Michael C. Sukop
Daniel T. Thorne, Jr.

Lattice
Boltzmann
Modeling

An Introduction
for Geoscientists
and Engineers

Springer

M.C. Sukop
D.T. Thorne, Jr.
Lattice Boltzmann Modeling

Michael C. Sukop
Daniel T. Thorne, Jr.

Lattice Boltzmann Modeling

An Introduction for Geoscientists and Engineers

With 83 Figures

Springer

Dr. Michael C. Sukop
Florida International University
Department of Earth Sciences
University Park
Miami FL 33199
USA

Email: sukopm@fiu.edu

Dr. Daniel T. Thorne, Jr.
Georgetown College
Department of Mathematics,
Physics, and Computer Science
400 E College Street
Georgetown KY 40324
USA

Email: danny_thorne@georgetowncollege.edu

Preface

This book represents our effort to convey the understanding of Lattice Boltzmann Methods (LBM) that we have developed over the last 4 years. This understanding is incomplete; consultation of any of the other main texts and journal articles on the subject will reveal the depth of the topic and the level of mathematical and physical sophistication necessary for complete mastery. Nevertheless, we are able to accomplish remarkable things with LBM and we wish the same for our readers. This book is aimed at our peers who may be curious about the technique or simply wish to use it as a tool now and, like us, continue learning about it in greater depth in the future. Rather than the 'last word' on the techniques, we present first introductions. Criticism from those more knowledgeable on details of some of the methods is probably inevitable and deserved. We take responsibility for all errors in the text, but cannot be responsible for any results of applying the ideas or models we present.

MS and DT, Miami, Florida USA, July 22, 2005

Table of Contents

1 Introduction

Lattice Boltzmann models (LBM) have a remarkable ability to simulate single and multiphase fluids. A rich variety of behaviors, including unsteady flows, phase separation, evaporation, condensation, cavitation, solute and heat transport, buoyancy, and interactions with surfaces can readily be simulated. Persistent metastable states can be realized.

This book is intended primarily as a basic introduction that emphasizes intuition and the most simplistic conceptualization of processes. It largely avoids the more difficult mathematics and physics that underlie LB models. The model is viewed from a particle perspective where collisions, streaming, and particle-particle/particle-surface interactions constitute the entire conceptual framework. The beauty of these models resides in this simplicity. The particular multiphase models we develop here evolved primarily from the landmark papers of Shan and Chen (1993, 1994). These models are not perfect and their shortcomings have been explored in the literature. Nevertheless, they are exceptionally powerful and, because of their largely intuitive 'bottom up' nature, are particularly well suited to this kind of introduction.

Much of the material contained here can be extracted from the open literature and a number of pioneering books, including Succi (2001), Wolf-Gladrow (2000), and Rothman and Zaleski (1997). Chen and Doolen (1998) presented a review paper. However, beginners and those with more interest in model application than detailed mathematical foundations should find this book a powerful 'quick start' guide. We focus on 2-dimensional models, though extension to 3 dimensions is not particularly difficult. We work simultaneously with the fundamental equations and their computer implementation to illustrate the practical use of the equations. The reader should be aware of our approach to presenting code. Code is presented in small pieces throughout the text and is designed only to be human readable. It is pseudo-code, although it resembles C (as it is adapted from our actual implementation). Shortcuts in syntax (e.g., abbreviated variable indexing like `fij` for `f[j][i]`, and the abbreviation `foo+=bar` for `foo=foo+bar`) are employed generously to keep the

code snippets brief and line-lengths short as well as to optimize for readability. Our intention is to convey the nature of the implementation clearly so that the reader is well equipped to begin an implementation of their own and/or browse and modify/extend an existing implementation. Readers interested in the details are encouraged to examine the working code.

We provide code on the Internet (LB2D_Prime) and offer exercises that focus on confirming the code's ability to match analytical or observed results; this helps to instill confidence and point out deficiencies in the simple LBM models we introduce. We include pertinent references to guide readers to more specialized sources. The field is expanding and evolving rapidly however and many papers have not been mentioned. Figure 1 shows the exponential growth in the number of papers published since 1992.
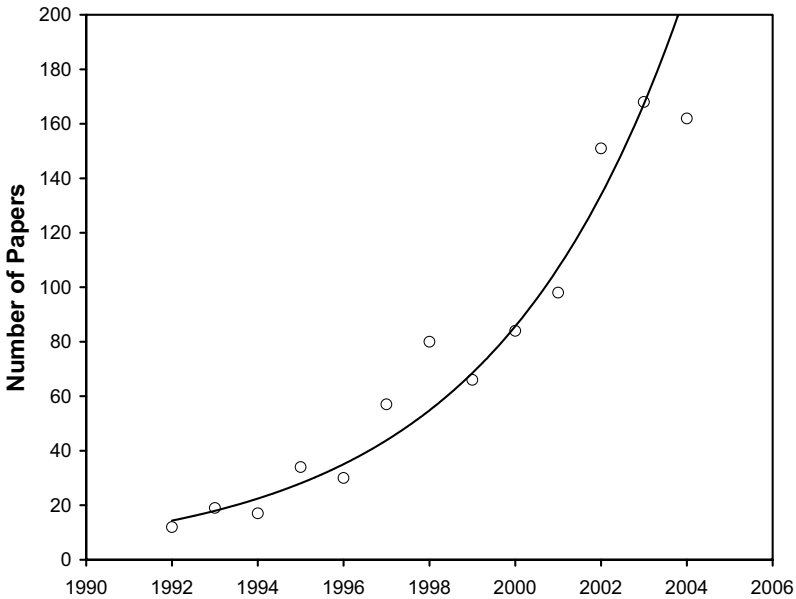
Figure 1. Growth in number of papers with 'lattice Boltzmann' as a 'topic' (search of article titles, abstracts, and keywords) in the Web of Science database 1992 - 2004. Solid line is fitted exponential growth curve. 2004 data may be incomplete.

It is also of interest to consider the nature of the published papers. Figure 2 gives the Web of Science Subject Categories for the papers published 1992-2004 that have lattice Boltzmann in their titles, abstracts, and/or keywords: most have appeared in physics and computer sciences. In our opinion, the distribution is likely to shift towards more applied areas (geosciences and engineering) as the power of these models is recognized.
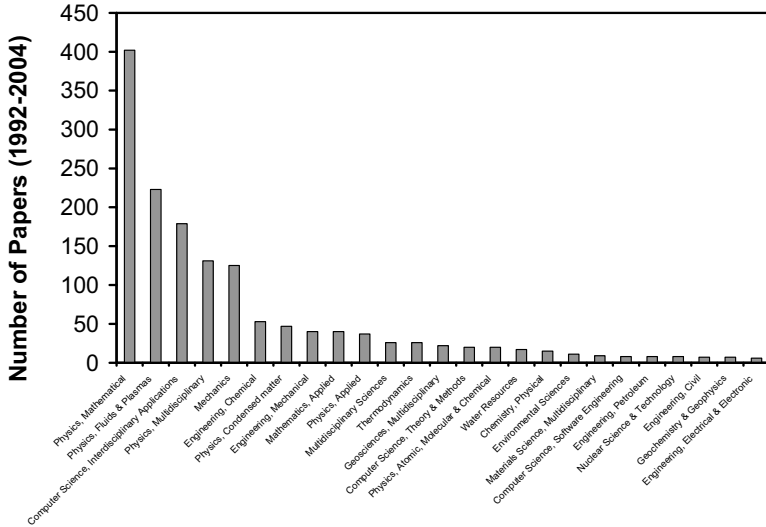
Figure 2. Web of Science Subject Categories for papers published 1992 - 2004 with lattice Boltzmann as a topic. Most papers so far have been published in Physics and Computer Science.

We begin our introduction to lattice Boltzmann models with a review of basic fluid mechanics concepts that are used later in the book. Cellular automata and lattice gases are covered briefly in the next chapter. Then we give a simplified introduction to Boltzmann gas concepts; it provides a basis for the 'stream and collide' mechanisms that are central to lattice gas models (the forebears of LBM) and LBM. Chapter 4 presents the core equations and computational aspects of LBM including a variety of boundary conditions. Chapter 5 introduces single component single phase LBM as the basis for extension to single component multiphase (SCMP LBM) in Chapter 6 and multi-component multiphase (MCMP LBM) models in Chapter 7. Solute transport is treated in Chapter 8 and Chapter 9 focuses on LBM for porous media at the macroscopic scale. Example simulations

are included at each stage of model extension to illustrate increasingly so-phisticated capabilities.

Another exciting use of LBM is for the simulation of shallow flows with the shallow water equations. We do not delve into this material as it has been covered in a recent book by Zhou (2003). We also do not touch on particle flows (e.g., Ladd 1993, 1994a,b; Ladd and Verberg 2001; Cates et al. 2004; Cook et al. 2004; Dupin et al. 2004).

Lattice Boltzmann models serve as exceptional numerical laboratories for a large number of physical and physicochemical processes. The ability to probe the simulations in detail for density and pressure gradients for ex-ample, has lead us to far deeper understanding of numerous phenomena than we would have achieved otherwise. While we expect quantitative re-sults from lattice Boltzmann methods, the learning value of playing with 'toy' models must not be underestimated.

Here we present elementary examples of a broad range of applications to illustrate the enormous potential of LBM. Assimilation of LBM into mainstream scientific computing in geosciences and engineering will re-quire extension of the models to larger applications that integrate databases and visualization as is characteristic of modern ground water models for example.

1.1 Review of Basic Fluid Mechanics

While some of our readers will need no introduction to or review of fluid mechanics, our experience indicates that for many it is worthwhile to re-view the most fundamental ideas on the behavior and quantitative treat-ment of fluids. This review is very minimalist in scope and focuses only on topics that are essential to basic understanding of LBM or will be the sub-ject of LBM simulations in subsequent chapters. More advanced physical chemistry needed for single component multiphase models and other top-ics are reviewed in later chapters.

1.1.1 Momentum

One fundamental concept that will be needed is that of momentum. The momentum $\mathbf{p}$ is defined as $\mathbf{p} = m\mathbf{u}$ with m the mass and $\mathbf{u}$ the velocity.

Conservation of mass and momentum are central to fluid mechanics and lattice Boltzmann models. Conservation of mass simply means that mass is not lost or created in the system under consideration. Conservation of momentum is well illustrated by the toy known as Newton's Cradle (Figure 3). Momentum attained by the moving ball just prior to its collision with the stationary balls is transmitted through the row of balls and converted back to motion of the ball on the opposite end of the row.
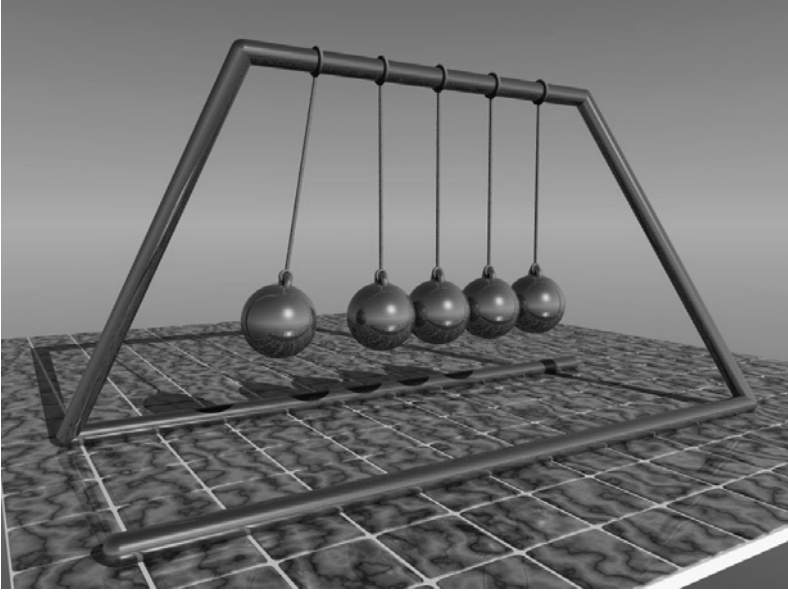
Figure 3. Newton's Cradle toy illustrates momentum conservation. (Rendering courtesy of Mark Hanford)

Not surprisingly, momentum is closely related to force. Newton's Second Law of Motion gives the force **F** as **F** = m**a**, where **a** is the acceleration. Acceleration is the time rate of change of velocity or d**u**/dt, so force can be written as

$$\mathbf{F} = m\frac{d\mathbf{u}}{dt} = \frac{d\mathbf{p}}{dt}. \tag{1}$$

1.1.2 Viscosity

The viscosity is a measure of the resistance to flow. Air has a very low viscosity relative to honey. Newton's Law of Friction relates the shear stress $\boldsymbol{\tau}$ to the velocity gradient in a Newtonian fluid:

$$\boldsymbol{\tau} = \mu \frac{d\mathbf{u}}{dx} \ . \tag{2}$$

The coefficient of proportionality is the dynamic viscosity μ. The kinematic viscosity is the dynamic viscosity divided by the fluid density μ/ρ. It is commonly denoted by ν and has dimensions of L^2T^{-1}. The kinematic viscosity can be thought of as a diffusion coefficient for momentum since

$$\boldsymbol{\tau} = \mu \frac{d\mathbf{u}}{dx} = \mu \frac{\rho}{\rho} \frac{d\mathbf{u}}{dx} = \nu \frac{d\mathbf{p}}{dx} \tag{3}$$

which is analogous to Fick's First Law of diffusion where a unit volume is implicit in the denominator of the rightmost term. This analogy is quite clear in the similarities between LBM simulations of fluids and solute transport that we will examine later.

1.1.3 Reynolds Number

The Reynolds Number (Re) is a non-dimensional number that reflects the balance between viscous and inertial forces. It is given by $Re = u\,L/\nu$ where u is the fluid velocity, L is a characteristic length, and ν is the kinematic viscosity. Low velocity, high viscosity, and confined fluid conditions lead to a low Re, the dominance of viscous forces, and laminar flow. If Re $<< 1$, the flow is known as Stokes or creeping flow (Figure 4). Such flow is traditionally thought to be common for liquids in many porous media due to small pore sizes.
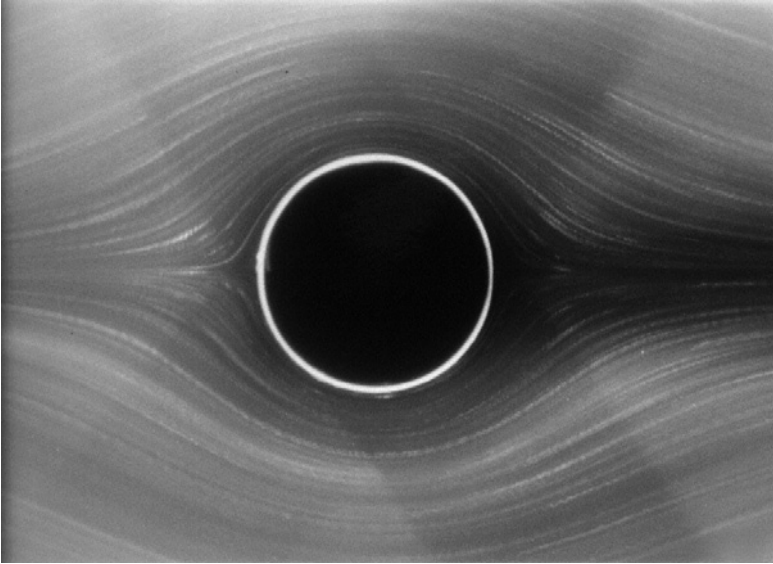
Figure 4. Stokes or creeping flow at low Reynolds number, Re ≈ 0.16 (Photograph by S. Taneda, with permission of the Society for Science on Form, Japan).

Higher velocities, larger length scales, or less viscous fluids lead to larger Reynolds numbers and the dominance of inertial forces over viscous forces. Under high Reynolds numbers the flow can become unstable (i.e., the onset of turbulence). Lattice Boltzmann models handle a range of Reynolds numbers very effectively and we will illustrate this later. The first departure from creeping flow is accompanied by a phenomenon known as flow separation and the formation of eddies as seen in Figure 5.
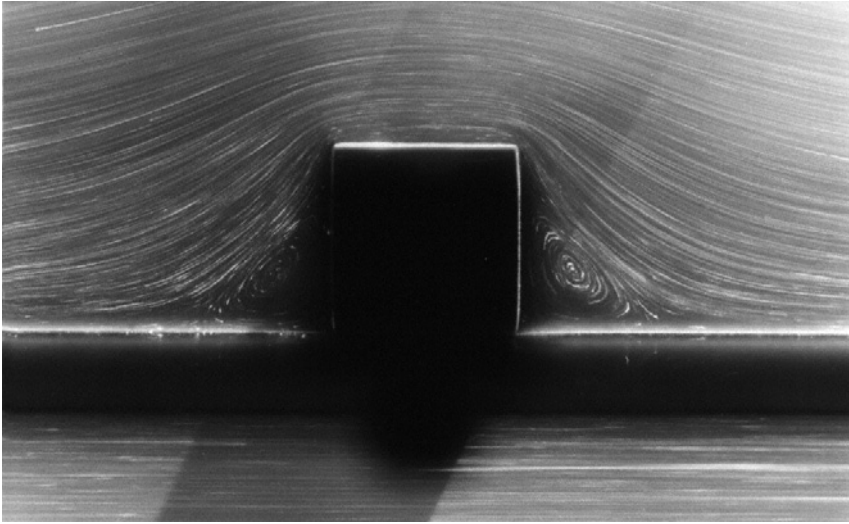
Figure 5. Separation at Re = 0.020 (Taneda (1979) with permission).

As the Reynolds number increases, unsteady and turbulent flows can ensue. We will investigate higher Reynolds number flows in Chapter 5.

