996) with unstructured grids. Therefore we usually choose a heuristic approach to obtain an optimal partitioning. We used a Metis graph-partitioning algorithm (Karypis and Kumar, 1998a; 1998b), which is a multilevel partitioning method. Figure 7.8 shows partitioning results with a simple bisectional partitioning and with the Metis graph partitioning. The graph partitioning method gives more complex partition structures than the bisectional partitioning method.

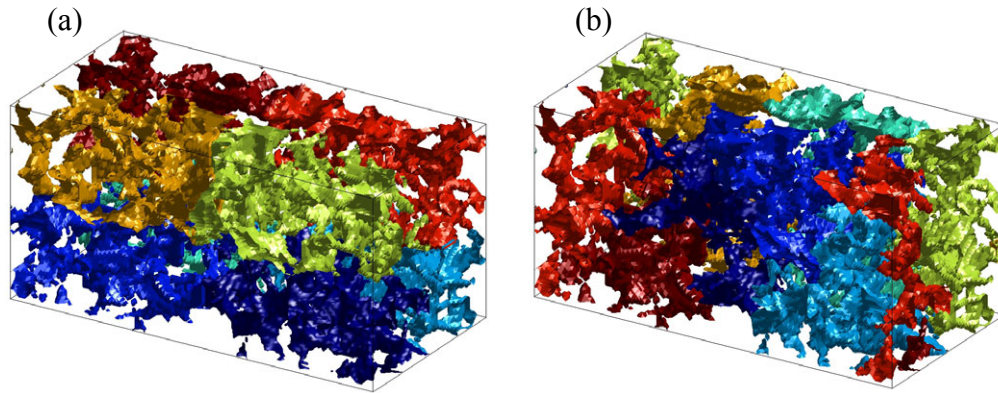


Figure 7.8: (a) Bisectional partitioning and (b) Metis partitioning with eight partitions. Different colors denote different partitions.

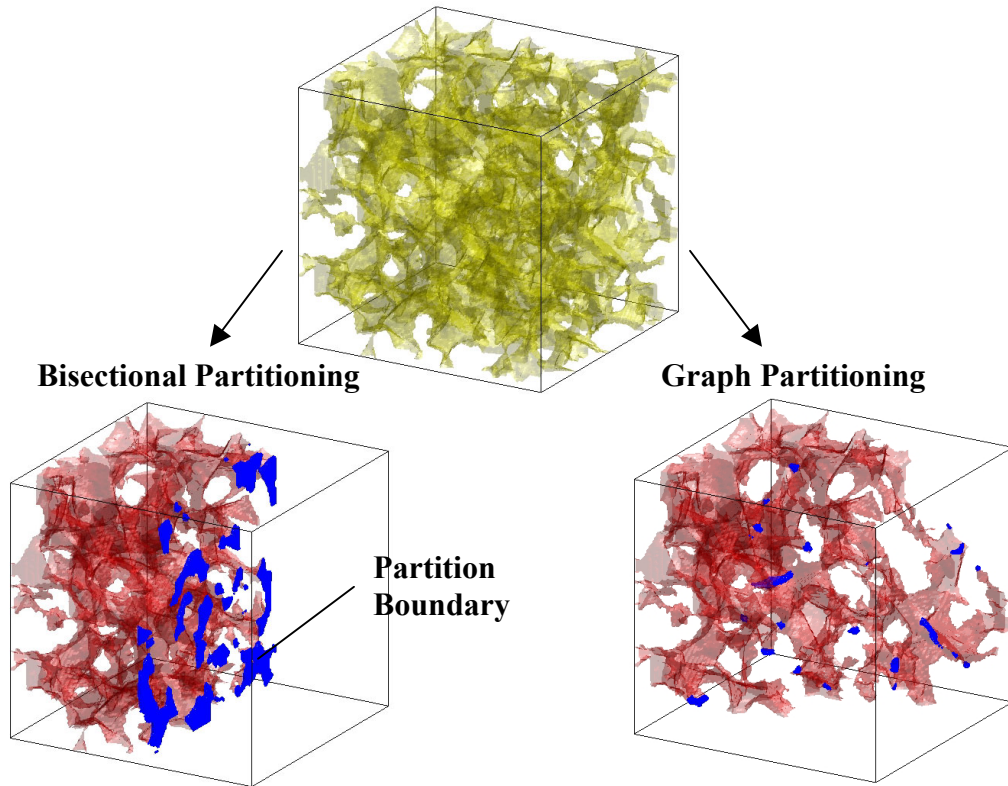


Figure 7.9: Example of bisectional partitioning and graph partitioning on a small digital pore geometry. The graph partitioning method set the partition boundary that has smallest area (pore throats) with good load balancing.

Figure 7.9 shows the benefits of the graph partition clearly. We can easily see that the graph partitioning method set partition boundaries at the pore throat, which can minimize the area of the boundary. The partition boundaries with the graph partitioning method are much smaller than those with the bisectional partitioning. Figure 7.10 shows load balancing and communication cost from both bisectional partitioning and Metis partitioning. Metis gives us a significant reduction of communication as well as perfect load balancing.

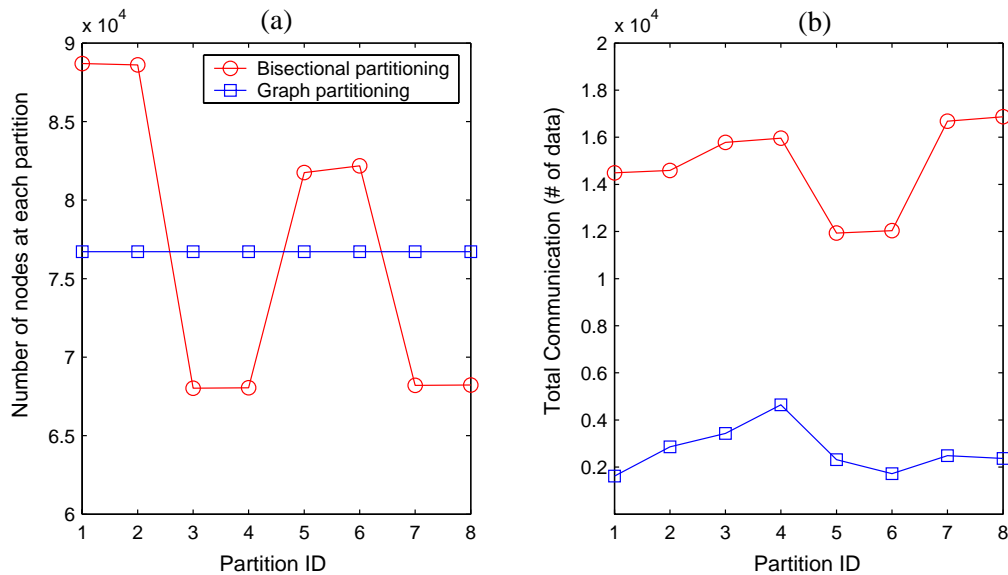


Figure 7.10: Load balances and sizes of communication by the bisectional partitioning and the Metis graph partitioning. (a) Graph partition gives perfect load balancing, while the bi-sectional partitioning shows more than 10% fluctuation in the number of nodes of each partition. (b) The communication cost is really reduced by the graph partitioning method, by almost an order of magnitude. The communication size means the number of data to be communicated.

7.5.3 Optimized Communication Order

From the Metis graph partitioning, we have optimized partitions with all communication information. Table 7.1 shows the communication size of each pair of partitions. For example, the size of communication between Partition 1 and Partition 2 is

345. The sizes of communication are quite different, and now the communication order has quite a big impact on the parallel performance. To have an idea about the importance of communication ordering, we will consider an example (Table 7.2). At Step 1, each processor performs a communication with its partner given in Table 2 (Figure 7.11). The problem arises at Step 2. Processors 1 and 2 have already finished the communication and are waiting for their next partners, while processors 3 and 4 are still doing the first communication between them. It would be better for processors 1 and 2 to find other available processors instead of waiting for processors 3 and 4. The goal of communication ordering is essentially to find an optimal communication order to minimize idle time during communications.

Table 7.1: Size of each communication pair. P1 denotes Partition 1 from 8 partitions.

	P1	P2	P3	P4	P5	P6	P7	P8
P1	0	345	189	472	251	0	206	154
P2	345	0	0	1450	240	0	627	198
P3	189	0	0	2406	11	35	574	215
P4	472	1450	2406	0	219	0	0	94
P5	251	240	11	219	0	1173	140	285
P6	0	0	35	0	1173	0	21	496
P7	206	627	574	0	140	21	0	922
P8	154	198	215	94	285	496	922	0

Table 7.2: An example of communication ordering.