1.1.4 Poiseuille Flow

An important and simple type of flow is that which occurs in a pipe or a slit between two parallel surfaces. These are called Poiseuille flows after the Frenchman Jean Léonard Marie Poiseuille (1797–1869) (Sutera and Skalak, 1993). In a slit or pipe, the velocities at the walls are 0 (no-slip boundaries) and the velocity reaches its maximum in the middle. As illustrated in Figure 6, the velocity profile in a slit of width $2a$ is parabolic and given by

$$u(x) = \frac{G^*}{2\mu}(a^2 - x^2) \tag{4}$$

where G^* can be the (linear) pressure gradient $(P_{in} - P_{out})/L$ or a gravitational pressure gradient (for example, in a vertical pipe $G^* = \rho g$). We will consider entry length effects later.
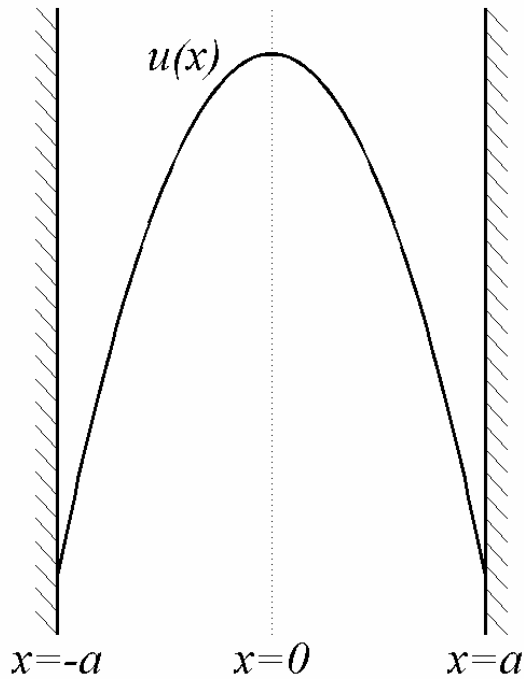
Figure 6. Poiseuille velocity profile.

It is useful to know that the average velocity in a slit is 2/3 of the maximum, or, since the maximum velocity is attained at $x = 0$,

$$u_{average} = \frac{2}{3}\frac{G^*}{2\mu}a^2 . \tag{5}$$

1.1.5 Laplace Law

There is a pressure difference between the inside and outside of bubbles and drops. The pressure is always higher on the inside of a bubble or drop (concave side) – just as in a balloon.
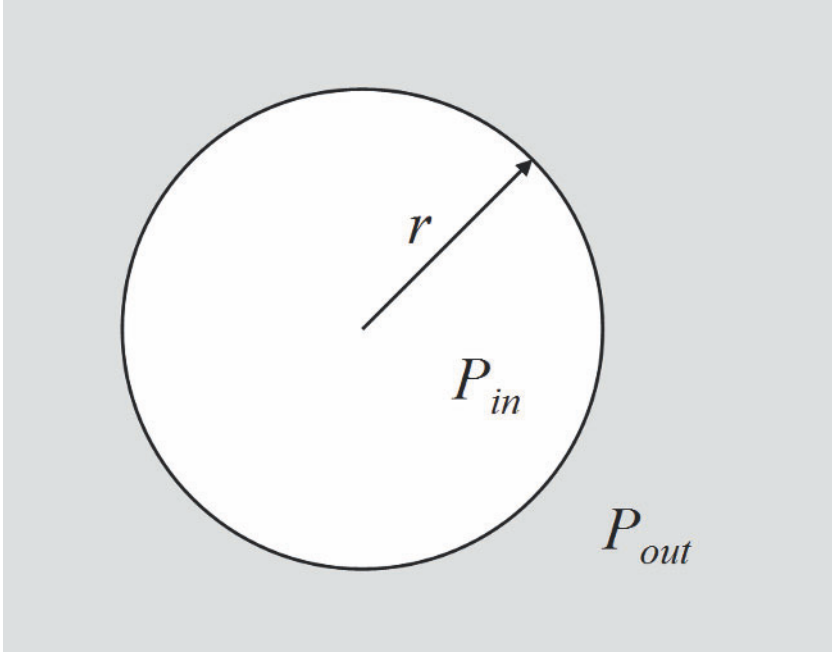
Figure 7. Definition diagram for Laplace Law. The difference in pressure inside and outside of a drop or bubble is inversely related to the radius r.

The pressure difference $\Delta P = |P_{outside} - P_{inside}|$ depends on the radius of curvature r and the surface tension σ for the fluid pair of interest. For two-dimensional drops and bubbles there is only one possible radius of curvature and

$$\Delta P = \frac{\sigma}{r}.$$

(6)

This Laplace Law indicates that ΔP is linear with respect to curvature $1/r$. We will use this later to estimate the surface tension in lattice Boltzmann simulations. The Laplace Law applies to both interfaces between a liquid and its own vapor (where σ is known as the surface tension) and between different fluids (like oil and water; where σ is referred to as the interfacial tension).

1.1.6 Young-Laplace Law

When solid surfaces are involved, in addition to the fluid1/fluid2 interface – where the interaction is given by the surface/interfacial tension – we have interfaces between both fluids and the surface. Often one of the fluids preferentially 'wets' the surface. This phenomenon is captured by the contact angle θ, and the Laplace relationship is modified as follows:

$$\Delta P = \frac{\sigma \cos\theta}{r}. \tag{7}$$

A zero contact angle means perfect wetting. In that case, $\cos\theta = 1$ and Eq. (7) reduces to Eq. (6). If the contact angle is 90°, $\cos\theta = 0$ and there is no pressure difference across the flat interface between the fluids.

2 Lattice Gas Models

Lattice gas cellular automaton models were the harbingers of LBM. We dedicate a chapter to basic cellular automata and lattice gases in part out of historical interest and in part because they represent a somewhat simpler and possibly more intuitive framework for learning gases on a lattice. Unfortunately, they require a perhaps less familiar Boolean mathematics (base 2 integers) on a less familiar triangular lattice. This material is not essential to applying LBM but it is interesting in its own right and might be helpful to developing a fuller understanding of LBM.

2.1 Cellular Automata

A cellular automaton (CA) is an algorithmic entity that occupies a position on a grid or lattice point in space and interacts with its identical neighbors. A cellular automaton generally examines its own state and the states of some number of its neighbors at any particular time step and then resets its own state for the next time step according to simple rules. Hence, the rules and the initial and boundary conditions imposed on the group of cellular automata uniquely determine their evolution in time.
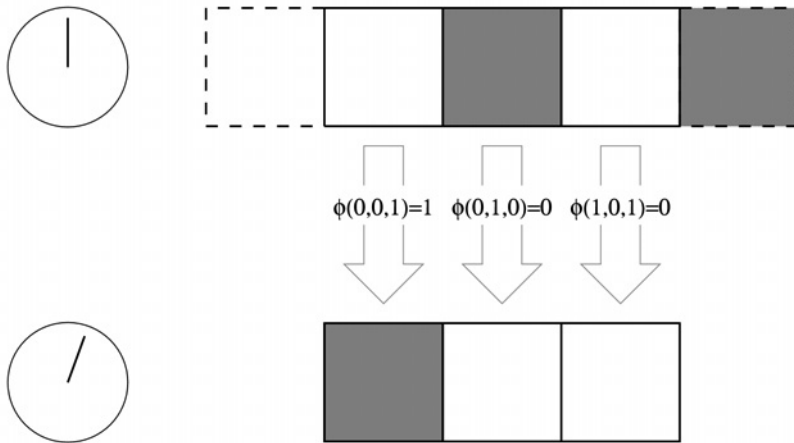
Figure 8. The basic components of a cellular automaton: a tiling of space, a clock that ticks out time, and a transition or update rule. The tiling of space in this illustration is a row of cells in a 1D space. A clock is represented at the left of the tilings. The update rule is denoted by the arrows from the state of the tiling from one time to the next. The update rule in this illustration maps the on/off state of a cell and its two neighbors at a given time tick on the clock to the on/off state of the cell at the next time tick on the clock.

The simplest cellular automata models are those that exist in one dimension on a line and consider only their own states and those of their two nearest, adjacent neighbors. If these automata have only two possible states (0 and 1, for example), then there are 256 possible rules for updating the central automaton. We can write the update rule symbolically as $a_i' = \phi(a_{i-1}, a_i, a_{i+1})$ where a_i' is the updated state, ϕ is one of 256 functions, and a_i, a_{i-1}, and a_{i+1} are the initial states of the automaton itself and its left and right neighbors respectively.

Many cellular automata can be computed using binary arithmetic. Wolfram (1986, 2002) presented a complete classification and analysis of the 265 rules for the 2-state, 2-neighbor automata. For each binary number from 00000000 to 11111111 (decimal 0 to 255) the update function ϕ is defined as follows.

Proceeding from right to left, each binary digit represents 2^0, 2^1, 2^2, ... , 2^7. Hence, the binary number 00000001 is $2^0 = 1$ while 00010010 is $2^4 + 2^1 = 18$. We take the exponents (4 and 1 in the case of binary 00010010)

as a variable n and solve for n_2, n_1, and n_0 in $n = 4n_2 + 2n_1 + n_0$. Then $\phi(n_2, n_1, n_0) = 0$ or 1 depending on the value of the binary digit. This is best illustrated by an example. The rightmost binary digit in 00010010 is 0. Its exponent in 2^0 is also 0. Hence, $n = 0$ and the only solution of $n = 4n_2 + 2n_1 + n_0$ is $n = 0 = 4(0) + 2(0) + (0)$ or $n_2 = n_1 = n_0 = 0$. Finally, $\phi(0, 0, 0) = 0$. Therefore, if the central automaton and its two nearest neighbors all have state 0 at a time step, the central automaton will be in the 0 state at the next time step.

The second binary digit is 1 and its exponent $n = 1 = 4(0) + 2(0) + (1)$. Thus, $\phi(0, 0, 1) = 1$. So, if the right neighbor has state 1 and the left neighbor and the central automaton are in the 0 state at a time step, the central automaton will update to state 1 at the subsequent time step.

The third binary digit is 0 and its exponent $n = 2 = 4(0) + 2(1) + (0)$. Thus, $\phi(0, 1, 0) = 0$ and, if the central automaton has state 1 and the right and left neighbors are in the 0 state at a time step, the central automaton will update to state 0 at the subsequent time step.

If we complete these computations for every combination of the 2 states for each of the 3 automata, we arrive at the following update table that defines ϕ:

$\phi(0, 0, 0) = 0$
$\phi(0, 0, 1) = 1$
$\phi(0, 1, 0) = 0$
$\phi(0, 1, 1) = 0$
$\phi(1, 0, 0) = 1$
$\phi(1, 0, 1) = 0$
$\phi(1, 1, 0) = 0$
$\phi(1, 1, 1) = 0$

In general, for n_s states and a neighborhood of n_n automata (including the one to be updated), the update table will require $n_s^{n_n}$ entries.

Despite the simplicity of this cellular automaton, it displays a complex evolution classed as chaotic and aperiodic by Wolfram (1986). To visualize its behavior, we can begin with a random initial condition of states, apply the update table, and show subsequent generations as a sequence of lines (Figure 9). This can easily be implemented on a spreadsheet. The ex-

ercises at the end of the chapter provide some hints. These and far more elaborate CA are discussed in Wolfram (2002).
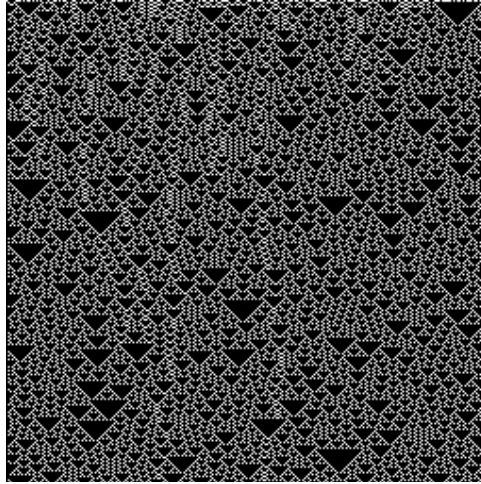
Figure 9. Evolution of a 1-dimensional, 2-state, 2-neighbor cellular automaton. Initial condition has 50% probability of sites in state 1 (black, top line). Subsequent generations are shown in each line progressing from top to bottom.

2.2 Two-Dimensional Lattice Gas Model of Fluid Flow

Lattice gas cellular automata were presented as a viable means of solving the Navier-Stokes equations of fluid motion in a landmark paper that appeared in 1986. Frish, Hasslacher, and Pomeau (Frish et al. 1986) provided the first lattice gas model that could properly simulate the 2-dimensional Navier-Stokes equations. It is commonly referred to as the 'FHP' model after these authors. This model is constructed on a equilateral triangular lattice that provides an isotropic solution. Lattice points are separated by 1 lattice unit (lu) and all particles have only one speed: 1 lu/time step ($lu\ ts^{-1}$). At each lattice point $\mathbf{x}$, there may be up to 6 particles – one for each of the possible velocities defined by the particle speed and one of the six possible directions: $\mathbf{e}_a = (\cos \pi a/3, \sin \pi a/3)$ where $a = 1, 2, …, 6$, and $\mathbf{e}_a$ is the velocity vector pointing from the origin (0,0) to the Cartesian coordinate $(\cos \pi a/3, \sin \pi a/3)$. A string of Boolean variables $\mathbf{n} = (n_1, n_2, …, n_6)$ contains the states ($n_a = 0$ or 1) indicating the presence (1) or absence (0) of

particles moving from a lattice gas site at **x** to a neighboring site at $\mathbf{x} + \mathbf{e}_a$ (Rothman and Zaleski 1997).
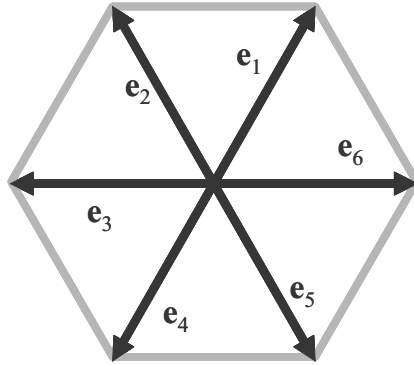
Figure 10. FHP unit velocity vectors.

The evolution of a lattice gas model proceeds in two steps that take place during each time step. The first step is a propagation, 'hopping' or 'streaming' step in which the particles move to new sites according to their previous positions and their velocities. Next, the particles collide and scatter according to collision rules.

2.2.1 Collision Rules

There are several possible types of collisions on the hexagonal lattice. Only two types are considered in the simplest FHP model; two-body collisions involve 2 particles while three-body collisions involve 3. Two critical features of the lattice gas that allow it to simulate the Navier-Stokes equations are mass conservation and momentum conservation. Thus, it is essential that the microscopic-scale collisions honor mass and momentum conservation. In the lattice gas, all particles have the same mass and speed so that momentum conservation reduces to conservation of the vector sum of the velocities. Head-on collisions between 2 particles (or 3 particles approaching one another from $\pi/3 = 120°$ separation) have no net momentum. Hence, the results of these collisions must also have zero net momentum.

Figure 11 illustrates the zero net momentum, 2- and 3-particle collisions respectively. The vectors shown in these figures represent velocity vectors

attributable to particles at the center of each hexagon just prior to and just after the collision step.

Pre-collision   Possible post-collision

Pre-collision   Post-collision
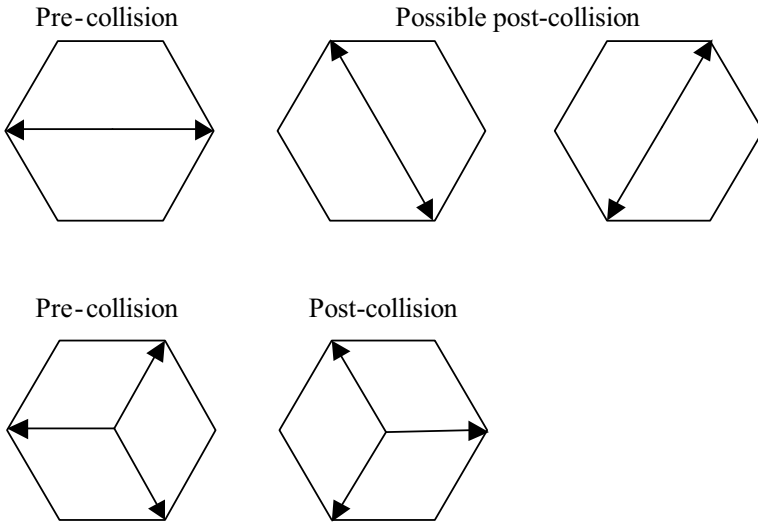
Figure 11. Zero net momentum, head-on, 2- and 3-particle collisions.

2.2.2 Implementation

In practice, the directions are coded to a variable A though F as shown in Figure 12.
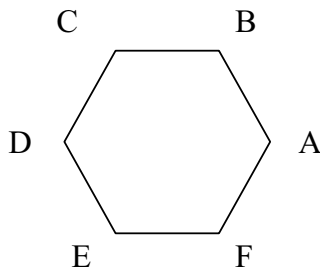
Figure 12. FHP Variable definitions

The variables correspond to bit strings as shown in Table 1. Note that only the first six bits (through a value of 32) are needed to describe the presence or absence of particles in all six directions. The additional bits play important roles however. The seventh bit signals the presence of a solid, while the eight bit is randomly 0 or 1 simply to decide between alternative post-collision states like those at the top of Figure 11.

Table 1. FHP model variables and their values.

	Bit Value							
	128	64	32	16	8	4	2	1
A	0	0	0	0	0	0	0	1
B	0	0	0	0	0	0	1	0
C	0	0	0	0	0	1	0	0
D	0	0	0	0	1	0	0	0
E	0	0	0	1	0	0	0	0
F	0	0	1	0	0	0	0	0
S	0	1	0	0	0	0	0	0
R	1	0	0	0	0	0	0	0

Now consider the collision illustrated in Figure 13. There is no change in the pre- and post-collision velocities because no other velocity configuration that is possible on the hexagonal lattice conserves the momentum present prior to the collision. The same is true for 2 particles that collide at 120º and all three-particle collisions with the exception of that shown in Figure 11.

Pre-collision   Post-collision
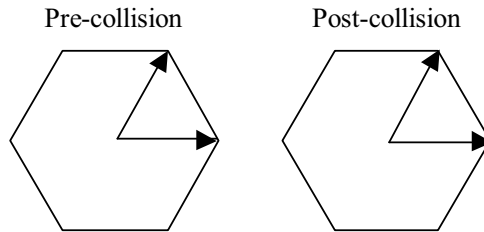
Figure 13. Pre- and post-collision velocities for 2-particle collision with initial velocities separated by 60°. No configurations other than the original conserve mass and momentum; the same is true for all 5- and 6-particle collisions.

Inclusion of these 2- and 3-particle collisions completes the simplest lattice gas model. More complex models that include 4-particle collisions and particles with zero velocity can be devised (Rothman and Zaleski, 1997). Five- and 6-particle collisions cannot be replaced with any other velocity configuration if momentum is to be conserved.

With these considerations, we are in a position to construct a look-up table (Figure 14) that reads the current configuration of particle velocities, solid presence, and random bit, and returns the new configuration. We begin by filling the table with the trivial information 'new configuration = old configuration' for each of the 256 possible configurations, because in fact we have decided many configurations will not change. For the first 64 configurations (00000000 through 00111111) the seventh bit (which signifies a solid surface) is 0 and no solid is present. The same is true for configurations 128 through 191 (10000000 through 10111111). We continue to assume (for the moment) that no changes to these configurations will be needed.

	Bit Value							
	128 R	64 S	32 F	16 E	8 D	4 C	2 B	1 A
0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	1
2	0	0	0	0	0	0	1	0
3	0	0	0	0	0	0	1	1
…								
255	1	1	1	1	1	1	1	1

Figure 14. Collision "look up" table. 256 entries. Start with all unchanged. 'new configuration' = 'old configuration'

In contrast, for configurations 64 through 127 and 192 through 255 (01000000 through 01111111 and 11000000 through 11111111), there are solids present. The particles cannot pass through the solids. One of the simplest boundary conditions to apply at the surface of the solid is the 'bounce back' condition. This consists of sending the particle directly back where it came from. So, for instance (referring to Figure 12), an A particle becomes a D particle, B becomes E, C becomes F, and so on. These are the first modifications we make to the table (see Figure 15).

	In-State Bit Value							
	128 R	64 S	32 F	16 E	8 D	4 C	2 B	1 A
64	0	1	0	0	0	0	0	0
65	0	1	0	0	0	0	0	1
66	0	1	0	0	0	0	1	0
…								
127	0	1	1	1	1	1	1	1

	Out-State Bit Value							
	128 R	64 S	32 F	16 E	8 D	4 C	2 B	1 A
64	0	1	0	0	0	0	0	0
72	0	1	0	0	1	0	0	0
80	0	1	0	1	0	0	0	0
…								
127	0	1	1	1	1	1	1	1

Figure 15. In-State/Out-State bit values with solids present (S = 1).

Next we take into account the collisions considered in Figure 11. There are three obvious possibilities for the two-particle, head on collision: AD, BE, and CF. Actually however, because of the eight bit and because there are two possible outcomes that must be equally likely, there are really two sets of these; one set has the eight bit = 0 and the other has it equal to 1. Now, for example, say we have the configuration AD with bit 8 = 0 (00001001 or 9). The new table configuration is BE (00010010 or 18). If the eight bit is 1, AD is 10001001 (= 137). Now the other configuration – CF (10100100 = 164) is selected. Bit 8 remains unchanged.

The randomness introduced by this procedure is essential to the ability of the lattice gas to simulate fluids.

	In-State Bit Value							
	128 R	64 S	32 F	16 E	8 D	4 C	2 B	1 A
9 (AD)	0	0	0	0	1	0	0	1
137 (AD)	1	0	0	0	1	0	0	1

	Out-State Bit Value							
	128 R	64 S	32 F	16 E	8 D	4 C	2 B	1 A
18 (BE)	0	0	0	1	0	0	1	0
164 (CF)	1	0	1	0	0	1	0	0

Figure 16. In-State/Out-State Bit values for two-particle head-on collisions.

Changing all of the zero-momentum, 3-particle collisions (ACE, BDF, and their bit 8 = 1 counterparts (Figure 17)), completes the definition of the model. Each of the 256 unique combinations of the 8 bits is accounted for.

	In-State Bit Value							
	128 R	64 S	32 F	16 E	8 D	4 C	2 B	1 A
21 (ACE)	0	0	0	1	0	1	0	1
42 (BDF)	0	0	1	0	1	0	1	0
149 (ACE)	1	0	0	1	0	1	0	1
170 (BDF)	1	0	1	0	1	0	1	0

	Out-State Bit Value							
	128 R	64 S	32 F	16 E	8 D	4 C	2 B	1 A
42 (BDF)	0	0	1	0	1	0	1	0
21 (ACE)	0	0	0	1	0	1	0	1
170 (BDF)	1	0	1	0	1	0	1	0
149 (ACE)	1	0	0	1	0	1	0	1

Figure 17. In-State/Out-State Bit Values for three-particle head-on collisions.

Because there are relatively few in-state => out-state changes in our table, it is easy to implement this in computer code. The following is from code provided by Rothman and Zaleski (1997). A 'table' array is indexed with the possible in-states from 0 to 255 and the out-states are the values contained in the array.

```
table[A + D]       = B + E;
table[B + E]       = C + F;
table[C + F]       = A + D;
table[A + D + EPS] = C + F;
table[B + E + EPS] = A + D;
table[C + F + EPS] = B + E;

table[A + C + E]       = B + D + F ;
table[B + D + F]       = A + C + E;
table[A + C + E + EPS] = B + D + F + EPS;
table[B + D + F + EPS] = A + C + E + EPS;
```

Figure 18. Code fragment from Lgapack Version 0.98 for the simulation of flow with lattice gas automata. Copyright (C) 1997 D.H. Rothman and S. Zaleski. This code is freely available under GNU General Public License from ftp://ftp.jussieu.fr/jussieu/labos/lmm/Lgapack/. EPS refers to the random bit R.

Three additional details are needed to implement a FHP model. First, the need to select randomly among the two possible configurations for the head-on two-particle collisions interjects a great deal of 'noise' into the simulations. In fact, this 'noise' is crucial to the ability to simulate hydrodynamics with a lattice gas and may even be viewed as an advantage in certain circumstances. But to obtain the smooth flow fields we expect in fluids at macroscopic scales under many conditions from a lattice gas simulation, a significant amount of temporal and/or spatial averaging is needed. Next, the equilateral triangular lattice is not particularly amenable

to computer computation (instead of the four or eight neighbors of a Cartesian point, there are six) and a remapping scheme is needed (Figure 19). The vertical separation of node points is $\sqrt{3}/2$.
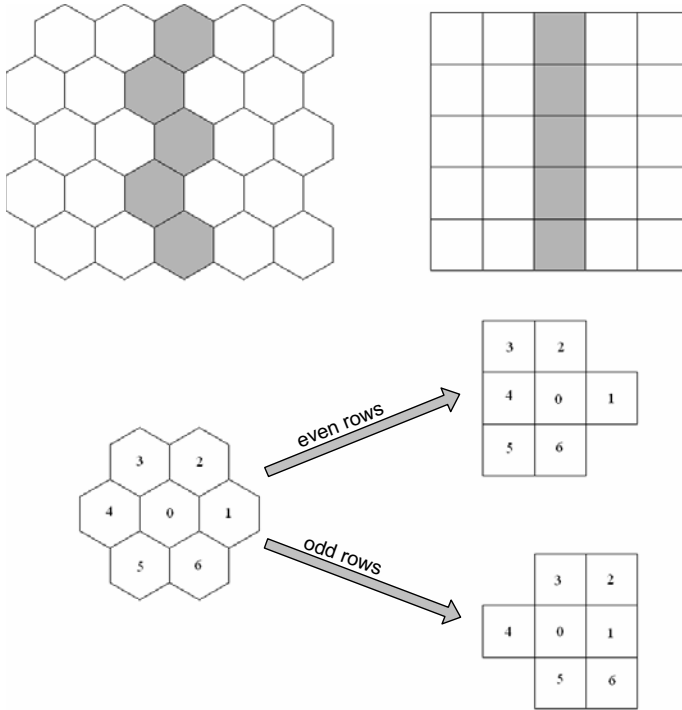
Figure 19. One possible scheme for remapping the equilateral triangular grid onto a more 'computer-friendly' system. Alternate rows are shifted left or right in the scheme.

Finally, a driving force is needed. The simplest approach is to 'flip' the momentum of some randomly selected fraction of the particles; for example, a fraction of the D direction particles become A direction. This corresponds to the addition of A-direction momentum to the system.

2.2.3 Example

With these pieces in place, we can compute reasonable hydrodynamics. Figure 20 shows the results of an early lattice gas simulation (see Rothman (1988) for a similar simulation). As noted by others (Rothman and Zaleski 1997; Succi 2001; Wolfram 2002), the model is remarkable for its great simplicity. That fluid flows can be computed on the basis of only 6 particle momenta and a handful of collisions attests to an amazing underlying simplicity in nature.
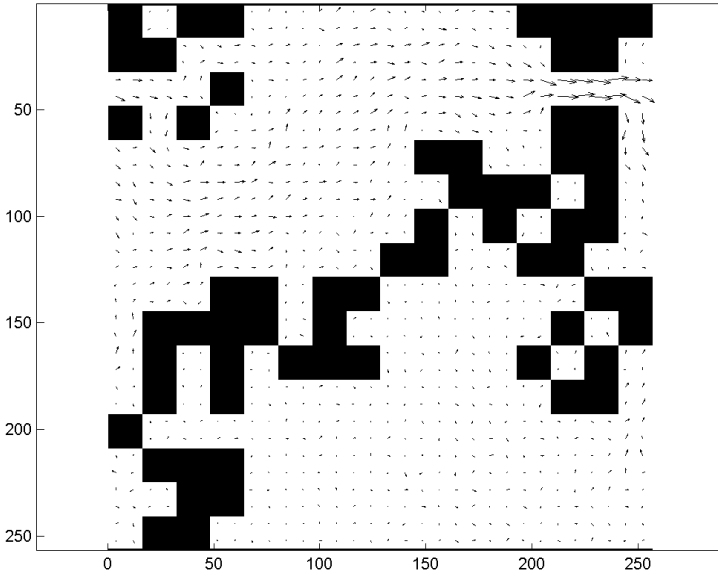
Figure 20. Crude lattice gas simulation of flow in a periodic 2-D network.

2.3 Exercises

1. Implement Wolfram's (1986, 2002) Rule 18 CA on a spreadsheet. In Microsoft Excel®, the 2nd row, 2nd column formula is:

```
=IF(OR(AND((A1=1),(B1=0),(C1=0)),AND((A1=0),(B1=0),(C1=1))),1,0).
```

It needs to be copied throughout the domain except on row 1, which is the initial condition. Try different initial conditions by seeding the first line with different patterns of 0s and 1s.

2. Download LGAPACK from the ftp site ftp://ftp.jussieu.fr/jussieu/labos/lmm/Lgapack/. Read the README file and study the code fhp6_simp4.c and the associated header (.h) files. Delete the original fhp6_simp.c and rename fhp6_simp4.c to fhp6_simp.c, then type 'make clean' and 'make' to compile the code. Modify the parameters.h file so that FORCING_RATE = 0.0001 and TPRINT, TMAX, and TAVG all equal 100000. Run the code and use the following MATLAB® code to visualize the results:

```
clear('all')
load x_mom
load y_mom
load mass
x_vel=x_mom./(2*mass)
y_vel=sqrt(3)*y_mom./(2*mass)
quiver(x_vel',y_vel')
axis equal
```

You should obtain a Poiseuille velocity profile. Reduce TPRINT, TMAX, and TAVG to 1000. What is the effect on the results and why?

3. Find the maximum x velocity in the simulation and use the velocity, FORCING_RATE (g), and channel width in Eq. (4) to estimate the kinematic viscosity of the simulated fluid. (Note that the densities in G^* and the bulk viscosity cancel, leaving g and the kinematic viscosity.) Compare your estimate to the theoretical density-dependent kinematic viscosity of the lattice gas model value given by (Rothman and Zaleski, 1997)

$$ v = \frac{1}{12f(1-f)^3} - \frac{1}{8}, \tag{8} $$

where f is the 'reduced density' (average number of particles per lattice link) as given in the parameters.h file. Can you improve your estimates by changing the simulation?

3 Basic Boltzmann Gas Concepts

Ludwig Boltzmann was born in Austria in 1844. He took his own life in 1906 probably at least in part as the result of despondency over the difficulty of having his ideas accepted. Shortly after his death, his notions of gases and the atomic theory of matter in general were broadly embraced by the scientific community and continue to play important roles (Harris 1971). His "Lectures on Gas Theory" (Boltzmann 1964/1995) are useful reading.

The basic idea of Boltzmann's work is that a gas is composed of interacting particles that can be described by classical mechanics, and, because there are so many particles, a statistical treatment is necessary and appropriate. The mechanics can be extremely simple and encapsulated by just the notions of streaming in space and billiard-like collision interactions. As we will see, lattice Boltzmann models simplify even further and yet, like lattice gas models, still reproduce the behavior of real fluids.

We will introduce some basic concepts from the kinetic theory of gases and statistical mechanics and 'derive' a simplified form of the Boltzmann equation. Readers interested in complete treatments are referred to the classic texts "Molecular theory of gases and liquids" by Hirchfelder et al. (1954/1965) and Chapman and Cowling's "The mathematical theory of non-uniform gases" (1990). Books by Kittel (1958/2004) and Mattis (2003) may also prove helpful.

3.1 Kinetic Theory

Consider a dilute gas consisting of hard spherical particles moving at great velocity ($\sim$300 ms^{-1}). We limit their interaction to elastic collisions. Hypothetically, it would be possible to know the position vector ($\mathbf{x}$) and momen-

tum (**p**) of each individual particle at some instant in time. Such information would give the exact dynamical state of the system which, together with classical mechanics, would allow exact prediction of all future states.

We could describe the system by a distribution function $f^{(N)}(\mathbf{x}^N, \mathbf{p}^N, t)$ where N is the number of particles. Here the distribution is thought of as residing in a 'phase space', which is a space in which the coordinates are given by the position and momentum vectors and the time. Changes in $f^{(N)}(\mathbf{x}^N, \mathbf{p}^N, t)$ with time are given by the Liouville equation ($6N$ variables). However, this level of description is not possible for real gases, where $\sim 10^{23}$ (a mole of) particles are involved in just 20 liters of gas at atmospheric temperature and pressure. Fortunately we are usually interested only in low order distribution functions ($N = 1, 2$).

3.2 First Order Distribution Function

Statistical Mechanics offers a statistical approach in which we represent a system by an ensemble of many copies. The distribution $f^{(1)}(\mathbf{x}, \mathbf{p}, t)$ gives the probability of finding a particular molecule with a given position and momentum; the positions and momenta of the remaining N-1 molecules can remain unspecified because no experiment can distinguish between molecules, so the choice of which molecule does not matter. This is the 'Single particle' distribution function. $f^{(1)}$ is adequate for describing all gas properties that do not depend on relative positions of molecules (dilute gas with long mean free path).

The probable number of molecules with position coordinates in the range $\mathbf{x} \pm d\mathbf{x}$ and momentum coordinates $\mathbf{p} \pm d\mathbf{p}$ is given by $f^{(1)}(\mathbf{x}, \mathbf{p}, t)d\mathbf{x}d\mathbf{p}$. Say we introduce an external force **F** that is small relative to intermolecular forces. If there are no collisions, then at time $t + dt$, the new positions of molecules starting at **x** are $\mathbf{x} + (\mathbf{p}/m)dt = \mathbf{x} + (d\mathbf{x}/dt)dt = \mathbf{x} + d\mathbf{x}$ and the new momenta are $\mathbf{p} = \mathbf{p} + \mathbf{F}dt = \mathbf{p} + (d\mathbf{p}/dt)dt = \mathbf{p} + d\mathbf{p}$.

Thus, when the positions and momenta are known at a particular time t, incrementing them allows us to determine $f^{(1)}$ at a future time $t + dt$:

$$f^{(1)}(\mathbf{x} + d\mathbf{x}, \mathbf{p} + d\mathbf{p}, t + dt)d\mathbf{x}d\mathbf{p} = f^{(1)}(\mathbf{x}, \mathbf{p}, t)d\mathbf{x}d\mathbf{p} . \qquad (9)$$

This is the streaming process.