	Communication Pairs
Step 1	(P1, P2) (P3, P4) (P5, P6) (P7, P8)
Step 2	(P1, P3) (P2, P4) (P5, P7) (P8, P8)
Step 3	(P1, P4) (P2, P3) (P5, P8) (P6, P7)
Step 4	(P1, P5) (P2, P6) (P3, P7) (P4, P8)
Step 5	(P1, P6) (P2, P5) (P3, P8) (P4, P7)
Step 6	(P1, P7) (P2, P8) (P3, P5) (P4, P6)
Step 7	(P1, P8) (P2, P7) (P3, P6) (P4, P5)

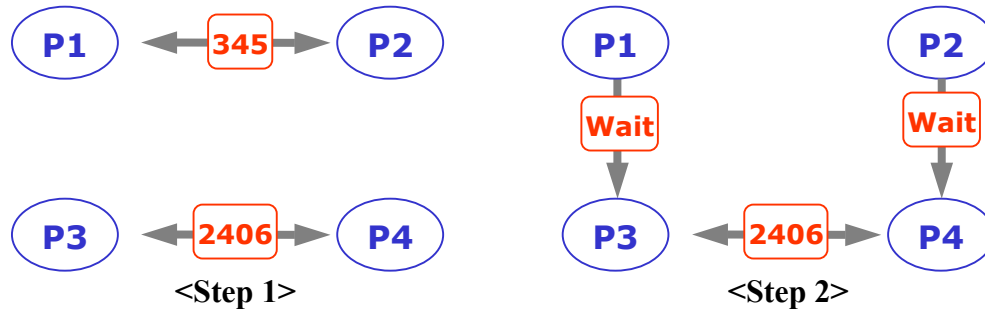


Figure 7.11: Schematic diagram of communication by the order given in Table 2.

Finding the optimal ordering of the communication is also a very difficult problem. However, we can choose a heuristic approach as follows.

- (1) Define an extra parameter for each partition, a cumulative sum of communication time. Set it zero as an initial value.
- (2) Find the biggest number of communication from Table 7.2 and make it the first step.
- (3) Find next available biggest communication until the first step is filled up.
- (4) To find the second step, find the smallest cumulative sum of communication, then find the next available smallest cumulative sum. This combination is the earliest available combination after step 1.
- (5) Update the cumulative sum in the following manner. If the two cumulative sum values of these two partitions are different, it means the one with the smaller cumulative sum should wait until the other partition is available. Thus the final cumulative sum should be a maximal value between the cumulative values from these two partitions. Update both cumulative sum values with the maximal value.
- (6) Repeat (4)-(5) until all communications are completed.

Table 7.3 shows an example of finding an optimal communication order by the procedure described above. Figure 7.8 shows the total communication time of each processor with the optimized order (diamonds) and non-optimized order in Table 7.2, which is regarded

as the best order when the all communication sizes are the same. The optimized ordering gives about 20% less communication time. In unstructured or irregular grids such as the one in the Lattice-Boltzmann method for our case, the sizes of the communication pairs are quite heterogeneous. Thus the optimized ordering will give a non-negligible performance increase.

Table 7.3: Procedure to find an optimal order of communication with eight partitions. Numbers in parentheses are partition numbers. The number outside parenthesis is a cumulative communication size in a given partition and at a given step.

	P1	P2	P3	P4	P5	P6	P7	P8
Step 1	(1)345	(1)345	(4)2406	(3)2406	(6)1173	(5)1173	(8)922	(7)922
Step 2	(7)1128						(1)1128	
Step 3		(8)1120						(2)1120
Step 4		(7)1755					(2)1755	
Step 5	(8)1282							(1) 1282
Step 6	(5) 1533				(1)1533			
Step 7						(8)1778		(6)1778

7.5.4 Communication Balancing – Weighted Graph Partitioning

Even with optimized communication ordering, the efficiency of communication becomes worse as the heterogeneity of the communication sizes increases. Thus making the communication sizes as homogenous as possible is very helpful for efficient communication (communication balancing). We can use a weighted graph partitioning (Karypis and Kumar, 1998a) for balancing the communication number. Briefly, the weighted graph partitioning tries to balance the communication sizes at the cost of less load balancing. Usually the cost of load balancing is much smaller than the advantage from better communication balance. Table 7.4 shows the comparison between unweighted and weighted graph-partitioning schemes. Since the next calculation is only possible after the all communications are finished, the maximum communication time is important. The weighted graph partitioning gives about 20% less communication time. This reduction is graphically shown in Figure 7.12.

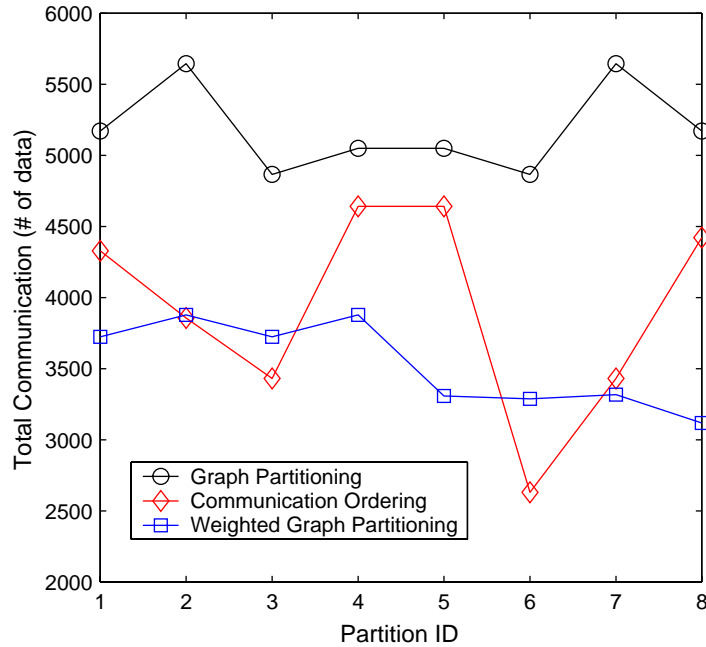


Figure 7.12: Total communication size at each partition by a conventional ordering (circles), the optimized ordering (diamonds) and the optimized ordering with a weighted graph partitioning (squares).

Figure 7.13 summarizes all the optimizations. Each optimization was applied on the top of the previous optimization. Brief observation on the section is as follows:

- (1) The non-overlapping partition method reduces the communication cost the most among four optimizations. Note the line breaks at the y-axis in Figure 7.13. The communication with the non-overlapping partition is about six times smaller than that with the conventional shared boundary partition.
- (2) Graph partitioning gives another big improvement in reducing the communication cost, which is more than three times smaller than with a simple bisectional partitioning. The improvement depends on the geometry of pore structure.
- (3) Optimization of order minimizes the overall waiting (idle) time. It does not give a big improvement. However, we found that we can get 20-30% smaller communication cost using the optimized ordering.

(4) Weighted graph partitioning gives better communication balance and eventually more efficient communication. It usually gives 20-30% less communication cost.

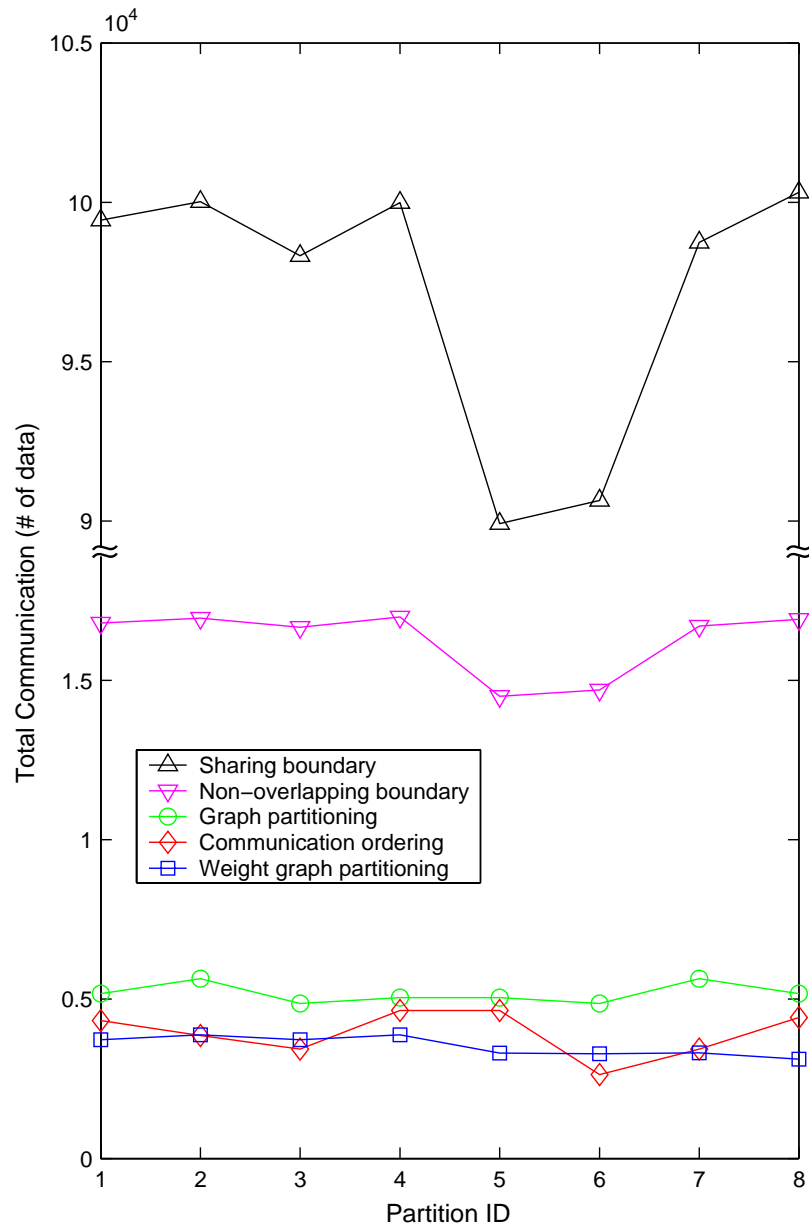


Figure 7.13: Improvement of total communication sizes with each optimization.

Table 7.4: Communication balance with weighted and unweighted graph partitioning.

	Total communication size (number of data)	
	Weighted graph partitioning	Unweighted graph partitioning
P1	3724	1617
P2	3652	2860
P3	3443	3430
P4	2667	4641
P5	1937	2319
P6	1484	1725
P7	1560	2490
P8	2181	2364

7.6 Algorithms for Parallel LBM Flow Simulations

After completing partitioning and communication mapping, we need to parallelize the Lattice-Boltzmann flow simulation algorithms for both single-phase and two-phase flow. From the partitioning and optimization, we already obtained information about the partition geometry and mapping index tables for communication between partitions. Parallel implementation using MPI is now quite straightforward. We can parallelize the code by simply adding the communication routine when necessary. The parallel algorithms of the Lattice-Boltzmann method are as follows:

Single-phase flow simulation

- (1) Partitioning and set up the communication indices (serial operation)
- (2) Collision step (local operation)
- (3) Apply force term (local operation)
- (4) Propagation step (communication)
- (5) Repeat (2)-(4)

Two-phase flow simulation

- (1) Partitioning and set up the communication indices (serial operation)
- (2) Collision step (local operation)
- (3) Calculating interfacial tension (communication)
- (4) Apply force term (local operation)

- (5) Propagation step (communication)
- (6) Repeat (2)-(5)

We need one communication per iteration with single-phase flow simulation, while two-phase simulation requires two communications per iteration, since we need neighboring information twice per iteration – when calculating the interfacial tensions and when performing the propagation. In addition, the two-phase simulation has more data to communicate, since each node has two kinds of fluids. We will see how the extra communication step and more data affect performance in the next section.

7.7 Simulations and Results

We applied our optimized parallel implementations to simulate two-phase fluid flow in X-ray tomographic Fontainebleau sandstone samples (Figure 7.5). Digital rock samples with sizes of $256 \times 128 \times 128$ and $512 \times 256 \times 256$ were used for two-phase flow simulations. The actual size of the bigger sample is about $4\text{mm} \times 2\text{mm} \times 2\text{mm}$. Calculation time is measured only for the flow simulation part; *i.e.* file I/O and other preprocessing procedures were excluded in measuring calculation time for fair comparison. Three machines described in the previous section were used: a Sun workstation with 16 SparcII processors, a PC with dual Xeon processors and PC clusters with AMD Althon processors.

Performance or Speedup (S_p) is a measure of parallel computing performance, which is given as follows:

$$S_p = \frac{t_1}{t_p}, \quad (7.1)$$

where p is the number of processors, t_p is the calculation time by p processors and t_1 is the calculation time by a single-processor code. In the ideal linear speedup, the speed up will be equal to p ; *i.e.* the parallel code runs p times faster than the single-processor counterpart. Similarly, the efficiency is expressed as follows:

$$E_p = \frac{S_p}{p}. \quad (7.2)$$

Figure 7.14 shows calculation time for single-phase and two-phase flow simulations. Red circles are from the Sun workstation, blue squares from the PC clusters and black stars from the dual Xeon PC. Unfortunately, the Sun workstation was not 100% free of other users during two-phase flow simulations. We could use up to 14 processors from the 16-node PC cluster. The dual-Xeon PC was used just for benchmarking performance. We can notice how well modern personal computers perform compared to the Unix workstation. In the following figures, we will show results only from the Sun workstation (shared-memory machine) and the PC cluster (distributed-memory machine).

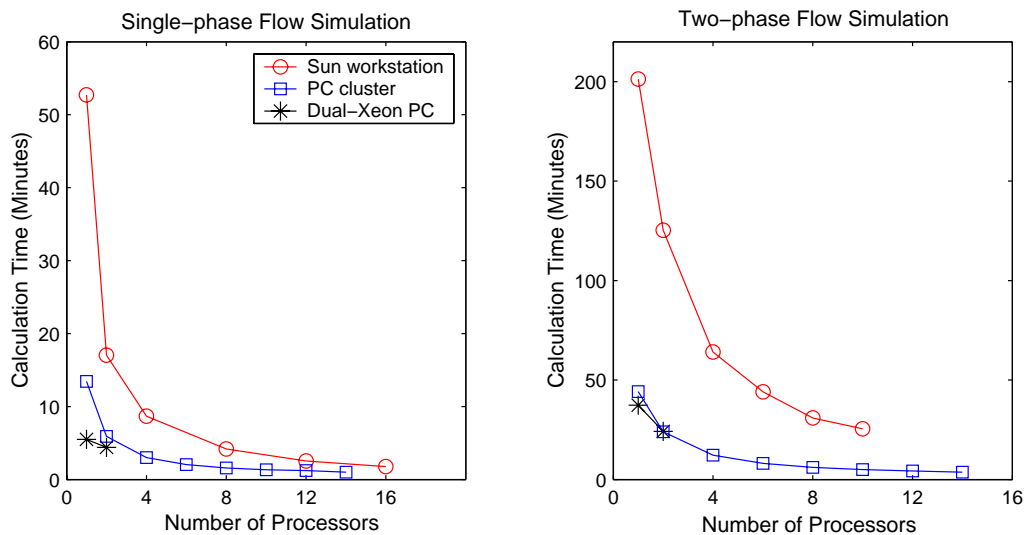


Figure 7.14: Calculation times of single-phase flow simulation (left) and two-phase flow simulation (right).

Figure 7.15 shows the speedup results from the shared-memory machine and the distributed-memory machine. We can observe very interesting results, especially from single-phase flow simulations. Speedup curves exceed the ideal speedups; *i.e.* the performance with p processors is more than p times bigger than the single-processor performance. This looks unphysical, however this is frequently observed and is called super-scalability or superlinear speedup (Parkinson, 1986; Helmbold and McDowell, 1989; Agrawal et al., 1994). Two main reasons for superlinear speedup are shown in Agrawal et al. (1994). The first reason is that in search problems, the termination time

can be reduced when several searches are executed at the same time. The second one is the more efficient utilization of resources by multiprocessors. Our super-scalability mainly comes from the second reason: the better reuse of cache in the system. Modern processors usually have several levels of cache for fast access of memory. When the problem size becomes smaller, the possibility of all necessary data being in the cache increases. As a result, the same operation on a small dataset will finish a lot faster than one on a bigger dataset. The speedup of single-phase simulation on the shared memory machine with 16 processors is about 30. The speedup results of two-phase flow simulation do not show super-scalability, because of the larger and more complicated data structures. However, they are close to the ideal speedups. This is a surprising result, considering the amount of data that we need to communicate per iteration. Our optimizations give a really big impact on the performance of parallel simulations. Another promising observation is that the performance by the distributed memory machine is very good, actually better than the one by the shared memory machine. Even with a very cost-effective and simple parallel machine such as the PC cluster, we can have more than 80% efficiency with two-phase flow simulations (Figure 7.16). The effect of optimization in the simulation will be covered in detail in the next section.