There are however collisions that result in some phase points starting at $(\mathbf{x}, \mathbf{p})$ not arriving at $(\mathbf{x} + \mathbf{p}/m \, dt, \mathbf{p} + \mathbf{F} \, dt) = (\mathbf{x} + d\mathbf{x}, \mathbf{p} + d\mathbf{p})$ and some not starting at $(\mathbf{x}, \mathbf{p})$ arriving there too. We set $\Gamma^{(-)} d\mathbf{x} d\mathbf{p} dt$ equal to the number of molecules that do not arrive in the expected portion of phase space due to collisions during time dt. Similarly, we set $\Gamma^{(+)} d\mathbf{x} d\mathbf{p} dt$ equal to the number of molecules that start somewhere other than $(\mathbf{x}, \mathbf{p})$ and arrive in that portion of phase space due to collisions during time dt. If we start with Eq. (9) and add the changes in $f^{(1)}$ due to these collisions we obtain

$$f^{(1)}(\mathbf{x} + d\mathbf{x}, \mathbf{p} + d\mathbf{p}, t + dt) d\mathbf{x} d\mathbf{p}$$
$$= f^{(1)}(\mathbf{x}, \mathbf{p}, t) d\mathbf{x} d\mathbf{p} + \left[\Gamma^{(+)} - \Gamma^{(-)} \right] d\mathbf{x} d\mathbf{p} dt. \tag{10}$$

The first order terms of a Taylor series expansion of the LHS of Eq. (10),

$$f^{(1)}(\mathbf{x} + d\mathbf{x}, \mathbf{p} + d\mathbf{p}, t + dt) =$$
$$f^{(1)}(\mathbf{x}, \mathbf{p}, t) + d\mathbf{x} \cdot \nabla_{\mathbf{x}} f^{(1)} + d\mathbf{p} \cdot \nabla_{\mathbf{p}} f^{(1)} + \left(\frac{\partial f^{(1)}}{\partial t} \right) dt + \cdots, \tag{11}$$

give the Boltzmann equation

$$\left[f^{(1)}(\mathbf{x}, \mathbf{p}, t) + d\mathbf{x} \cdot \nabla_{\mathbf{x}} f^{(1)} + d\mathbf{p} \cdot \nabla_{\mathbf{p}} f^{(1)} + \left(\frac{\partial f^{(1)}}{\partial t} \right) dt + \cdots \right] d\mathbf{x} d\mathbf{p}$$
$$= f^{(1)}(\mathbf{x}, \mathbf{p}, t) d\mathbf{x} d\mathbf{p} + \left[\Gamma^{(+)} - \Gamma^{(-)} \right] d\mathbf{x} d\mathbf{p} dt \tag{12}$$

or

$$\mathbf{v} \cdot \nabla_{\mathbf{x}} f^{(1)} + \mathbf{F} \cdot \nabla_{\mathbf{p}} f^{(1)} + \frac{\partial f^{(1)}}{\partial t} = \Gamma^{(+)} - \Gamma^{(-)}. \tag{13}$$

Note that this can be derived for an arbitrary number of different chemical components as well.

In its complete form with the collision operator written more explicitly, the Boltzmann equation is a nonlinear integral differential equation and is particularly complicated. According to Harris (1971), 50 years elapsed from the time that Boltzmann derived the equation before an approximate

solution was found. With lattice Boltzmann methods, we approximately solve the equation from the particle perspective and focus on an equation strongly akin to Eq. (10); it explicitly contains the 'collide and stream' notion central to LBM.

4 Lattice Boltzmann Models (LBMs)

4.1   Basic LBM Framework and Equations

Lattice Boltzmann models vastly simplify Boltzmann's original conceptual view by reducing the number of possible particle spatial positions and microscopic momenta from a continuum to just a handful and similarly discretizing time into distinct steps. Particle positions are confined to the nodes of the lattice. Variations in momenta that could have been due to a continuum of velocity directions and magnitudes and varying particle mass are reduced (in the simple 2-D model we focus on here) to 8 directions, 3 magnitudes, and a single particle mass. Figure 21 shows the Cartesian lattice and the velocities $\mathbf{e}_a$ where a = 0, 1, …, 8 is a direction index and $\mathbf{e}_0$ = 0 denotes particles at rest. This model is known as D2Q9 as it is 2 dimensional and contains 9 velocities. This LBM classification scheme was proposed by Qian et al. (1992) and is in widespread use. Because particle mass is uniform (1 mass unit or *mu* in the simplest approach), these microscopic velocities and momenta are always effectively equivalent. The lattice unit (*lu*) is the fundamental measure of length in the LBM models and time steps (*ts*) are the time unit.
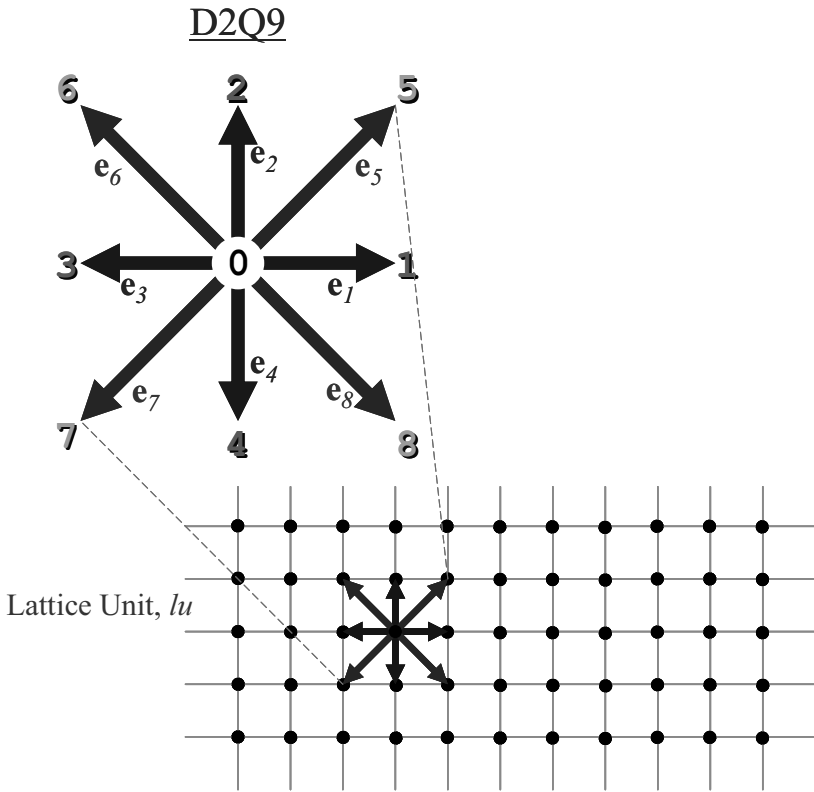
D2Q9

Figure 21. D2Q9 lattice and velocities.

The velocity magnitude of e_1 through e_4 is 1 lattice unit per time step or 1 *lu ts*$^{-1}$, and the velocity magnitude of e_5 through e_8 is $\sqrt{2}$ *lu ts*$^{-1}$. (While this is probably the most common velocity indexing scheme, be aware that others are in use.) These velocities are exceptionally convenient in that all of their *x*- and *y*-components are either 0 or ±1 (Figure 22).
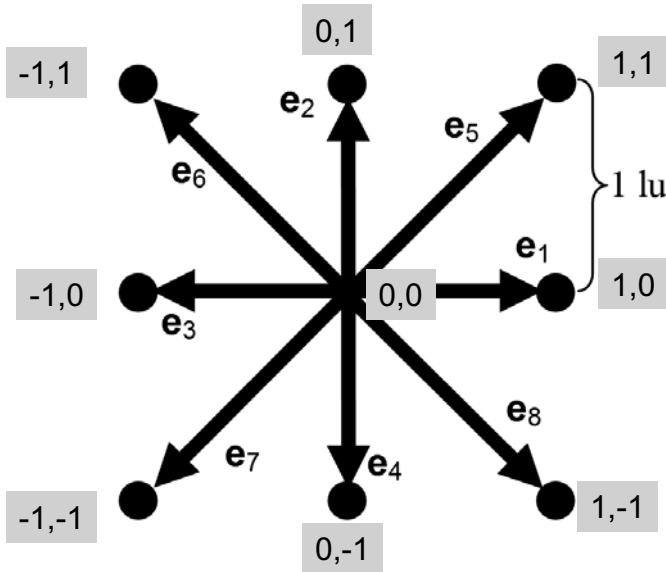
Figure 22. D2Q9 x, y velocity components.

The next step is to incorporate the single-particle distribution function f, which is essentially the one that appears in Eq. (10), except that it has only nine discrete 'bins' instead of being a continuous function. The distribution function can conveniently be thought of as a typical histogram representing a frequency of occurrence (Figure 23). The frequencies can be considered to be direction-specific fluid densities. Accordingly, the macroscopic fluid density is

$$\rho = \sum_{a=0}^{8} f_a \tag{14}$$
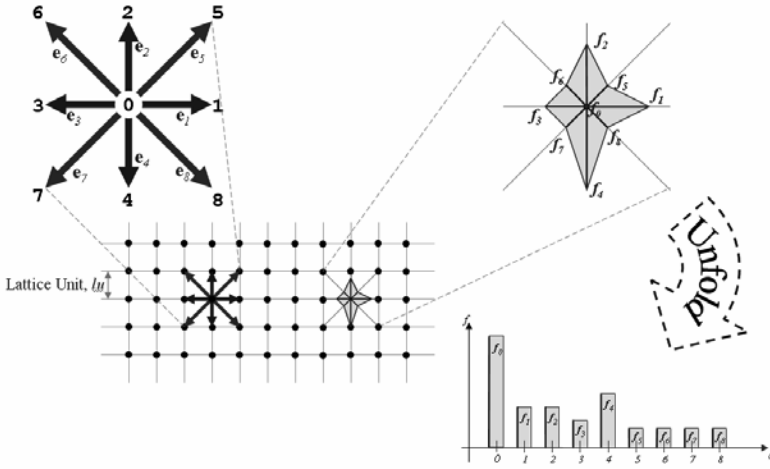
Figure 23. On-lattice and histogram views of the discrete single particle distribution function/direction-specific densities f_a.

The macroscopic velocity **u** is an average of the microscopic velocities $\mathbf{e}_a$ weighted by the directional densities f_a:

$$\mathbf{u} = \frac{1}{\rho} \sum_{a=0}^{8} f_a \mathbf{e}_a \; . \tag{15}$$

This simple equation allows us to pass from the discrete microscopic velocities that comprise the LBM back to a continuum of macroscopic velocities representing the fluid's motion.

The next steps are streaming and collision of the particles via the distribution function. The simplest approach uses the Bhatnagar-Gross-Krook Approximation for collision.

4.2 Single Relaxation Time BGK

The BGK (Bhatnagar-Gross-Krook) Approximation is used in the simplest LBM. Succi (2001) provides excellent discussions of more complex models and the path to BGK.

Streaming and collision (i.e., relaxation towards local equilibrium) look like this:

$$\underbrace{f_a\left(\mathbf{x}+\mathbf{e}_a\Delta t,t+\Delta t\right)=f_a\left(\mathbf{x},t\right)}_{\text{Streaming}}-\underbrace{\frac{\left[f_a\left(\mathbf{x},t\right)-f_a^{eq}\left(\mathbf{x},t\right)\right]}{\tau}}_{\text{Collision}} \tag{16}$$

where $f_a(\mathbf{x}+\mathbf{e}_a\Delta t,t+\Delta t)=f_a(\mathbf{x},t)$ is the streaming part and $(f_a(\mathbf{x},t)\text{-}f_a^{eq}(\mathbf{x},t))/\tau$ is the collision term. Although they can be combined into a single statement as they are in Eq. (16), collision and streaming steps must be separated if solid boundaries are present because the bounce back boundary condition is a separate collision.

Collision of the fluid particles is considered as a relaxation towards a local equilibrium and the D2Q9 equilibrium distribution function f^{eq} is defined as

$$f_a^{eq}(\mathbf{x})=w_a\rho(\mathbf{x})\left[1+3\frac{\mathbf{e}_a\cdot\mathbf{u}}{c^2}+\frac{9}{2}\frac{(\mathbf{e}_a\cdot\mathbf{u})^2}{c^4}-\frac{3}{2}\frac{\mathbf{u}^2}{c^2}\right] \tag{17}$$

where the weights w_a are 4/9 for the rest particles ($a=0$), 1/9 for $a=1, 2,$ 3, 4, and 1/36 for $a=5, 6, 7, 8$, and c is the basic speed on the lattice (1 lu ts^{-1} in the simplest implementation). Note that if the macroscopic velocity $\mathbf{u}=0$, the equilibrium f_a are simply the weights times the fluid density.

Implementing Eqs. (14) through (17) is relatively straightforward, and we provide details in the following sections. In the pseudo-code included in the text, the major axis weights w_a are referred to as WM and the diagonal weights are referred to as WD.

4.2.1 Macroscopic Variables

Computation of the macroscopic fluid density and velocity via Eqs. (14) and (15) simply involves looping through $a=0, 1, …, 8$ and computing the appropriate sums:

```
// Computing macroscopic density, rho, and velocity, u=(ux,uy).
for( j=0; j<LY; j++)
{
  for( i=0; i<LX; i++)
  {
```

```
  u_x[j][i] = 0.0;
  u_y[j][i] = 0.0;
  rho[j][i] = 0.0;

  if( !is_solid_node[j][i])
  {
    for( a=0; a<9; a++)
    {
      rho[j][i] +=        f[j][i][a];
      u_x[j][i] += ex[a]*f[j][i][a];
      u_y[j][i] += ey[a]*f[j][i][a];
    }
    u_x[j][i] /= rho[j][i];
    u_y[j][i] /= rho[j][i];
  }
}
}
}
```

4.2.2 Streaming

In streaming, we move the direction-specific densities f_a to the nearest neighbor lattice nodes. The scheme (originally due to Louis Colonna-Romano) shown in Figure 24 provides a convenient neighbor address relative to the point from which the fs are being streamed. The ip, in, jp, and jn are computed at the beginning of their respective loops.
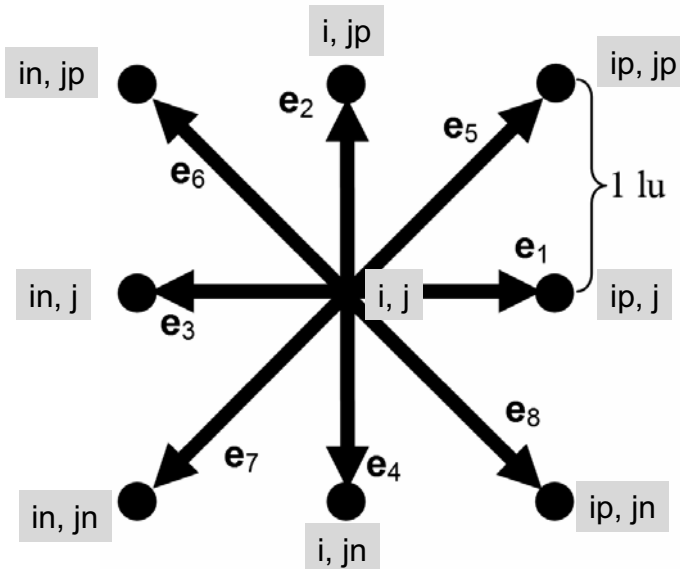
Figure 24. Neighbor referencing following Louis Colonna-Romano.

```
// Streaming step.
for( j=0; j<LY; j++)
{
  jn = (j>0   )?(j-1):(LY-1)
  jp = (j<LY-1)?(j+1):(0   )

  for( i=0; i<LX; i++)
  {
    if( !is_interior_solid_node[j][i])
    {
      in = (i>0   )?(i-1):(LX-1)
      ip = (i<LX-1)?(i+1):(0   )

      ftemp[j ][i ][0] = f[j][i][0];
      ftemp[j ][ip][1] = f[j][i][1];
      ftemp[jp][i ][2] = f[j][i][2];
      ftemp[j ][in][3] = f[j][i][3];
      ftemp[jn][i ][4] = f[j][i][4];
      ftemp[jp][ip][5] = f[j][i][5];
      ftemp[jp][in][6] = f[j][i][6];
      ftemp[jn][in][7] = f[j][i][7];
      ftemp[jn][ip][8] = f[j][i][8];
    }
  }
}
```

4.2.3 Equilibrium Distribution Function

Coding of the equilibrium distribution function is more involved due to the vector dot products (Eq. (17)). The implementation of these requires working with the individual x- and y-components of the velocity vectors.

Note that f^{eq} is computed in terms of a new velocity $\mathbf{u}^{eq}$. This is an adjusted velocity that incorporates external forces as will be discussed later. For now, suppose there are no external forces acting on particles, and so $\mathbf{u}^{eq} = \mathbf{u}$. We still want to put the $\mathbf{u}^{eq}$ mechanism in place now to avoid having to rewrite the computation of f^{eq} for such a minor (in terms of implementation) adjustment later when external forces are introduced.

```
// Compute the equilibrium distribution function, feq.
f1=3.;
f2=9./2.;
f3=3./2.;

for( j=0; j<LY; j++)
{
 for( i=0; i<LX; i++)
 {
  if( !is_solid_node[j][i])
   {
     rt0 = (4./9. )*rhoij;
```

```
     rt1 = (1./9. )*rhoij;
     rt2 = (1./36.)*rhoij;

     ueqxij =  uxij;
     ueqyij =  uyij;

     uxsq   =  ueqxij * ueqxij;
     uysq   =  ueqyij * ueqyij;

     uxuy5  =  uxeqij +  uyeqij;
     uxuy6  = -uxeqij +  uyeqij;
     uxuy7  = -uxeqij + -uyeqij;
     uxuy8  =  uxeqij + -uyeqij;

     usq    =  uxsq + uysq;

     feqij[0] = rt0*( 1.                                - f3*usq);
     feqij[1] = rt1*( 1. + f1*ueqxij + f2*uxsq          - f3*usq);
     feqij[2] = rt1*( 1. + f1*ueqyij + f2*uysq          - f3*usq);
     feqij[3] = rt1*( 1. - f1*ueqxij + f2*uxsq          - f3*usq);
     feqij[4] = rt1*( 1. - f1*ueqyij + f2*uysq          - f3*usq);
     feqij[5] = rt2*( 1. + f1*uxuy5  + f2*uxuy5*uxuy5 - f3*usq);
     feqij[6] = rt2*( 1. + f1*uxuy6  + f2*uxuy6*uxuy6 - f3*usq);
     feqij[7] = rt2*( 1. + f1*uxuy7  + f2*uxuy7*uxuy7 - f3*usq);
     feqij[8] = rt2*( 1. + f1*uxuy8  + f2*uxuy8*uxuy8 - f3*usq);
   }
  }
 }
```

4.2.4 Collision

Collision is the last key element of an LBM. Traverse the domain and implement Eq. (16) at each node:

```
// Collision step.
for( j=0; j<LY; j++)
  for( i=0; i<LX; i++)
    if( !is_solid_node[j][i])
      for( a=0; a<9; a++)
      {
        fij[a] = ftempij[a] - ( ftempij[a] - feqij[a])/tau;
      }
```

Note that we include a statement to skip this collision if the site at (i,j) is occupied by a solid (obstacle).