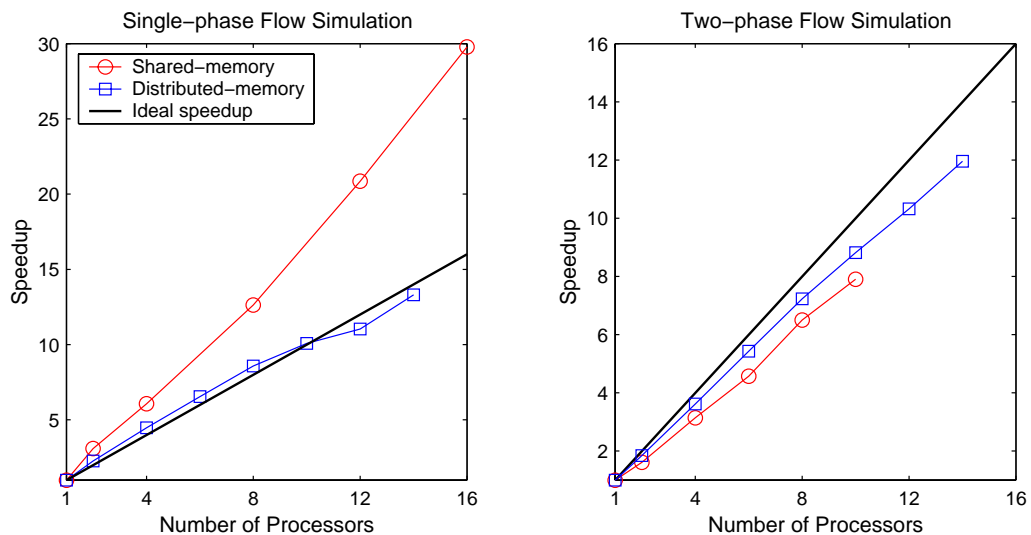


Figure 7.15: Speedup results for single-phase flow simulation (left) and two-phase flow simulation (right). So-called super-scalability is observed on both a shared-memory machine and a distributed-memory machine with the single-phase flow simulation.

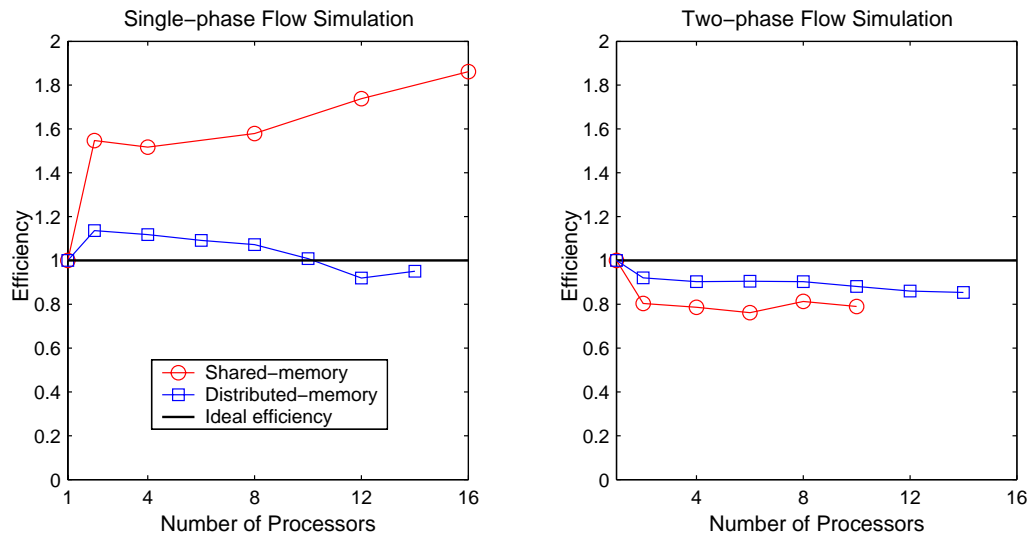


Figure 7.16: Efficiency curve from single-phase flow simulations (left) and from two-phase flow simulations (right). Either case has an efficiency better than 80%.

Figure 7.17 shows snapshots of drainage and imbibition simulations with a bigger rock sample than the one we used in Chapter 6. The whole simulations took 8 hours with 14 processors, while it took about three days with a single-processor machine. We could also perform a single-phase flow simulation with a much larger pore structure, $512 \times 256 \times 256$ grid size, which cannot be run on most single-processor machines. The parallel implementations with optimization schemes can perform flow simulations much more efficiently and can also solve larger problems.

Now we can go back to the question of the REV for two-phase flow simulation. We used a bigger pore structure, doubling the length of each side (Figure 7.18). The pore structures came from the same Fontainebleau sandstone as in previous chapters, and we have two different pore structures from different locations. The length scale of the bigger sample (L) is about $20a$. The relative permeability curves from each sample are shown at the bottom of Figure 7.18. Both non-wetting and wetting relative permeability curves are quite similar to each other, even though the two samples are from the different locations. This may mean that this sample size can represent the whole rock in terms of two-phase fluid flow. It is clear that the length scale of $L=10a$ is not big enough to represent the relative permeability of the whole rock.

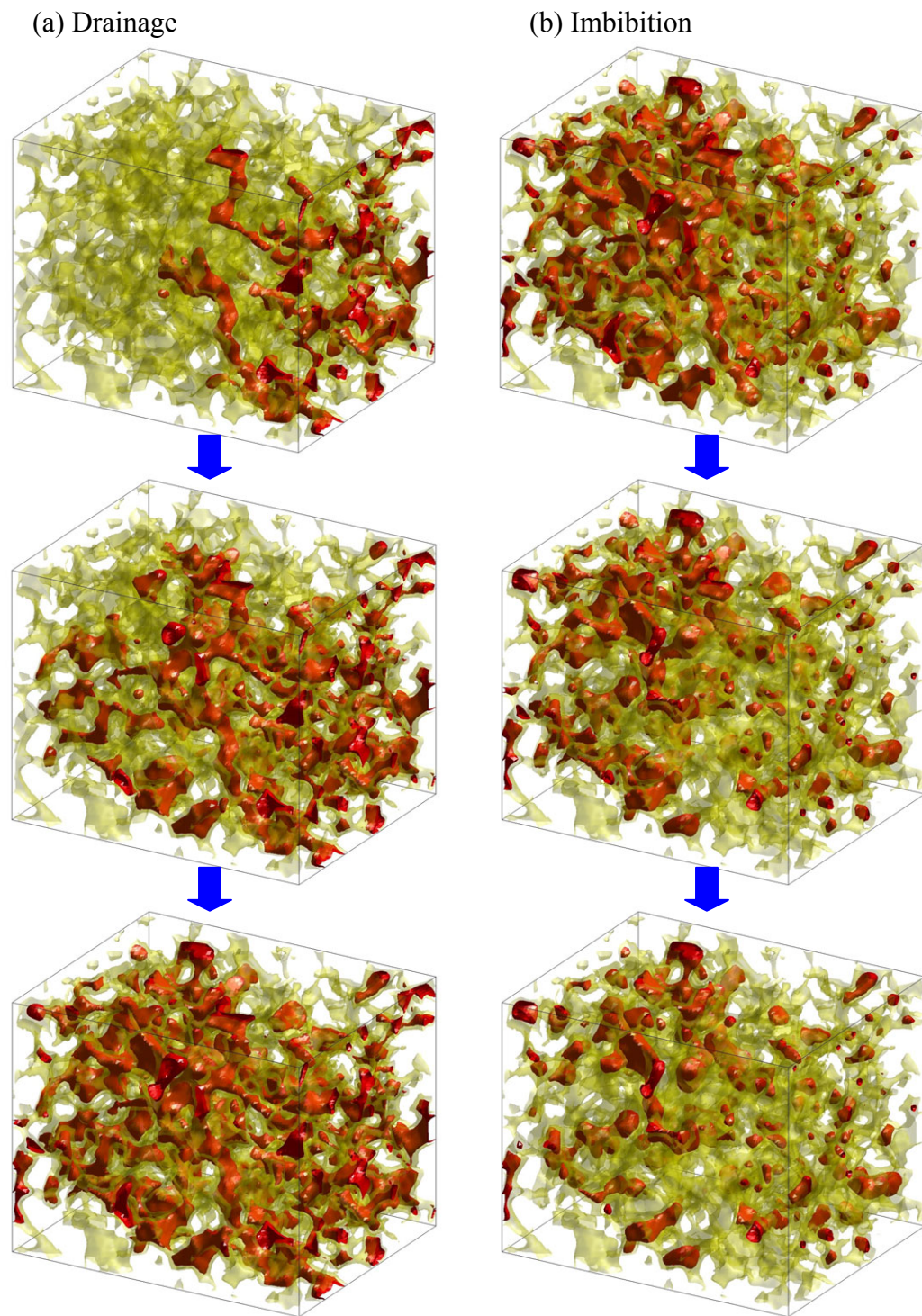


Figure 7.17: Two-phase flow in a Fontainebleau sandstone. Isosurface plots for (a) drainage and (b) imbibition. The red part is imbibed oil and green denotes pores filled with water. The sample size is about twice bigger than one that we used in Chapter 6.

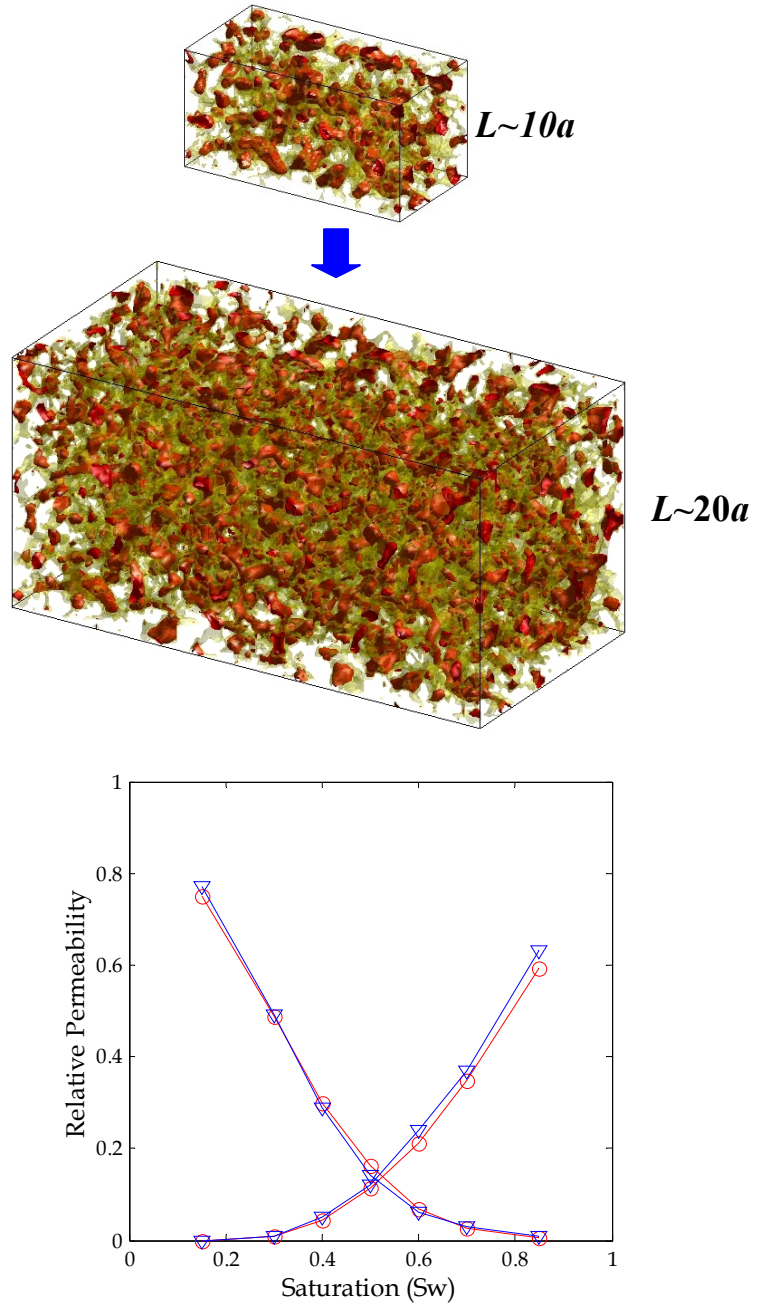


Figure 7.18: Bigger digital rock sample for testing the representative elementary volume for two-phase fluid flow (top). Relative permeability curves on two different bigger rock samples (bottom).

We preliminarily recommend the size of $L=20-30a$ for two-phase flow simulations. We also tried finer grid spacing ($dx=a/20$), but the difference in relative permeability between two fine ($dx=a/20$) and coarse ($dx=a/10$) grid spacing was almost negligible with our Fontainebleau sandstone samples. It seems that $dx=a/10$ is fine enough to resolve two interfaces (pore-grain interface and interface between two fluids) with our well-sorted sandstone samples. Care should be taken in choosing grid spacing with a wide range of pore size distribution, such as poorly sorted sandstones.

7.8 Effect of Optimizations

We have already seen that the implementations have very good performance both on shared-memory machines and a distributed memory machine. Now we will compare the performance of the optimized codes with un-optimized codes to see how the optimizations affect performance. Three cases were considered for the comparison of performance: sharing boundary partition, non-overlapping partitioning with a simple bisectional partitioning, and full optimizations – non-overlapping partition, weighted graph partitioning, and optimized ordering of communication. The PC cluster machine is used for the comparison of performance.

Figure 7.19 shows the results from single-phase flow simulations. The graph partitioning gives about 20-30% better performance than the bisectional partitioning (black arrow). The effect of non-overlapping partition is much bigger (gray arrow). The speedup by the shared boundary partition is less than 4 with 14 processors, while the speedup of the fully optimized code is very close to 14, which is the ideal linear speedup. The results of two-phase flow simulations show similar results to the single-phase flow case (Figure 7.20).

Figure 7.21 shows the effect of each optimization more clearly. The efficiency curves show that the sharing boundary scheme has less than 30% efficiency. In the right-hand side of the figure, the communication time by the shared boundary partition is more than 70% of the total calculation time, while the full-optimized code spends only 10% of its calculation time in communication. This effect will be much bigger as the number of processors increases.

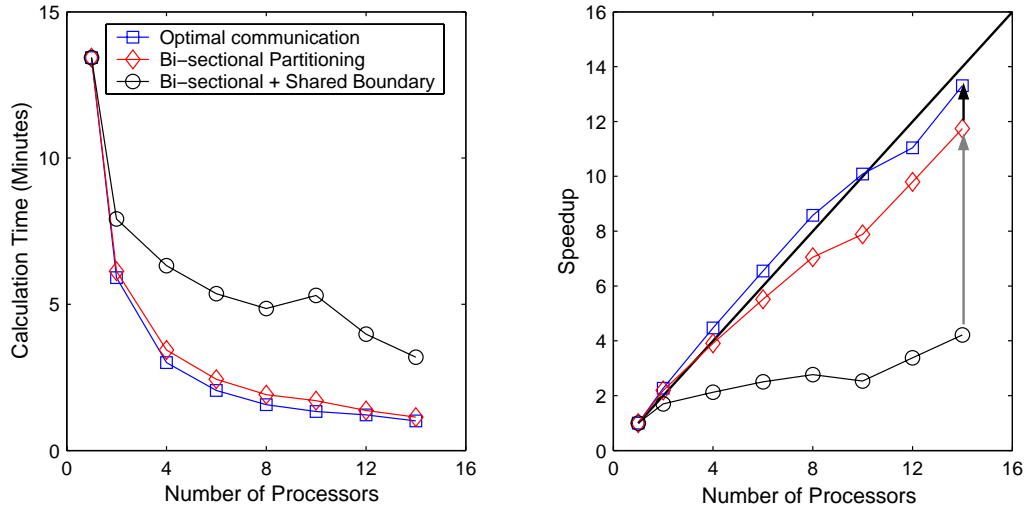


Figure 7.19: Performance results from single-phase flow simulations. Circles denote results from the shared boundary partition scheme, diamonds from a bisectonal partitioning with non-overlapping partition scheme, and squares from full optimizations. The arrows denotes the performance increase from each optimization.

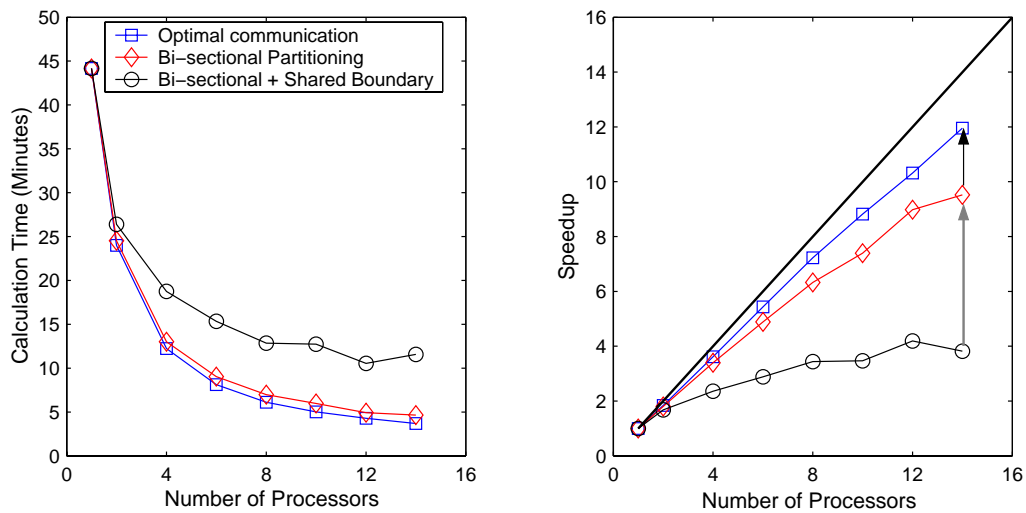


Figure 7.20: Performance results with two-phase flow simulations. Notations are the same as in Figure 7.19.