4.3 Viscosity

Fluid kinematic viscosity ν in the D2Q9 model is given by

$$\nu = \tfrac{1}{3}\left(\tau - \tfrac{1}{2}\right). \tag{18}$$

Its units are lu^2ts^{-1}. Note that $\tau > \tfrac{1}{2}$ for positive (physical) viscosity. Numerical difficulties can arise as τ approaches $\tfrac{1}{2}$. A value of $\tau = 1$ is safest and leads to $\nu = 1/6\ lu^2ts^{-1}$.

4.4 Boundary Conditions

Boundary conditions are necessary before we can compute any meaningful results. Since the early 1990s, many papers have proposed and investigated the behavior of various boundary conditions (Ziegler 1993; Skordos 1993; Inamuro et al. 1995; Noble et al. 1995; Ginzbourg and d'Humieres 1996; Maier et al. 1996; Zou and He 1997; Fang et al. 1998; Verberg and Ladd 2000; Zhang et al. 2002; Ansumali and Karlin 2002; Chopard and Dupuis 2003). This work continues, though workable boundary conditions for many types of simulations are now available. In this chapter, we will give details for periodic, bounceback, and, following Zou and He (1997), constant pressure and constant velocity boundaries. Readers wishing to explore more advanced work on boundary conditions are referred to Yu et al. (2003), Ginzburg and d'Humières (2003), Zhou et al. (2004), and the open literature. In Chapter 8 we present boundary conditions for solute transport simulation.

   In general, we have a great deal of temporal/spatial flexibility in applying boundary conditions in LBM. In fact, the ability to easily incorporate complex solid boundaries is one of the most exciting aspects of these models and has made it possible to simulate realistic porous media for example.

4.4.1 Periodic Boundaries

The simplest boundary conditions are 'periodic' in that the system becomes closed by the edges being treated as if they are attached to opposite edges. Most early papers used these boundaries along with bounceback boundaries. In simulating flow in a slit for example, bounceback boundaries would be applied at the slit walls and periodic boundaries would be applied to the 'open' ends of the slit. Figure 25 illustrates the resulting topology of the computational domain using such 'wrap-around' boundaries.
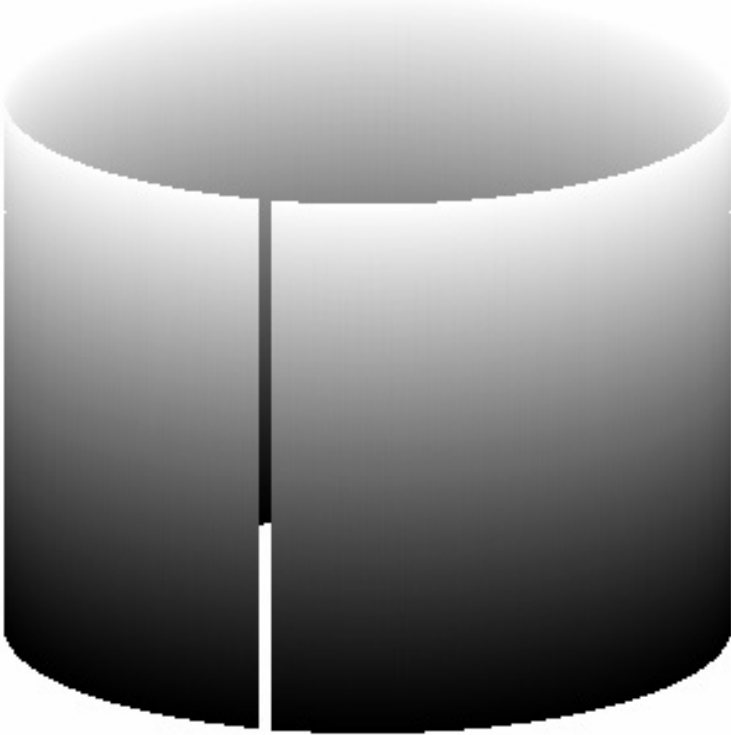
Figure 25. Cylindrical topology of computational domain when periodic in one direction. Note that the gap in the cylinder is just for illustration purposes to emphasize how the domain wraps around on itself.

Fully periodic boundaries are also useful in some cases (for example, simulation of an infinite domain of multiphase fluids). In this case the computational domain topology is that of a torus (Figure 26).

For boundary nodes, neighboring points are on the opposite boundary. Using the normal referencing of neighbors (Figure 24), here are conditionals that that check if the neighboring nodes lie outside the computational

domain and assign the appropriate node on the opposite boundary in that case to achieve periodicity:

```
ip = ( i<LX-1)?( i+1):( 0   );
in = ( i>0   )?( i-1):( LX-1);
jp = ( j<LY-1)?( j+1):( 0   );
jn = ( j>0   )?( j-1):( LY-1);
```

where

```
LHS = (COND)?(TRUE_RHS):(FALSE_RHS);
```

means

```
if( COND) { LHS=TRUE_RHS;} else{ LHS=FALSE_RHS;}
```
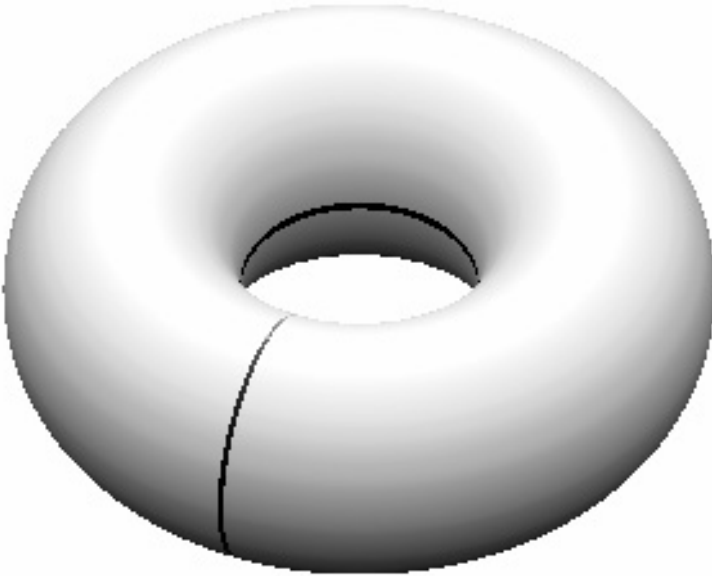

Figure 26. Torroidal topology of computational domain that is periodic in both directions. Gaps in torus illustrate how the domain wraps around on itself.

4.4.2 Bounceback Boundaries

As already mentioned, bounceback boundaries are particularly simple and have played a major role in making LBM popular among modelers interested in simulating fluids in domains characterized by complex geometries such as those found in porous media. Their beauty lies in that one simply needs to designate a particular node as a solid obstacle and no special programming treatment is required. Thus it is trivial to incorporate images of porous media for example and immediately compute the flow in them.
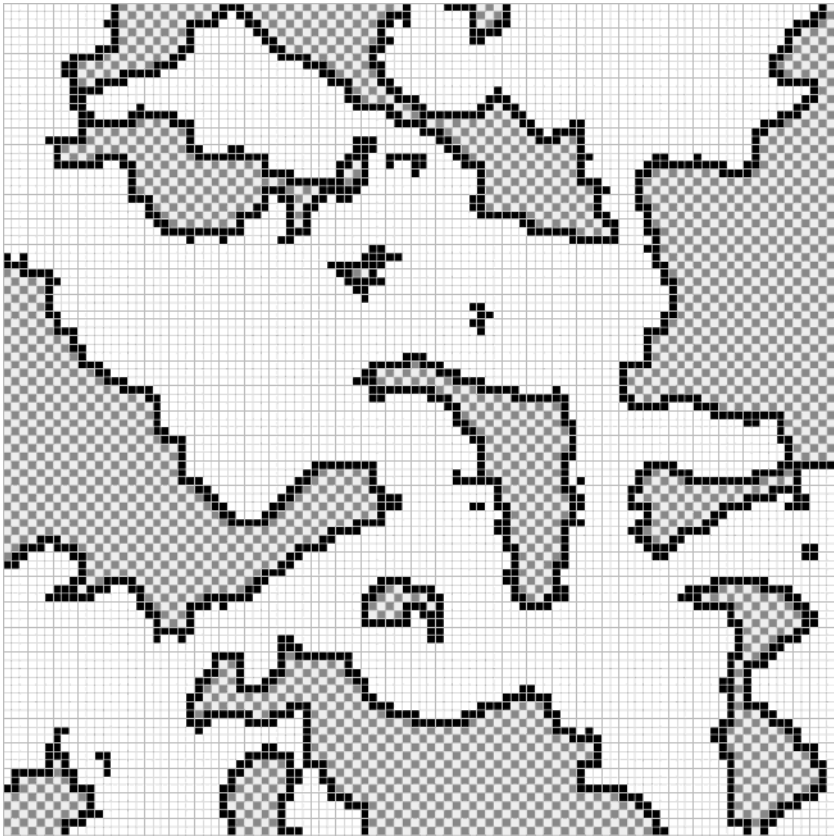
Figure 27. Classification of solids: black nodes are surface (boundary) solids; gray checker nodes denote interior (isolated) solids. (Often the percentage of isolated solids is much greater than surface solids.)

As indicated in Figure 27, we separate solids into two types – boundary solids that lie at the solid-fluid interface and isolated solids that do not contact fluid. With this division it is possible to eliminate unnecessary computations at inactive nodes; this can be particularly important in the simulation of fluid flow in fractured media for example, where the fraction of the total domain occupied by open space accessible to fluids can be very small.

Bounceback boundaries come in several variants (Succi 2001) and do not work perfectly (e.g., Gallivan et al. 1997; Inamuro et al. 1995). Nevertheless, with proper consideration of the effective boundary location and for τ ≈ 1 (Chen et al. 1996), quite satisfactory results can be obtained as will be demonstrated below. Here we use the 'mid-plane' bounceback scheme in which the densities are temporarily stored inside the solids and re-emerge at the next time step. Figure 28 illustrates the process.
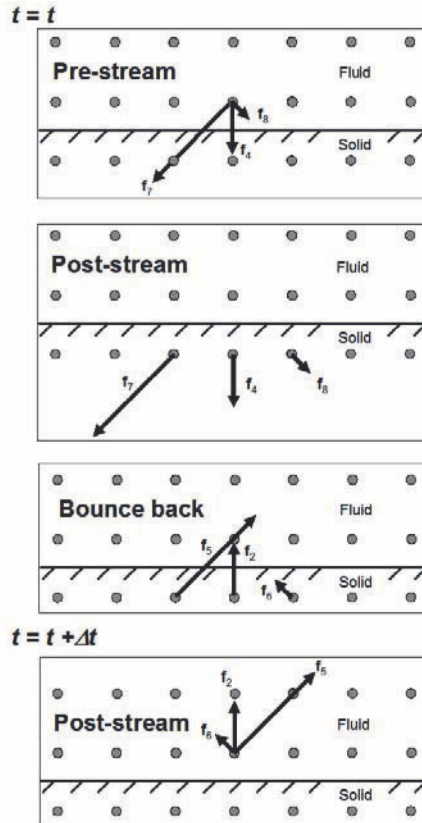
Figure 28. Illustration of mid-plane bounceback movement of direction-specific densities f_a. The effective wall location is halfway between the fluid and solid nodes. (Figure from Sukop and Or, 2004).

Code to accomplish this can be included with the collision routine. While traversing the lattice to perform the collision step, if node (i,j) is determined to be a boundary solid, the normal collision computation is omitted and the densities are bounced back as illustrated in Figure 28. The subsequent streaming step moves the densities back into the fluid domain. Here is the code for performing the bounceback step on node (i,j):

```
// Standard bounceback.
temp    = fij[1]; fij[1]  = fij[3]; fij[3] = temp;
temp    = fij[2]; fij[2]  = fij[4]; fij[4] = temp;
temp    = fij[5]; fij[5]  = fij[7]; fij[7] = temp;
temp    = fij[6]; fij[6]  = fij[8]; fij[8] = temp;
```

4.4.3 Von Neumann (Flux) Boundaries

Von Neumann boundary conditions constrain the flux at the boundaries. A velocity vector consisting of x and y components $\mathbf{u}_0 = \begin{bmatrix} u_0 \\ v_0 \end{bmatrix}$ is specified from which density/pressure is computed on the basis of conditions inside the domain.

   Macroscopic density/pressure is only part of what needs to be computed. The unknown directional densities also need to be computed. After the streaming step, there are three unknown directional densities at each lattice node pointing from the boundary into the domain. These unknowns can be solved for in a way that maintains a specified velocity at their lattice nodes. Because symmetry makes determination of the boundary condition on other boundaries almost trivial, we only derive one case – a north boundary – in detail. The other three are illustrated in pseudocode only.

   Figure 29 shows the unknowns at a north boundary after streaming.
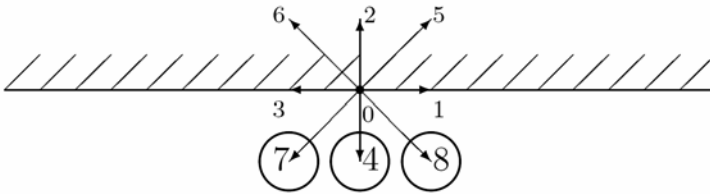
Figure 29. Direction-specific densities f_4, f_7, and f_8 are unknown after streaming at a north surface/boundary.

Suppose the boundary condition is that vertical velocity $v = v_0$ and horizontal velocity $u = 0$. That is,

$$\mathbf{u}_0 = \begin{bmatrix} 0 \\ v_0 \end{bmatrix}. \tag{19}$$

The contributions from f_a for a in $\{0,1,2,3,5,6\}$ are already known because they arrived from other nodes inside the domain. We need to solve for ρ, f_4, f_7 and f_8, which means we need four equations.

The macroscopic density formula is one equation:

$$\rho = \sum_a f_a.$$ (20)

By considering the individual fs that can contribute to x and y velocities, the formula for macroscopic velocity

$$\mathbf{u}_0 = \frac{1}{\rho} \sum_a f_a \mathbf{e}_a$$ (21)

gives two equations, one for each direction:

$$0 = f_1 - f_3 + f_5 - f_6 - f_7 + f_8$$ (22)

and

$$\rho v_0 = f_2 - f_4 + f_5 + f_6 - f_7 - f_8.$$ (23)

A fourth equation can be written by assuming that the bounceback condition holds in the direction normal to the boundary

$$f_2 - f_2^{eq} = f_4 - f_4^{eq}$$ (24)

as proposed by Zou and He (1997).

This is a system of four equations with four unknowns, and it can be solved as follows.

Eqs. (20) and (23) have the directional density unknowns f_4, f_7 and f_8 in common, so rewrite them with those variables on the left hand side:

$$f_4 + f_7 + f_8 = \rho - f_0 - f_1 - f_2 - f_3 - f_5 - f_6$$ (25)

$$f_4 + f_7 + f_8 = f_2 + f_5 + f_6 - \rho v_0$$ (26)

Then equate the right hand sides

$$\rho - f_0 - f_1 - f_2 - f_3 - f_5 - f_6 = f_2 + f_5 + f_6 - \rho v_0 \qquad (27)$$

and solve for ρ:

$$\rho + \rho v_0 = f_0 + f_1 + f_2 + f_3 + f_5 + f_6 + f_2 + f_5 + f_6 \Rightarrow$$
$$\rho = \frac{f_0 + f_1 + f_3 + 2(f_2 + f_5 + f_6)}{1 + v_0}. \qquad (28)$$

Now, from equation (24), we solve for f_4:

$$f_4 = f_2 - f_2^{eq} + f_4^{eq} = f_2 - \frac{2}{3} \rho v_0. \qquad (29)$$

Here is a detailed look at how most of the terms in $f_4^{eq} - f_2^{eq}$ cancel out:

$$f_4^{eq} - f_2^{eq} =$$
$$\left(\frac{1}{9} \rho + \frac{1}{3} \rho (-1 \cdot v_0) + \frac{1}{2} \rho v_0^2 - \frac{1}{6} \rho \left(u_0^2 + v_0^2 \right) \right)$$
$$- \left(\frac{1}{9} \rho + \frac{1}{3} \rho (1 \cdot v_0) + \frac{1}{2} \rho v_0^2 - \frac{1}{6} \rho \left(u_0^2 + v_0^2 \right) \right) = -\frac{2}{3} \rho v_0 \qquad (30)$$

Eqs. (28) and (29) give the fluid density and the direction-specific density in the '4' direction respectively.