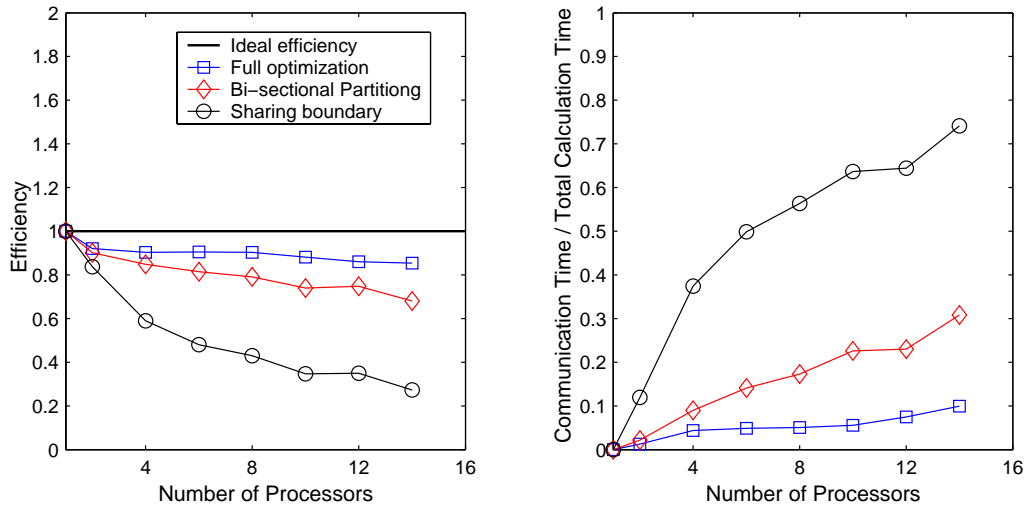


Figure 7.21: Comparison of efficiency (left) and communication time (right) with two-phase flow simulations. The communication time is normalized by total calculation time. With the shared boundary partition, the simulation will spend 70% of the total calculation time in communication.

7.9 Conclusions

We implemented parallel codes for the simulations of single-phase and two-phase fluid flow by the Lattice-Boltzmann method. In spite of the communication overhead, the performance of the parallel codes is good and very promising for further applications. The good performance comes from the customized optimizations for the Lattice-Boltzmann algorithm. The optimized code runs 12 times faster with 14 processors than its serial counterpart, while the generic parallel code is only 4 times faster. With this efficient parallel two-phase flow simulator, we investigated the representative elementary volume for two-phase flow simulation. We found that $L=10a$ is not big enough to represent two-phase fluid flow of the whole rock, although this size is big enough for representative single-phase flow of the whole rock. We preliminary suggest that the length scale of a digital rock should be greater than 20 mean pore size ($L=20a$) for two-phase flow simulation.

In terms of optimizations, our new development, the non-overlapping partition scheme has the greatest effect on the performance. We observe that the communication

cost is almost an order of magnitude smaller with this optimization. The Metis graph partitioning also has a big impact on the performance of the parallel codes. We observed that the graph partitioning method set partition boundaries at the pore throat, which can minimize the area of the boundary. The ordering of communication and communication balancing by a weighted graph partitioning has smaller effects than the first two. However, they non-negligible performance increase, and the effect would be bigger as the number of partitions increases. These efficient parallel implementations can help us efficiently simulate very time-consuming applications involving multiple two-phase flow simulations, such as diagenesis modeling with two-phase flow and capillary pressure curve simulation.

The parallel paradigm is now very widespread (Golub and Ortega, 1993; Culler et al., 1998; Dongarra et al., 1999; Fosdick et al., 1999). Both software and hardware is getting mature and easy to use. The implementation is not very complicated thanks to many good parallel libraries. As modern processors become faster, dependence of performance on cache also becomes more important. As we have seen the super-scalability from the better cache reuse, we can obtain good increase of performance even without well-designed parallel implementations. However, a little customization of parallel implementation may have a big impact on the scalability. Without several optimizations for minimizing communications, our parallel implementation would not scale well.

References

- Adler, P. and Brenner, H., 1988, Multiphase flow in porous media, *Ann. Rev. Fluid Mech.*, **20**, 35-59.
- Adler, P. M., Jacquin, C. G. and Quiblier, J. A., 1990, Flow in simulated porous media, *Int. J. Multiphase Flow*, **16**, 691-712.
- Agrawal, P., Agrawal, V. D., Bushnell, M. L. and Sienicki, J., 1994, Superlinear Speedup in a multiprocessing environment, *Proc. First Int'l Workshop on Parallel Processing*, 261-265.
- Aharonov, E., and Rothman, D. E., 1996, Growth of correlated pore-scale structures in sedimentary rocks: A dynamic model, *J. Geophys. Res.*, **101**, 2973-2987.
- Aharonov, E., Rothman, D. E. and Thompson, A. H., 1997, Transport properties and diagenesis in sedimentary rocks: The role of micro-scale geometry, *Geology*, **25**, 547-550.
- Amati, G., Succi, S. and Piva R., 1997, Massively parallel lattice-Boltzmann simulation of turbulent channel flow, *Intl. J. Modern Phys.*, **8**, 869-877.
- Archie, G. E., 1942, The electrical resistivity log as an aid in determining some reservoir characteristics, *J. Pet. Technol.*, **5**, 1-8.
- Bakke, S. and Øren, P. E., 1997, 3-D pore-scale modeling of sandstones and flow simulations in the pore networks, *SPEJ*, **2**, 136.
- Balay, S., Buschelman, K., Gropp, W. D., Kaushik, D., Knepley, M., McInnes, L. C., Smith, B. F. and Zhang, H., 2002, *PETSc Users Manual*, ANL-95/11, revision 2.1.3, Argonne National Laboratory.
- Berryman, J. G. and Blair, S. C., 1987, Kozeny-Carman relations and image processing methods for estimating Darcy's constant, *J. Appl. Phys.*, **60**, 1930-1938.
- Berryman, J. G., 1987, Relationship between specific surface area and spatial correlation functions for anisotropic porous media, *J. Math. Phys.*, **28**, 245-246.
- Blair, S. C., Berge, P. A., and Berryman, J. G., 1993, Two-point Correlation Functions to Characterize Microgeometry and Estimate Permeabilities of Synthetic and Natural Sandstones, Lawrence Livermore National Laboratory Report, Livermore.

- Bosl, W. J., 1999, *Computational Studies of Crustal Fluids and Earthquakes*, Ph.D. Dissertation, Stanford University, Stanford.
- Bosl, W. J., Dvorkin, J., and Nur, A., 1998, A numerical study of pore structure and permeability using a Lattice Boltzmann Simulation, *Geophys. Res. Lett.*, **25**, 1475–1478.
- Bryant, S. L., King, P. R. and Mellor, D. W., 1993, Network model evaluation of permeability and spatial correlation in a real random sphere packing, *Transport in Porous Media*, **11**, 53.
- Caers, J., and Xianlin, M., 2002, Modeling conditional distributions of facies from seismic using neural nets, *Mathematical Geology*, **34**, 2, 143-167.
- Cancelliere, A., Chang, C., Foti, E., Rothman, D. H. and Succi, S., The permeability of a random medium: comparison of simulation with theory, *Phys. Fluids A*, **2**, 2085-2088.
- Carman, P. C., 1961, *L'écoulement des Gaz à Travers les Milieux Poreux*, Bibliothèque des Sciences et Techniques Nucléaires, Presses Universitaires de France, Paris, 194pp.
- Chandra, R., Dagum, R., Kohr, D., Maydan, D., McDonald, J. and Menon, R., 2000, *Parallel Programming in OpenMP*, Morgan Kaufmann Publisher, 230pp.
- Chatzis, I. and Dullien, F. A. L., 1983, Dynamic immiscible displacement mechanisms in pore doublets: Theory versus experiment, *J. Colloid Interface Sci.*, **91**, 199-222.
- Chen, H., Chen, S. and Matthaeus, W. H., 1992a, Recovery of the Navier-Stokes equations using a lattice-gas Boltzmann method, *Phys. Rev. A*, **45**, 5339-5342.
- Chen, S., Wang, C., Shan, X. and Doolen, G. D., 1992b, Lattice Boltzmann computational fluid dynamics in three dimensions, *J. Stat. Phys.*, **68**, 379-400.
- Chopard, B. and Droz, M., 1998, *Cellular Automata Modeling of Physical Systems*, Cambridge University Press, Cambridge, 341pp.
- Craig, F. F., Jr., 1971, *The Reservoir Engineering Aspects of Water-Flooding*, Society of Petroleum Engineering of AIME, Monograph, **3**, New York, 134pp.
- Culler, D.E., Singh, J.P. and Gupta, A., 1998, *Parallel Computer Architecture: A Hardware/Software Approach*, Morgan Kaufmann Publishers, San Francisco, 1100pp.
- D'Humières, D., Lallemand, P. and Frisch, U., 1986, Lattice-gas models for 3D hydrodynamics, *Europhys. Lett.*, **2**, 291-297.
- Debye, P., Anderson, H. R., Jr., and Brumberger, H., 1957, Scattering by an inhomogeneous solid: II. The correlation function and its application, *J. Appl. Phys.*, **28**, 679–683.
- Deutsch, C. V., and Journel, A. G., 1998, *GSLIB: Geostatistical Software Library and User's Guide*, Oxford University Press, New York, 369pp.
- Dillard, L. A. and Blunt, M. J., 2000, Development of pore network simulation model to study nonaqueous phase liquid dissolution, *Water Resources Res.*, **36**, 439-454.

- Dongarra J., Duff, I., Sorenson D. and van der Vort, H., 1999, *Numerical Linear Algebra for High Performance Computers*, Society for Industrial & Applied Mathematics, Philadelphia, 342pp.
- Doolen, G. D. (Editor), 1990, *Lattice Gas Methods for Partial Differential Equations*, Addison-Wesley, Redwood City, 554pp.
- Dullien, F. A. L., 1992, *Porous Media: Fluid Transport and Pore Structure*, Academic Press, San Diego, 574pp.
- Fatt, I., 1956, The network model of porous media: I. Capillary pressure characteristics, *Pet. Tran. AIME*, **207**, 144-159.
- Finney, J., 1970, Random packings and structure of simple liquid: I. The geometry of random close packing, *Proc. Roy. Soc.*, 319A, 479.
- Fosdick, L. D., Jessup, E. R., Schauble, C. J. C. and Domik, G., 1996, *An Introduction to High-performance Scientific Computing*, MIT Press, Cambridge, MA, 760pp.
- Fredrich, J. T., Noble, D. R., O'Connor, R. M., and Lindquist, W. B., 1999, Development, Implementation, and Experimental Validation of the Lattice-Boltzmann Method for Modeling Three-dimensional Complex Flows, Sandia National Laboratory report SAND99-0369, Sandia National Laboratory, Sandia.
- Frisch, U., D'Humières, D., Hasslacher, B., Lallemand, P., Pomeau, Y. and Rivet, J.-P, 1987, Lattice gas dynamics in two and three dimensions, *Complex System*, **1**, 649-707.
- Frisch, U., Hasslacher, B. and Pomeau, Y., 1986, Lattice-gas automata for the Navier-Stokes equation, *Phys. Rev. Lett.*, **56**, 1505-1508.
- Geist, A., Beguelin, A. and Dongarra, J., 1994, PVM: Parallel Virtual Machine: A Users' Guide and Tutorial for Network Parallel Computing, MIT Press, 299pp.
- Golub, W. and Ortega, J. M., 1993, *Scientific Computing: An Introduction with Parallel Computing*, Academic Press, San Diego, CA, 442pp.
- Gropp, W., Lusk, E. and Skjellum, A., 1999, *Using MPI: Portable Parallel Programming with the Message Passing Interface*, 2nd Ed., MIT Press, Cambridge, 350pp.
- Guéguen, Y. and Palciauskas, V., 1994, *Introduction to the Physics of Rocks*, Princeton University Press, Princeton, 294pp.
- Gunstensen, A. K. and Rothman D. H., 1992, Microscopic modeling of immiscible fluids in three dimensional by a Lattice Boltzmann method, *Europhys. Lett.*, **18**, 157-161.
- Gunstensen, A. K. and Rothman D. H., 1993, Lattice-Boltzmann studies of immiscible two-phase flow through porous media, *J. Geophys. Res.*, **98**, 6431-6441.
- Gunstensen, A. K., Rothman, D. H., Zaleski, S. and Zanetti, G., 1991, Lattice Boltzmann model of immiscible fluids, *Phys. Rev. A*, **43**, 4320-4327.
- Halliday, I., Hammond, L. A., Care, C. M., Good, K. and Stevens, A., 2001, Lattice Boltzmann equation hydrodynamics, *Phys. Rev. E*, **64**, 011208.
- Hardy, J., de Pazzis, O. and Pomeau, Y., 1976, Molecular dynamics of a classical lattice gas: Transport properties and time correlation functions, *Phys. Rev. A*, **13**, 1949-1961.

- Hardy, J., Pomeau, Y. and de Pazzis, O., 1973, Time evolution of a two-dimensional model system: I. Invariant states and time correlation functions, *J. Math. Phys.*, **14**, 1746-1759.
- Heeger, D., and Bergen, J., 1995, Pyramid-based texture analysis/synthesis, *Proc. ACM SIGGRAPH*, 229-238.
- Helmbold, D. P. and McDowell, C. E., 1989, Modeling speedup(n) greater than n, *International Conference on Parallel Processing Proceedings*, **III**, 219-225.
- Higuera, F., Succi, S. and Benzi, R., 1989, Lattice gas dynamics with enhanced collisions, *Europhys. Lett.*, **9**, 345-349.
- Hilfer, R. and Manwart, C., 2001, Permeability and conductivity for reconstruction models of porous media, *Phys. Rev. E.*, **64**, 021304.
- Inamuro, T., Yoshino, M. and Ogino F., 1995, A non-slip boundary condition for lattice Boltzmann simulations, *Phys. Fluids*, **7**, 2928-2930.
- Kandhai, D., Vidal, D. J.-E., Hoekstra, A. G., Hoefsloot, H., Iedema, P. and Sloot, P. M. A., 1998, A comparison between lattice-Boltzmann and finite-element simulations of fluid flow in static mixer reactors, *Int. J. Modern Phys.*, **9**, 1123-1128.
- Kandhai, D., Vidal, D. J.-E., Hoekstra, A. G., Hoefsloot, H., Iedema, P. and Sloot, P. M. A., 1999, Lattice-Boltzmann and finite-element simulations of fluid flow in a SMRX static mixer reactor, *Int. J. Numer. Meth. Fluids*, **31**, 1019-1033.
- Karypis, G. and Kumar, V., 1998a, Metis: A Software Package for Partitioning Unstructured Graphs, Partitioning Meshes, and Computing Fill-Reducing Orderings of Sparse Matrices, Version 4.0, University of Minnesota, Minneapolis, 44pp.
- Karypis, G. and Kumar, V., 1998b, Multilevel k-way partitioning scheme for irregular graphs, *J. Parallel and Distributed Computing*, **48**, 96-129.
- Karypis, G. and Kumar, V., 1998c, A fast and high quality multilevel scheme for partitioning irregular graphs, *SIAM Journal on Scientific Computing*, **20**, 359-392.
- Keehm, Y., Mukerji, T., and Nur. A., 2001, Computational rock physics at the pore scale: Transport properties and diagenesis in realistic pore geometries, *The Leading Edge*, **20**, 180-183.
- Klimentos, T. and McCann, C., 1990, Relationships between compressional wave attenuation, porosity, clay content, and permeability of sandstones, *Geophysics*, **55**, 998-1014.
- Knackstedt, M. A. and Zhang, X., 1994, Direct evaluation of length scales and structural parameters associated with flow in porous media, *Phys. Rev. E*, **50**, 2134-2138.
- Koponen, A., Kandhai, D., Hellén, E., Alava, A., Hoekstra, A. G., Kataja, M., Niskanen, K., Sloot, P. and Timonen, K., 1998, Permeability of three-dimensional random fiber webs, *Phys. Rev. Lett.*, **80**, 716-719.
- Küntz, M. and Mareschal, J. C., 1997, Numerical estimation of the effective conductivity of heterogeneous media with a 2D cellular automata fluid, *Geophys. Res. Lett.*, **24**, 2865-2868.