We proceed by substituting equations (22) and (29) into equation (23) to solve for f_7. Eq. (30) is used to replace f_4, and equation (22) is used to replace f_8:

$$\rho v_0 = f_2 - f_4 + f_5 + f_6 - f_7 - f_8 \Rightarrow$$

$$\rho v_0 = f_2 - \underbrace{\left(f_2 - \frac{2}{3}\rho v_0 \right)}_{f_4} + f_5 + f_6 - f_7 - \underbrace{\left(-f_1 + f_3 - f_5 + f_6 + f_7 \right)}_{f_8} \Rightarrow$$

$$\rho v_0 = \frac{2}{3}\rho v_0 + 2f_5 - 2f_7 + f_1 - f_3 \Rightarrow$$

$$f_7 = f_5 + \frac{1}{2}(f_1 - f_3) - \frac{1}{6}\rho v_0.$$

$$(31)$$

To solve for f_8, the last unknown, we can repeat the last step except equation (22) is used to substitute for f_7 this time:

$$\rho v_0 = f_2 - f_4 + f_5 + f_6 - f_7 - f_8 \Rightarrow$$

$$\rho v_0 = f_2 - \underbrace{\left(f_2 - \frac{2}{3}\rho v_0 \right)}_{f_4} + f_5 + f_6 - \underbrace{\left(f_1 - f_3 + f_5 - f_6 + f_8 \right)}_{f_7} - f_8 \Rightarrow$$

$$\rho v_0 = \frac{2}{3}\rho v_0 + 2f_6 - 2f_8 - (f_1 - f_3) \Rightarrow$$

$$f_8 = f_6 - \frac{1}{2}(f_1 - f_3) - \frac{1}{6}\rho v_0.$$

$$(32)$$

To summarize the procedure, we specify a velocity, e.g., equation (19), at the boundary and solve for the macroscopic density and three unknown directional densities via four equations. The equations come from the usual macroscopic variable formulae and the assumption that bounceback is satisfied in the direction normal to the boundary.

Below we show pseudocode implementing Zou and He flux boundaries on all four sides. First we solve for the fluid density and subsequently use that to solve for the unknown direction-specific densities.

```
// Zou and He velocity BCs on north side.
for( i=0; i<LX; i++)
{
  fi    = ftemp[LY-1][i];
  rho0  = (      fi[0] + fi[1] + fi[3]
         + 2.*( fi[2] + fi[5] + fi[6])) / ( 1. + uy0);
  ru    = rho0*uy0;
```

```
   fi[4] = fi[2] - (2./3.)*ru;
   fi[7] = fi[5] - (1./6.)*ru + (1./2.)*( fi[1]-fi[3]);
   fi[8] = fi[6] - (1./6.)*ru + (1./2.)*( fi[3]-fi[1]);
}

// Zou and He velocity BCs on south side.
for( i=0; i<LX; i++)
{
   fi    = ftemp[0][i];
   rho0  = (      fi[0] + fi[1] + fi[3]
          + 2.*( fi[4] + fi[7] + fi[8])) / ( 1. - uy0);
   ru    = rho0*uy0;
   fi[2] = fi[4] + (2./3.)*ru;
   fi[5] = fi[7] + (1./6.)*ru - (1./2.)*( fi[1]-fi[3]);
   fi[6] = fi[8] + (1./6.)*ru - (1./2.)*( fi[3]-fi[1]);
}

// Zou and He velocity BCs on east side.
for( j=0; j<LY; j++)
{
   fj    = ftemp[j][LX-1];
   rho0  = (      fj[0] + fj[2] + fj[4]
          + 2.*( fj[1] + fj[5] + fj[8])) / ( 1. + ux0);
   ru    = rho0*ux0;
   fj[3] = fj[1] - (2./3.)*ru;
   fj[7] = fj[5] - (1./6.)*ru + (1./2.)*( fj[2]-fj[4]);
   fj[6] = fj[8] - (1./6.)*ru + (1./2.)*( fj[4]-fj[2]);
}

// Zou and He velocity BCs on west side.
for( j=0; j<LY; j++)
{
   fj    = ftemp[j][0];
   rho0  = (      fj[0] + fj[2] + fj[4]
          + 2.*( fj[3] + fj[7] + fj[6])) / ( 1. - ux0);
   ru    = rho0*ux0;
   fj[1] = fj[3] + (2./3.)*ru;
   fj[5] = fj[7] + (1./6.)*ru - (1./2.)*( fj[2]-fj[4]);
   fj[8] = fj[6] + (1./6.)*ru - (1./2.)*( fj[4]-fj[2]);
}
```

The assumption of zero velocity parallel to the boundaries is a simplification we employ that is not essential to the method.

4.4.4 Dirichlet (Pressure) Boundaries

Dirichlet boundary conditions constrain the pressure/density at the boundaries. The solution for these boundaries is closely related to that given above for the velocity boundaries. A density ρ_0 is specified from which velocity is computed. (Note that specifying density is equivalent to specifying pressure since there is an equation of state (EOS) relating them directly. For the single component D2Q9 model, the relationship is simply P

multiphase models and will be discussed later.) We assume that velocity tangent to the boundary is zero and solve for the component of velocity normal to the boundary.

In addition to the macroscopic velocity, we need to determine a proper distribution function at the boundary nodes. After the streaming step, there are three unknown directional densities at each lattice node pointing from the boundary into the domain. These unknowns can be solved for in a way that maintains the specified pressure/density ρ_0 at their lattice nodes.

Due to symmetry, we only derive one case – a north boundary – in detail. The other three cases are illustrated in pseudocode.

Figure 30 shows the unknowns at a north boundary after streaming (which are the same unknowns as the unknowns on the north boundary in the velocity boundary condition derivation above).
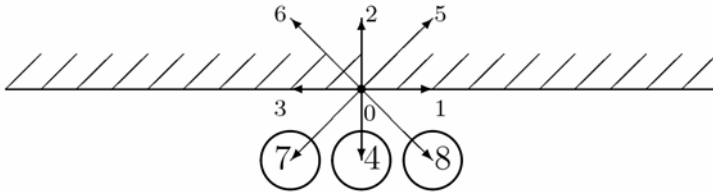
Figure 30. Unknowns (circled) after streaming at a north surface/boundary.

Given the boundary condition $\rho = \rho_0$ and the known directional densities f_a for a in $\{0,1,2,3,5,6\}$, we need to solve for v, f_4, f_7 and f_8. (Recall that v is the y-component of velocity and is in the direction normal to the north boundary.) As in the velocity boundary conditions described above, this means we need four equations. We get them again from the macroscopic density formula

$$\rho_0 = \sum_a f_a, \tag{33}$$

the formula for macroscopic velocity

$$\mathbf{u} = \frac{1}{\rho_0} \sum_a f_a \mathbf{e}_a , \tag{34}$$

which gives the two equations

$$0 = f_1 - f_3 + f_5 - f_6 - f_7 + f_8 \tag{35}$$

$$\rho_0 v = f_2 - f_4 + f_5 + f_6 - f_7 - f_8 , \tag{36}$$

and the assumption that bounceback holds in the direction normal to the boundary

$$f_2 - f_2^{eq} = f_4 - f_4^{eq} \tag{37}$$

as proposed by Zou and He (1997).

This is a system of four equations with four unknowns, which can be solved as follows.

Eqs. (20) and (23) have the directional density unknowns f_4, f_7 and f_8 in common, so rewrite them with those variables on the left hand side:

$$f_4 + f_7 + f_8 = \rho_0 - f_0 - f_1 - f_2 - f_3 - f_5 - f_6 \tag{38}$$

$$f_4 + f_7 + f_8 = f_2 + f_5 + f_6 - \rho_0 v \tag{39}$$

Then equate the right hand sides

$$\rho_0 - f_0 - f_1 - f_2 - f_3 - f_5 - f_6 = f_2 + f_5 + f_6 - \rho_0 v \tag{40}$$

and solve for v:

$$\rho_0 + \rho_0 v = f_0 + f_1 + f_2 + f_3 + f_5 + f_6 + f_2 + f_5 + f_6 \Rightarrow$$
$$v = -1 + \frac{f_0 + f_1 + f_3 + 2(f_2 + f_5 + f_6)}{\rho_0} . \tag{41}$$

Now, from equation (24), we solve for f_4:

$$f_4 = f_2 - f_2^{eq} + f_4^{eq} = f_2 - \frac{2}{3}\rho_0 v \,. \tag{42}$$

The detailed cancellation of terms in f_4^{eq}–f_2^{eq} is identical to Eq. (30).

We proceed by substituting equations (22) and (29) into equation (23) to solve for f_7. Eq. (30) is used to replace f_4, and equation (22) is used to replace f_8:

$$\rho_0 v = f_2 - f_4 + f_5 + f_6 - f_7 - f_8 \Rightarrow$$

$$\rho_0 v = f_2 - \underbrace{\left(f_2 - \frac{2}{3}\rho_0 v \right)}_{f_4} + f_5 + f_6 - f_7 - \underbrace{\left(-f_1 + f_3 - f_5 + f_6 + f_7 \right)}_{f_8} \Rightarrow$$

$$\rho_0 v = \frac{2}{3}\rho_0 v + 2f_5 - 2f_7 + f_1 - f_3 \Rightarrow \tag{43}$$

$$f_7 = f_5 + \frac{1}{2}(f_1 - f_3) - \frac{1}{6}\rho_0 v.$$

To solve for f_8, the last unknown, we can repeat the last step except using equation (22) to substitute for f_7 this time:

$$\rho_0 v = f_2 - f_4 + f_5 + f_6 - f_7 - f_8 \Rightarrow$$

$$\rho_0 v = f_2 - \underbrace{\left(f_2 - \frac{2}{3}\rho_0 v \right)}_{f_4} + f_5 + f_6 - \underbrace{\left(f_1 - f_3 + f_5 - f_6 + f_8 \right)}_{f_7} - f_8 \Rightarrow$$

$$\rho_0 v = \frac{2}{3}\rho_0 v + 2f_6 - 2f_8 - (f_1 - f_3) \Rightarrow \tag{44}$$

$$f_8 = f_6 - \frac{1}{2}(f_1 - f_3) - \frac{1}{6}\rho_0 v.$$

To summarize the procedure, we specify a velocity, e.g., equation (19), at the boundary and solve for the macroscopic density and three unknown directional densities via four equations. The equations come from the usual macroscopic variable formulae and the assumption that bounceback is satisfied in the direction normal to the boundary.

Pseudocode implementing the Zou and He pressure boundaries on all four sides is listed below. First we solve for the velocity and subsequently use it to solve for the unknown direction-specific densities.

```
// Zou and He pressure boundary on north side.
for( i=0; i<LX; i++)
{
  fi    = ftemp[LY-1][i];
  uy0   = -1. + (     fi[0] + fi[1] + fi[3]
                + 2.*( fi[2] + fi[5] + fi[6])) / rho0;
  ru    = rho0*uy0;
  fi[4] = fi[2] - (2./3.)*ru;
  fi[7] = fi[5] - (1./6.)*ru + (1./2.)*( fi[1]-fi[3]);
  fi[8] = fi[6] - (1./6.)*ru + (1./2.)*( fi[3]-fi[1]);
}

// Zou and He pressure boundary on south side.
for( i=0; i<LX; i++)
{
  fi    = ftemp[0][i];
  uy0   = -1. + (     fi[0] + fi[1] + fi[3]
                + 2.*( fi[4] + fi[7] + fi[8])) / rho0;
  ru    = rho0*uy0;
  fi[2] = fi[4] - (2./3.)*ru;
  fi[5] = fi[7] - (1./6.)*ru + (1./2.)*( fi[3]-fi[1]);
  fi[6] = fi[8] - (1./6.)*ru + (1./2.)*( fi[1]-fi[3]);
}

// Zou and He pressure boundary on east side.
for( j=0; j<LY; j++)
{
  fj    = ftemp[j][LX-1];
  ux0   = -1. + (     fj[0] + fj[2] + fj[4]
                + 2.*( fj[1] + fj[5] + fj[8])) / rho0;
  ru    = rho0*ux0;
  fj[3] = fj[1] - (2./3.)*ru;
  fj[7] = fj[5] - (1./6.)*ru + (1./2.)*( fj[2]-fj[4]);
  fj[6] = fj[8] - (1./6.)*ru + (1./2.)*( fj[4]-fj[2]);
}

// Zou and He pressure boundary on west side.
for( j=0; j<LY; j++)
{
  fj    = ftemp[j][0];
  ux0   = -1. + (     fj[0] + fj[2] + fj[4]
                + 2.*( fj[3] + fj[7] + fj[6])) / rho0;
  ru    = rho0*ux0;
  fj[1] = fj[3] + (2./3.)*ru;
  fj[5] = fj[7] + (1./6.)*ru + (1./2.)*( fj[4]-fj[2]);
  fj[8] = fj[6] + (1./6.)*ru + (1./2.)*( fj[2]-fj[4]);
}
```

4.5 Incorporating Gravity

The force imparted by gravitational acceleration is incorporated in a velocity term. Consider,

$$\mathbf{F} = m\mathbf{a} = m\frac{d\mathbf{u}}{dt} \, . \tag{45}$$

Recognizing that the density is proportional to the mass and that the relaxation time τ is the elementary time of collisions, we can rearrange Eq. (45) to

$$\Delta\mathbf{u} = \frac{\tau\mathbf{F}}{\rho} \, , \tag{46}$$

where $\Delta\mathbf{u}$ is a change in velocity. Finally we can write

$$\mathbf{u}^{eq} = \mathbf{u} + \Delta\mathbf{u} = \mathbf{u} + \frac{\tau\mathbf{F}}{\rho} \, , \tag{47}$$

where $\mathbf{u}^{eq}$ is used in computation of f_{eq} and was introduced above. As we will see below, other forces can be added in the same way.

5 Single component, single phase (SCSP) LBM

The simplest LBM models that can be implemented based on the concepts presented in the preceding chapter are the Single Component, Single Phase (SCSP) models. These are single fluid models that might represent the behavior of a single gas or liquid phase for example. Here we present examples of Poiseuille flows driven by gravity, pressure gradients, and fixed velocity boundaries and compare them with analytical solutions.

We also demonstrate flows at higher Reynolds numbers in a well-studied geometry (flow past a cylinder) and show good agreement between classic experimental results and the LBM. Flows in more complex geometries are trivial to implement with bounceback boundaries; the geometry of the solids simply needs to be called into the model.

5.1 Poiseuille Flow

5.1.1 Gravity

The Poiseuille flow in a slit driven by gravity may be the simplest flow system that can be simulated with LBM. It requires only bounceback boundaries along the walls. Periodic boundaries can be used in the flow direction and flow that leaves the domain reenters on the opposite end of the slit. The system is effectively infinite in the flow direction and there are no entry or exit effects. In this case, G^* in Eqs. (4) and (5) is the hydrostatic pressure gradient ρg. Figure 31 show the results from such a run. Agreement with the analytical solution is excellent and comparable agreement is obtained for channels as small as 5 lattice units wide.
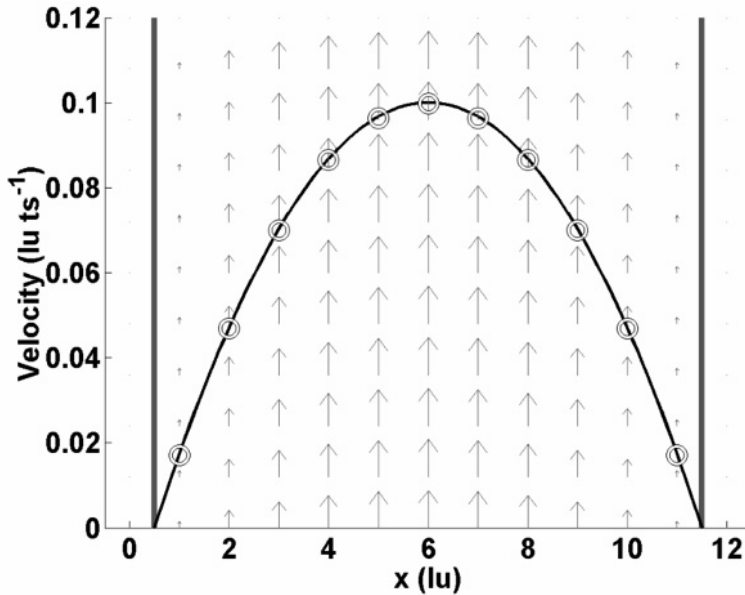
Figure 31. LBM velocity vector field and node-wise simulated velocities (circles) with analytical Poiseuille velocity profile (solid line) for gravity-driven flow with bounceback and periodic boundaries at Re ≈ 4.4. Gravity $g \approx$ 1.102 · 10⁻³ and $\tau = 1$. Note that no-slip zero velocity conditions apply halfway between wall nodes and the interior nodes; the effective wall position lies between nodes 0 and 1 on the left hand side and between nodes 11 and 12 on the right.

It is critical that lattice velocities be restricted to less than approximately 0.1 $lu\ ts^{-1}$. This is a consequence of the low Mach number approximation that allows LBM to simulate hydrodynamics. The procedure for solving a Poiseuille problem driven by gravity for example can thus involve steps such as the following:

- Choose a Reynolds number. This will permit linkage with a given real flow problem (but unsteady flow may not begin at Re ≈ 1000 in a slit or Re ≈ 2000 in a pipe (see Succi 2001 p. 99)). We assume the Reynolds number will be determined by the average flow velocity.
- Choose a combination of slit width and viscosity that will give a maximum velocity < ≈0.1 $lu\ ts^{-1}$; smaller is better. The maximum velocity will be 3/2 of the average for Poiseuille flow in a slit. Note that it is best to use $\tau = 1$ for the simple bounceback boundaries yielding a kinematic viscosity of 1/6 $lu^2\ ts^{-1}$.
- Choose a fluid density.
- Solve for the gravitational acceleration needed to drive the flow by rearranging Eq. (5).