- Küntz, M., Lavallée, P. and Mareschal, J. C., 1997, Determination of elastic properties of very heterogeneous media with cellular automata, *J. Geophys. Res.*, **102**, 7647-7658.
- Ladd, A. J. C., 1993, Short-time motion of colloidal particles: Numerical simulation via a fluctuating Lattice-Boltzmann equation, *Phys. Rev. Lett.*, **70**, 1339-1342.
- Ladd, A. J. C., 1994a, Numerical simulations of particulate suspensions via a discretized Boltzmann equation: Part 1. Theoretical foundation, *J. Fluid. Mech.*, **271**, 285-309.
- Ladd, A. J. C., 1994b, , Numerical simulations of particulate suspensions via a discretized Boltzmann equation: Part 2. Numerical results, *J. Fluid. Mech.*, **271**, 311-339.
- Langaas, K. and Yeomans, J. M., 2000, Lattice Boltzmann simulation of a binary fluid with different phase viscosities and its application to fingering in two dimensions, *Euro. Phys. J. B*, **15**, 133-141.
- Langtangen, H. P., 1999, Computational Partial Differential Equations: Numerical Methods and Diffpack Programming, Springer-Verlag, Germany, 682pp.
- Lenormand, R., Zarcone, C. and Sarr, A., 1983, Mechanisms of the displacement of one fluid by another in a network of capillary ducts, *J. Fluid Mech.*, **135**, 337-353.
- Liang, Z. R., Ioannidis, M. A. and Chatzis, I., 2000, Permeability and electrical conductivity of porous media from 3D stochastic replicas of the microstructure, *Chem. Eng. Sci.*, **55**, 5247-5262.
- Liang, Z. R., Philippi, P. C., Fernandes, C. P. and Magnani, F. S., 1999, Prediction of permeability from the skeleton of three-dimensional pore structure, *SPE Reservoir Eval. & Eng.*, **2**, 161-168.
- Maier, R. S., Bernard, R. S. and Grunau, D. W., 1996, Boundary conditions for the lattice Boltzmann method, *Phys. Fluids*, **8**, 1788-1801.
- Manwart, C., Torquato, S. and Hilfer, R., 2000, Stochastic reconstruction of sandstones, *Phys. Rev. E*, **62**, 893-899.
- Martys, N. S. and Chen, H., 1996, Simulation of multicomponent fluids in complex three-dimensional geometries by the lattice Boltzmann method, *Phys. Rev. E*, **53**, 743-750.
- Martys, N. S., Hagedorn, J.G., Goujon, D. and Devaney, J.E., 1999, Large scale simulations of single and multi-component flow in porous media, *Proc. SPIE*, **3772**, 205-213.
- Mavko, G., and Nur, A., 1997, The effect of a percolation threshold in the Kozeny-Carman relation, *Geophysics*, **62**, 1480-1482.
- Mavko, G., Mukerji, T. and Dvorkin, J., 1998, *The Rock Physics Handbook*, Cambridge University Press, New York, 329pp.
- MPI Forum, 1994, *MPI: A Message Passing Interface Standard*, University of Tennessee, Knoxville, 228pp.
- Nagy, K. L. and Lasaga, A. C., 1991, Dissolution and precipitation kinetics of Gibbsite at 80c and pH3, *Geochim. Cosmochim. Acta*, **56**, 3093.
- Nannelli, F. and Succi, S., 1992, The lattice Boltzmann equation on irregular lattices, *J. Stat. Phys.*, **68**, 401.

- Nie, X., Qian, Y. H., Doolen, G. D. and Chen, S., 1998, Lattice Boltzmann simulation of the two-dimensional Rayleigh-Taylor instability, *Phys. Rev. E*, **58**, 6861-6864.
- Olson, J. F. and Rothman D. H., 1997, Two-fluid flow in sedimentary rock: simulation, transport and complexity, *J. Fluid Mech.*, **341**, 343-370.
- Olson, J. F., 1995, Two-fluid Flow in Sedimentary Rock: Complexity, Transport, and Simulation, Ph.D. Dissertation, MIT, Cambridge.
- Orlandini, E., Swift, M. R. and Yeomans, J. M., 1995, A Lattice Boltzmann model of binary-fluid mixtures, *Europhys. Lett.*, **32**, 463-468.
- Pacheco, P.S., 1997, *Parallel Programming with MPI*, Morgan Kaufmann Publishers, San Francisco, 418pp.
- Parkinson, D., 1986, Parallel efficiency can be greater than unity, *Parallel Computing*, **3**, 261-262.
- Prasad, M. and Nur, A., *submitted to Geophysics*, Acoustic and depositional properties of fluvial sandstones.
- Qian, Y. H. and Deng, Y. F., 1997, A lattice BGK model for viscoelastic media, *Phys. Rev. Lett.*, **79**, 2742-2745.
- Rimstidt, J. D. and Barnes, H. L., 1980, The kinetics of silica-water reaction, *Geochim. Cosmochim. Acta*, **44**, 1683-1699.
- Roberts, A. P. and Torquato, S., 1999, Chord-distribution functions of three-dimensional random media: Approximate first-passage times of Gaussian processes, *Phys. Rev. E*, **59**, 4953-4963.
- Roberts, J. N. and Schwartz, L. M., 1985, Grain consolidation and electrical conductivity in porous media, *Phys. Rev. B*, **31**, 5990-5997.
- Rothman, D. H., 1988, Cellular-automaton fluids: a model for flow in porous media, *Geophysics*, **53**, 509-518.
- Rothman, D. H. and Keller, J. M., 1988, Immiscible cellular-automata fluids, *J. Stat. Phys.*, **52**, 1119-1127.
- Rothman, D. H., and Zaleski, S., 1997, *Lattice-Gas Cellular Automata: Simple Models for Complex Hydrodynamics*, Cambridge University Press, Cambridge, 297pp.
- Saeger R. B., Scriven, L. E. and Davis, H. T., 1995, Transport processes in periodic media, *J. Fluid Mech.*, **299**, 1-15.
- Sangani, A. S., and Acrivos, A., 1982, Slow flow past periodic arrays of cylinders with application to heat transfer, *Intl. J. Multiphase Flow*, **8**, 193-206.
- Sengupta, M., 2000, Integrating Rock Physics and Flow Simulation to Reduce Uncertainties in Seismic Reservoir Monitoring, Ph.D. Dissertation, Stanford University, Stanford.
- Shan, X. and Chen, H., 1994, Simulation of nonideal gases and liquid-gas phase transitions by the lattice Boltzmann equation, *Phys. Rev. E*, **49**, 2941-2948.

- Simoncelli, E. P., and Portilla, J., 1998, Texture characterization via joint statistics of wavelet coefficient magnitudes, IPCIP'98, 5th Int. Conf. on Image Processing, *IEEE Signal Processing Society*, **1**, 62-66.
- Skordos, P.A., 1995, Parallel simulation of subsonic fluid dynamics on a cluster of workstations, *Proc. IEEE Intl. Sym. High Performance Distributed Computing*, 6-16.
- Smith, B., Bjorstad, P. and Gropp, W., 1996, Domain Decomposition: Parallel Multilevel Methods for Elliptical Partial Differential Equations, Cambridge Univ. Press, Cambridge, 224pp.
- Spaid, M. A. A., and Phelan, F. R., Jr., 1997, Lattice Boltzmann methods for modeling microscale flow in fibrous porous media, *Phys. Fluids*, **9**, 2468–2474.
- Succi, S., 2001, *The Lattice Boltzmann Equation for Fluid Dynamics and Beyond*, Oxford University Press, New York, 304pp.
- Tiab, D. and Donaldson, E., 1996, Petrophysics: Theory and Practice of Measuring Reservoir Rock and Transport Properties, Gulf Publishing Company, Houston, 706pp.
- Torquato, S. and Stell, G., 1982, Microstructure of two-phase random media: I. The n-point probability functions, *J. Chem. Phys.*, **77**, 2071-2077.
- Torquato, S., 2001, Random Heterogeneous Materials: Microstructure and Macroscopic Properties, Springer-Verlag, New York, 712pp.
- Verberg, R. and Ladd, A. J. C., 1999, Simulation of low-Reynolds-number flow via a time-independent lattice-Boltzmann method, *Phys. Rev. E*, **60**, 3366-3373.
- von Neumann, J., 1966, *Theory of Self-Reproducing Automata* (Edited by A.W. Burks), Univ. of Illinois Press.
- Walsh, J. B., and Brace, W. F., 1984, The effect of pressure on porosity and transport properties of rock, *J. Geophys. Res.*, **89**, 9425–9431.
- Wong, P., Koplik, J. and Tomanic, J. P., 1984, Conductivity and permeability of rocks, *Phys. Rev. B*, **30**, 6606-6614.
- Xu, W. and Ruppel, C., 1999, Predicting the occurrence, distribution, and evolution of methane gas hydrate in porous marine sediments, *J. Geophys. Res.*, Vol. 104, 5081-5095.
- Yeong, C. L. Y., and Torquato, S., 1998a, Reconstructing random media, *Phys. Rev. E*, **57**, 495–506.
- Yeong, C. L. Y., and Torquato, S., 1998b, Reconstructing random media. II. Three-dimensional media from two-dimensional cuts, *Phys. Rev. E*, **58**, 224–233.
- Zhang, D., Zhang, R., Chen, S. and Soll, W. E., 2000, Pore scale study of flow in porous media: Scale dependency, REV, and statistical REV, *Geophys. Res. Lett.*, **27**, 1195-1198.