To make this explicit, suppose we wish to simulate a Poiseuille flow with Re ≈ 4.4 as above. The Reynolds number is Re = $u\ 2a/v$ where u is the average fluid velocity, $2a$ is the characteristic length (the channel width in this case), and v is the kinematic viscosity. If $v = 1/6\ lu^2\ ts^{-1}$ and $u = 2/3\ u_{max} = 2/3\ 0.1\ lu\ ts^{-1} = 0.0667\ lu\ ts^{-1}$, then $2a = \mathrm{Re}\,v/u$ needs to be 11 lu.

Eq. (5) can be re-written as

$$G^* = \frac{3u_{average}\mu}{a^2}.$$ (48)

Replacing G^* with ρg for this gravitationally-driven problem and replacing the dynamic viscosity μ with $v\rho$ lead to

$$g = \frac{3u_{average}v}{a^2}.$$ (49)

For the problem at hand, $g = 1.102{\times}10^{-3}\ lu\ ts^{-2}$.

5.1.2 Velocity Boundaries

The ability to specify the velocity/fluid flux on a boundary is appealing and potentially useful. However, given the compressible nature of the simple LBM we have introduced here, some difficulties can arise.

Velocity boundaries are of particular interest in examining the notion of entry length effects. Basically, if fluid enters a pipe from a tank for example, the Poiseuille profile will not be developed immediately inside the pipe; some distance of flow in the pipe will be necessary before the Poiseuille flow is fully developed. This distance can be remarkably long, which suggests that in many applications, Poiseuille flow may not be a good approximation.
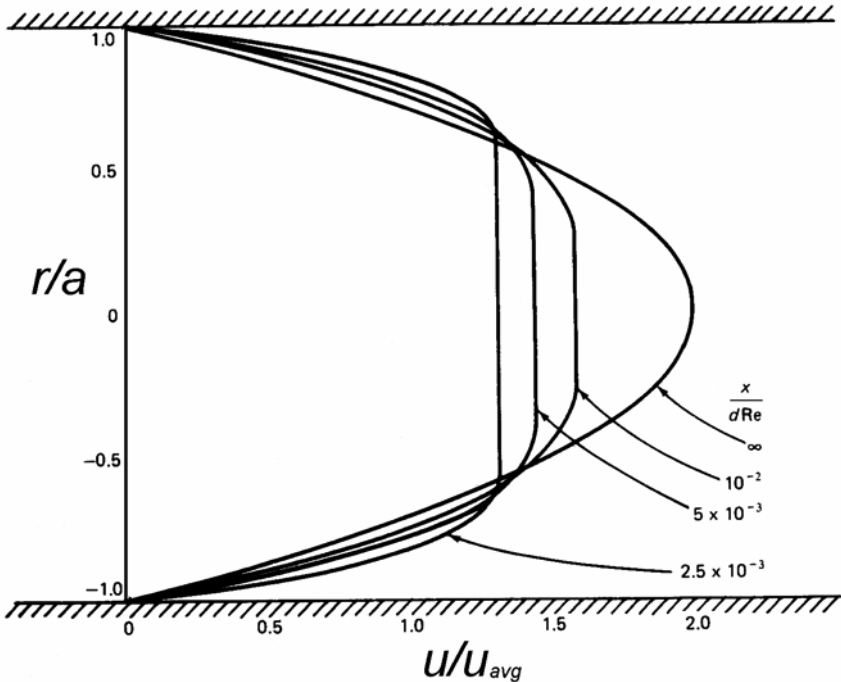
Figure 32. Simplified view of entry length effects in a pipe (Reprinted from Tritton, 1988, with permission from Oxford University Press). x/dRe **is the dimensionless distance from the pipe entrance with** x **the dimensional distance,** d **the pipe diameter (= 2a), and Re the Reynolds number.**

In Figure 32 we show a simplified version of entry length effects due to Smith (1960) as they were presented in Tritton (1988). We can set up a LB model to investigate entry length effects by placing constant velocity boundaries on the opposite ends of a slit. In Figure 33, we show results for Re ≈ 100 where we use the incompressible LBM of Zou and He (1997). Clearly the results differ from the simplified representation in Figure 32.
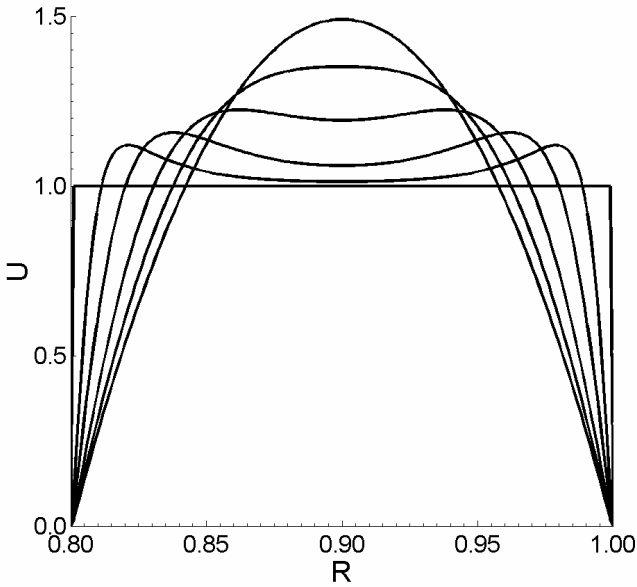
Figure 33. Simulated entry length effects in a slit using incompressible LBM. Velocity profiles at dimensionless distance from inlet Z = 0, 0.016, 0.033, 0.066, 0.1083, and 0.245 (scaled relative to slit width of 0.2). Re = 100. U is dimensionless velocity and R is dimensionless position relative to slit width of 0.2. Compare with Figure 34.

Nouar et al. (1995) presented results for the flow in the annulus between two pipes based on solution of the complete Navier-Stokes equations (Figure 34). As the inner and outer pipe radii approach one another, this is comparable to flow in a slit and the agreement between the finite difference solution of Nouar et al. (1995) and the LBM model for the slit is good.
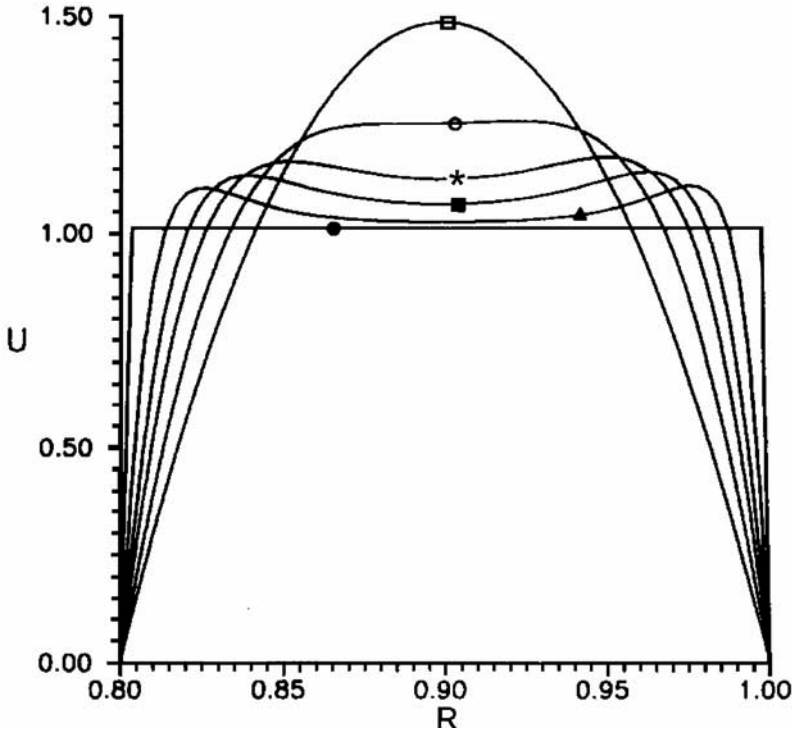
Figure 34. Entry length effects in annular space between two pipes. Velocity profiles at dimensionless distance from inlet $Z = 0$, 0.016, 0.033, 0.066, 0.1083, and 0.245 (scaled relative to radius of large pipe). Re = 100. U is dimensionless velocity and R is dimensionless radial position relative to radius of large pipe. (Reprinted from Nouar et al. (1995), Copyright (1995), with permission from Elsevier).

The difficulty with velocity boundaries in the simple compressible fluid model is that if one uses identical constant velocities at each end of the domain, the mass of fluid in the system will change with time. The boundary-driven flow must be accompanied by a pressure gradient and hence the pressure must be lower at the outlet. The pressure and density are related through an ideal gas law of the form $P = \rho/3$ in this model. Thus the densities at the input and output must be different. If the velocity boundaries on each end of the domain are equal, mass will accumulate in the system because the mass influx of fluid ($v_{in}\rho_{in}$) will exceed the outflux ($v_{out}\rho_{out}$).

Naturally, this problem increases in severity as the pressure difference increases. The incompressible model of Zou and He (1997) (which we use to obtain the results in Figure 33), pressure boundaries, or gravity-driven flow can be used to avoid this complication.

5.1.3 Pressure Boundaries

The Equation of State (EOS) relating pressure and density in the SCSP D2Q9 model is

$$P = c_s^2 \rho = \frac{\rho}{3}. \qquad (50)$$

Flows can be driven with pressure gradients of any desired magnitude by setting the boundary densities in accordance with this equation (provided the maximum velocity remains small relative to 1 $lu\ ts^{-1}$. In some of our work, we have found it useful to begin with a real-world, non-dimensional head gradient, convert to a real-world pure pressure gradient, compute the Reynolds number and then define an equivalent LBM system. Pressure boundaries will be used extensively in conjunction with single component multiphase models below.

5.2 Flows Past a Cylinder

Flows past a cylinder have long been a subject of interest to fluid dynamicists. Extensive work has been done to develop empirical charts of drag coefficient vs. Reynolds number (e.g., Tritton 1988 and Douglas et al. 2001). The drag force F_D is defined in terms of the drag coefficient C_D as

$$F_D = r\rho u_0^2\ C_D \qquad (51)$$

By solving for the force exerted on the fluid by the cylinder (a disk in 2-D), we can compute a force balance and solve for the necessary opposing force that will drive the fluid to a constant velocity. If we choose to use gravitational acceleration to drive the fluid, then the downward force will be given by

$$F = \rho g\left(LW - \pi r^2\right) \qquad (52)$$

where ρgLW is simply the weight of fluid in a domain of dimensions $L \times W$ and πr^2 is the disk area (which is not occupied by fluid and hence does not contribute to the weight). Equating this gravitational force and the drag force at a particular Reynolds number allows computation of g for use in the simulation. A potential complication arises from the discretization of the disk. The surface area (length) of the 'staircase' approximation in 2-D is $8r$ instead of the $2\pi r$ of a smooth disk. Thus the drag force may be higher by $8/2\pi \approx 1.27$. More experimentation is required.

Taneda photographically captured flow past a cylinder behavior for various Reynolds numbers and his results provide an excellent means of verifying LBM computations. For example in Figure 35 we simulate creeping flow under conditions like those shown in Figure 4.
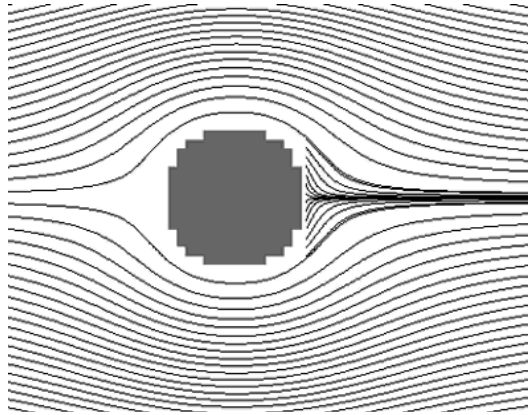
Figure 35. Stokes or creeping flow past a cylinder at Re = 0.16 simulated with LBM. Flow from left to right. Additional streamlines start near the cylinder to illustrate the absence of flow separation.

As the Reynolds number increases from a creeping flow regime, separation occurs and eddies form behind the disk. These elongate as the Reynolds number increases to approximately 50 (Figure 36).
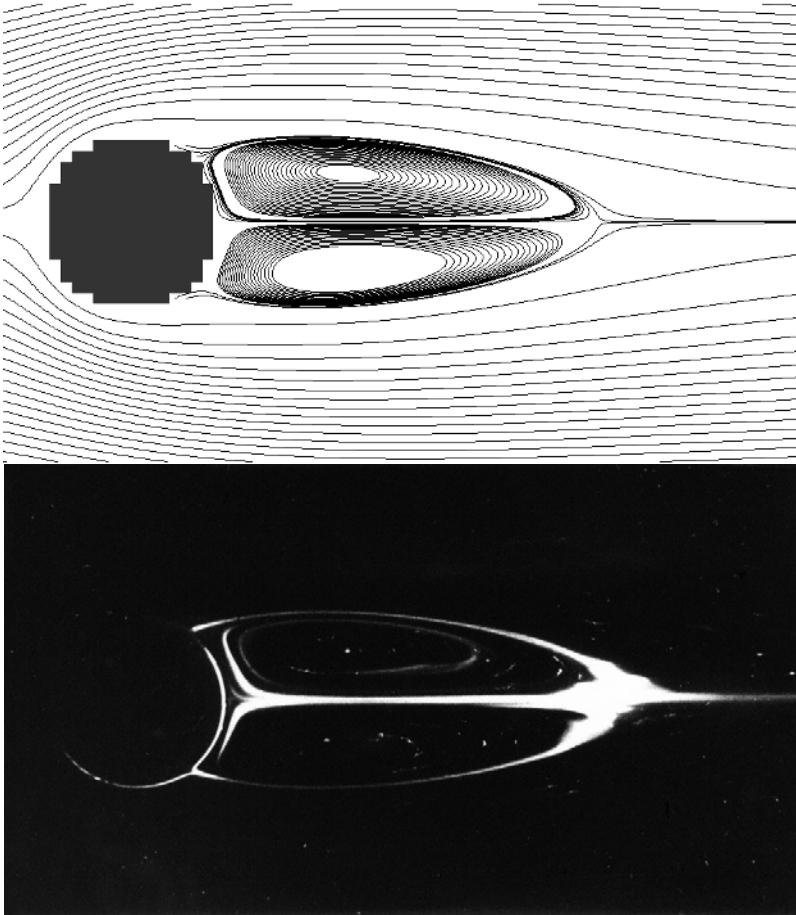
Figure 36. Eddies and cylinder wakes at Re ≈ 41. Top: LBM calculation; Bottom: (Taneda 1956 with permission.) Note that streamlines were intentionally started at positions that gave patterns like those in the photograph; the solution itself is symmetric.

5.3 Unsteady Flows at Higher Reynolds Numbers

Further increases in Reynolds number result in the development of un-steady flows in which eddies behind the disk break free alternately from opposite sides of the disk (vortex shedding) and move downstream as vor-tices. An expression that predicts the spacing of vortices in this 'von Kár-mán street' is (Douglas et al. 2001):

$$h/l = (1/\pi)\sinh^{-1}(l) = 0.281 \tag{53}$$

where h is the distance between the centers of the two 'lanes' and l is the distance between vortices in a lane. A photograph of a real flow is com-pared with an LBM simulation of a similar flow in Figure 37; there is qualitative agreement.

High Reynolds number flows are of great interest in many disciplines and are the subject of considerable LBM research. Succi (2001) includes a chapter that addresses turbulence modeling with LBM.
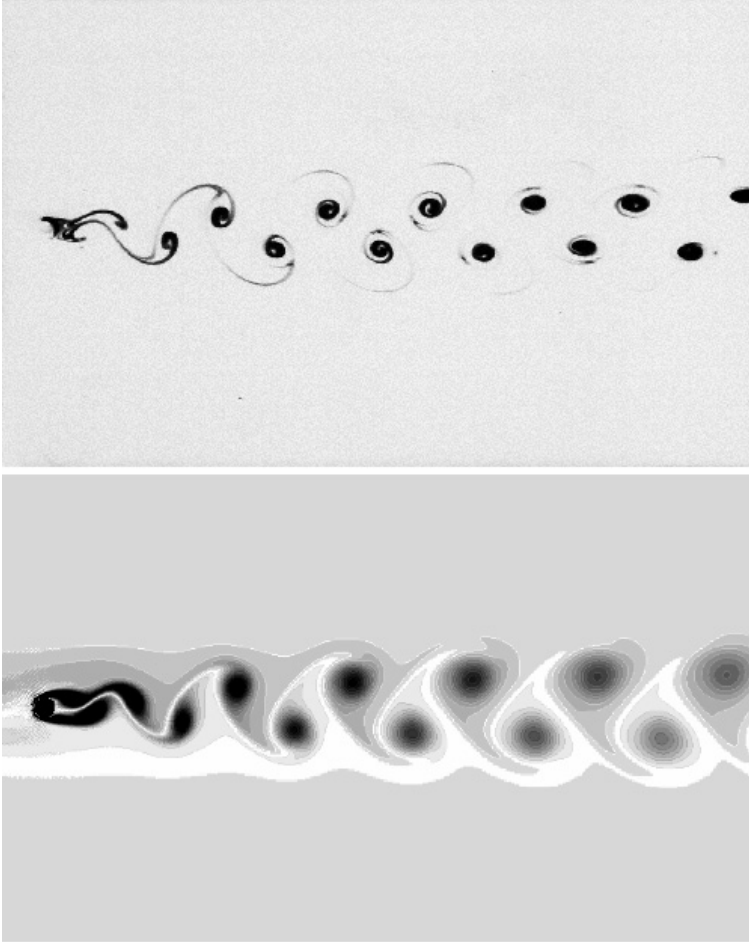
Figure 37. Top: Vortex shedding and von Kármán street at Re = 105. (Photograph by S. Taneda, with permission of S. Taneda and the Society for Science on Form, Japan; black and white inverted from original.) Bottom: Vorticity magnitude from LBM simulation at Re = 105.

5.4   Flows in More Complex Geometries

One of the most appealing aspects of LBM is that any geometric arrangement of obstacles and open areas can be easily incorporated into simple models that use bounceback. Our code reads the locations of obstacles

from bitmap files. Currently we support 24-bit 'truecolor' BMPs with black corresponding to obstacles and white signifying open space. The file naming convention is based on the lattice size and one pixel corresponds to 1 lattice unit. Thus, the input file for a domain 100 *lu* long in *x* by 25 *lu* in *y* would be '100x25.bmp'. Keep in mind the rule of thumb that 4 or 5 lattice units in an open flow channel are the minimum for the simulation of realistic hydrodynamics (Succi, 2001).

5.5 Exercises

1. Run a simulation of Poiseuille flow in a slit driven by gravity. Choose a slit width, fluid density, viscosity, and gravitational acceleration. Estimate the maximum and average velocities *a priori* using Eqs. (4) and (5) and ensure that they will not exceed 0.05 *lu ts^{-1}*. Compare the results to the analytical Poiseuille velocity profile.

2. Modify the input solids file to include a square obstacle like that in Figure 5 and induce separation. Note that a longer flow domain or different boundary conditions may be necessary to maintain approximately uniform velocities at the boundaries.

3. Look up the drag coefficient for a cylinder at a particular Reynolds number. Compute the drag force and design a LBM model to solve the flow field. MATLAB® or any program capable of generating streamlines or pathlines can be used to produce figures similar to those shown above.

4. Use constant velocity boundaries at $x = 0$ and $x = $ Lx. Choose a slit geometry, viscosity, and average velocity (equal to the boundary velocity) that will lead to a Reynolds number greater than 50. Specify all parameters and include units. Run the model and plot $u/u_{average}$ at 1/16, 1/8, 1/4, and 1/2 the total length of the domain in the *x* direction. Plot the expected Poiseuille velocity profile for fully developed flow on the same graph.

6 Single Component, Multiphase (SCMP) LBM

In the previous chapter we demonstrated that LBM is useful for simulating the flow of a single fluid. The true strengths of LBMs however lie in their ability to simulate multiphase fluids. Both single and multi-component multiphase fluids can be simulated. 'Component' refers to a chemical constituent such that a 'single component' (say H_2O) multiphase system would involve the liquid and vapor phases of water. These are particularly rich systems to consider as surface tension, evaporation, condensation, and cavitation are possible. Liquid-vapor behavior in partially saturated porous media can be simulated. In contrast, a multi-component system can consist of separate chemical components such as oil and water; such systems have been studied more extensively because of their economic importance. For example, Darcy's law-based relative permeability concepts for multicomponent oil/water-like systems have been investigated using LBM (Buckles et al. 1994; Soll et al. 1994; Martys and Chen, 1996; Langaas and Papatzacos, 2001).
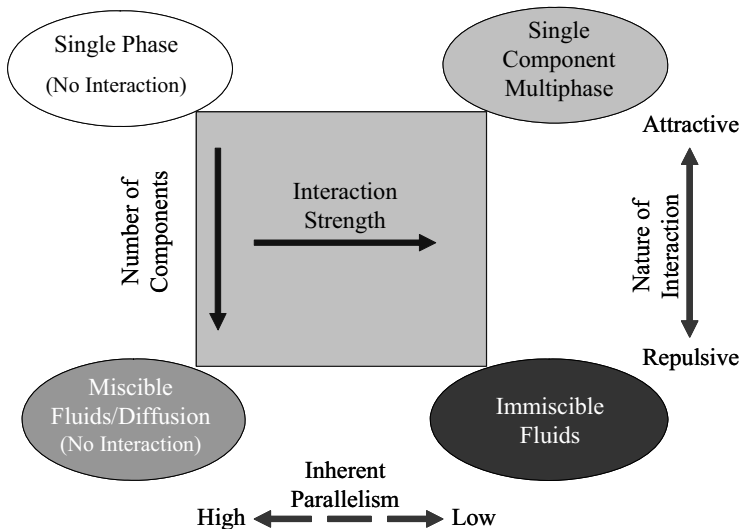
Figure 38. Conceptual framework for LBM models.

Figure 38 gives a conceptual framework for thinking about the various LBM models to be considered in this book. At the upper left we have the simple single fluid models discussed in the last two chapters: these involve a single chemical component whose molecules are not subjected to any 'long-range' interaction forces. Adding a long-range attractive force makes phase separation into a liquid and its own vapor possible as discussed below (upper right, single component multiphase).

If we add a second chemical component, we have the possibility of simulating completely miscible fluids (basically chemical solutions) in the absence of long range interactions (lower left), and completely immiscible fluids (oil and water for example) when there are long range repulsive interactions (lower right). Finally, note that the widely acclaimed inherent parallelism of LBM is lost when long range interactions are included.

This chapter focuses on Single Component Multiphase (SCMP) models. Early examples of lattice gas SCMP models can be found in Rothman (1988) and Appert and Zaleski (1990). The lattice Boltzmann implementation of these models began with Shan and Chen (1993, 1994). There are also so-called ''free energy'' approaches proposed by Swift et al. (1996), and ''finite density'' models that use the Enskog equation for dense gases (Luo 2000; He and Doolen 2002). Zhang and Chen (2003) have also proposed an approach based on tracking an energy (temperature) component. Such finite density or energy models seem to hold the key to the ultimate development of the LBM for practical applications due to the more realistic and consistent treatment of the equation of state that preserves the essential (molecular) physics of the process. Here we work with the Shan and Chen (1993, 1994) model extended for solids interactions. Although this model has numerous shortcomings, it is exceptionally versatile, and problems that have long defied quantitative treatment can now be examined.

First we recall basic physical chemistry theory relevant to these models. Then we provide details on incorporating long-range forces into the LBM model and on the resulting LBM equation of state. The determination of surface tension in the model is illustrated. Simulation of homogeneous and heterogeneous cavitation with the model are presented. Then interactions with surfaces are included and contact angles, capillary rise, adsorption, and capillary condensation are discussed.

The principal distinguishing characteristic of single component multi-phase LBMs is the incorporation of an attractive force between fluid 'particles'. This notion is part of the foundation for the famous van der Waals equation of state, which we now review.

6.1 Non-ideal Equation of State

The 'ideal' or 'perfect' gas law characterizes the behavior of gases at low density. Such gas laws are also known as Equations of State (EOS). The ideal gas law is commonly written as

$$PV = nRT \text{ or } P = \frac{nRT}{V} \tag{54}$$

where

- P is pressure (*atm*)
- V is volume (*L*)
- n is number of mols
- R is gas constant (0.0821 *L atm mol^{-1} K^{-1}*)
- T is temperature (*K*).

$V_m = V/n$ is the volume occupied by one mol of substance. The gas laws can be re-written to eliminate the number of mols n

$$P = \frac{RT}{V_m}. \tag{55}$$

Eqs. (54) and (55) are linear relationships between pressure and density (which is proportional to n/V).

The van der Waals EOS was developed to account for behaviors observed in real gases while retaining conceptual simplicity. It is given by

$$P = \frac{nRT}{V - nb} - a\left(\frac{n}{V}\right)^2. \tag{56}$$

The second term on the right accounts for attractive forces between molecules. Note that because a, n, and V are all positive, this term results in a reduction of the pressure relative to that of a perfect gas.

The $-nb$ term in the denominator accounts for the non-negligible volume of molecules. If the 'hard sphere', closest packed volume of one mol of molecules is b, then the minimum volume that can be occupied by n mols of molecules is nb. As the pressure increases, the volume of the gas V may approach nb. This will cause the denominator to approach zero and the pressure will rise very rapidly, effectively preventing further compression.

The van der Waals gas law can also be rewritten in terms of the molar volume:

$$P = \frac{RT}{V_m - b} - a\left(\frac{1}{V_m}\right)^2 \tag{57}$$

.

6.1.1 P–V_m, and P–ρ Presentations

The gas laws are typically presented graphically in one of two formats: in P–V space, pressure is plotted against volume while in P–ρ space, pressure is plotted against density. P–ρ plots are more intuitive and useful in general, but there is one very important manipulation that requires use of P–V plots.

Figure 39 shows the P–V_m plot for CO_2. In this plot, the perfect gas law is non-linear (like $y = 1/x$, because P is inversely proportional to V_m). The van der Waals EOS at various temperatures with parameters a and b taken from Atkins (1978) was used to plot the other curves. Temperatures were selected to illustrate supercritical, critical, and subcritical behaviors. At high temperature ($373K$) CO_2 is supercritical and no distinct liquid and vapor phases can be discerned. As the temperature is decreased, a critical temperature is reached; below this temperature phase separation into liquid and vapor is possible. The key difference in the EOS curves above and below the critical temperature is that above the critical temperature the curves decrease monotonically. Below the critical temperature the curves are no longer monotonic and this allows the coexistence of different molar volumes (different densities) of the substance at a single pressure.
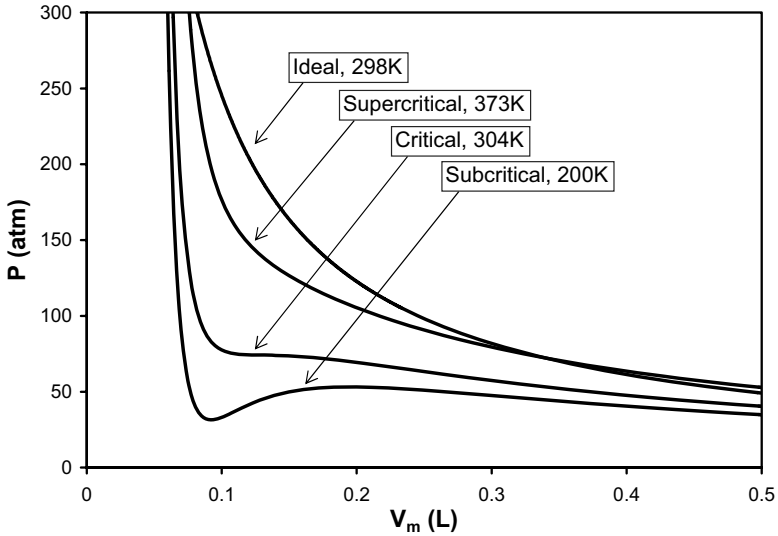
Figure 39. *P–V* plot of perfect and van der Waals equations of state for CO_2. van der Waals constants $a = 3.592 \; l^2 \; atm \; mol^{-2}$ and $b = 0.4267 \; l \; mol^{-1}$ (Atkins, 1978). $R = 0.0821 \; l \; atm \; mol \; K^{-1}$.

If we zoom in on a subcritical curve (Figure 40) we see that the EOS is intersected 3 times at the vapor pressure of CO_2 at 293K. Projecting the first intersection at small molar volume (high density) down to the *x*-axis gives the molar volume of the liquid. The second intersection is in what is referred to as a 'non-physical' portion of the EOS because the positive slope here indicates that increasing the pressure would cause the vapor to expand. The final intersection gives the molar volume of the equilibrium vapor phase. It is important to note that the vapor pressure applies only to flat liquid-vapor interfaces; we will explore the impact of curvature later.
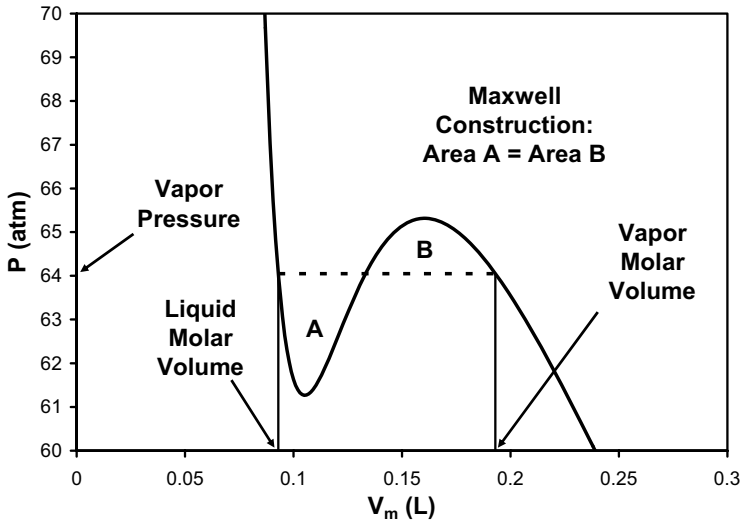
Figure 40. Detail of subcritical EOS for CO_2 illustrating liquid-vapor coexistence (293K). The molar volumes of the liquid and vapor and the vapor pressure can be determined from the Maxwell Construction.

6.1.2 Maxwell Construction and its Solution

In the context of LBM, the key reason for using P–V representations of the EOS is the Maxwell Construction. The Maxwell Construction allows the vapor pressure and the densities of the liquid and vapor phases to be found analytically when a functional form of the EOS is available. Boltzmann (1964/1995) discusses the Maxwell Construction and Figure 40 illustrates its use.

The Maxwell Construction can be stated as

$$\int_{V_{m,l}}^{V_{m,v}} P \, dV_m = P(V_{m,v} - V_{m,l}). \tag{58}$$

So the area under the curve (use caution if the EOS goes below $P = 0$) must equal the area of the rectangle defined by the liquid and vapor molar volumes and the vapor pressure. This is equivalent to specifying that the areas

A and B shown on Figure 40 must be equal. Despite its apparent simplicity, the Maxwell construction can be challenging to solve. One approach is simply to estimate the vapor pressure from the plot. For example, the integral PdV_m where P is given by the van der Waals equation (57) is

$$\int PdV_m = RT \ln(V_m - b) + \frac{a}{V_m}. \tag{59}$$

Based on inspection of the EOS curve in Figure 40, suppose we estimate a vapor pressure of 63 *atm*. For $T = 293.15K$ we can estimate $V_{m,l} = 0.095$ *liters* and $V_{m,v} = 0.206$ *liters* (by looking at a table or graph of values of $P(V_m)$ for example) and evaluate the integral between these limits to obtain a value of approximately 7.09. For $P = 63$ *atm* the right hand side of Eq. (58) is 63(0.206-0.095) = 6.99; comparing this with the 7.09 from the left hand side indicates that improvement is possible. Increasing the vapor pressure to 64 *atm* we can estimate $V_{m,l}$ and $V_{m,v} = 0.93$ and 0.194 *liters* respectively. Then the integral value is 6.45 and the right hand side is 6.46, which is considerably better than the initial guess. Further refinements can be made. We will revisit the Maxwell Construction after the LBM EOS is introduced.

6.1.3 EOS for Water/Water Vapor and *P–ρ* Presentations

Water is of particular interest to a broad range of scientists and engineers but it is also more complex than can be adequately described by the standard van der Waals EOS. We explore water in some additional detail here and use it to introduce P–ρ plots.

Figure 41 shows the perfect gas and van der Waals equations of state for water at 298K. There are significant differences between the CO_2 and water equations of state. Perhaps most striking is that the curve extends into strong negative pressures. Figure 42 shows the same information plotted as $P(\rho)$. The virtues of such a plot are immediately apparent. The perfect gas law plots as a straight line through the origin and the low and high density (vapor and liquid) portions of the curve are clear and in familiar units.
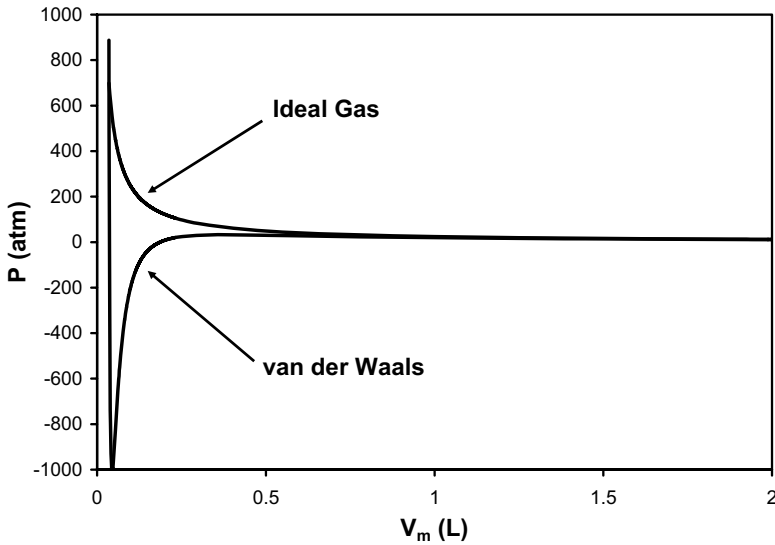
Figure 41. *P–V_m representation of perfect and van der Waals EOS for water.*

It is also clear however that the van der Waals equation fails to quantitatively reproduce the known density (1000 $kg\ m^{-3}$) of water. Finally, the known compressibility of liquid water is not well matched. Truskett et al. (1999) incorporated directional hydrogen bonds in a model to estimate the water EOS and achieved good success (Figure 43).

One additional feature of the non-linear equations of state discussed here is the bottom of the valley; it is referred to as the *spinodal* and represents the maximum tension that a pure liquid can sustain. So, referring to Figure 43 for example, we can imagine beginning with pure liquid water in a cylinder at its familiar density of 1000 $kg\ m^{-3}$ and then withdrawing a piston that stretches the water to a density somewhat less than 800 $kg\ m^{-3}$. The pressure in the water will follow the EOS and be approximately -2000 *atm*. If the density is reduced beyond the minimum of the EOS curve, the liquid will catastrophically phase separate into liquid and vapor in the process known as *cavitation*. For pure liquids this will occur at the spinodal and is called homogeneous cavitation. Heterogeneous cavitation is much more common and occurs at much lower tension when the structure of the liquid is disrupted by pre-existing bubbles or particles. We will examine both homogeneous and heterogeneous cavitation with LBM below.
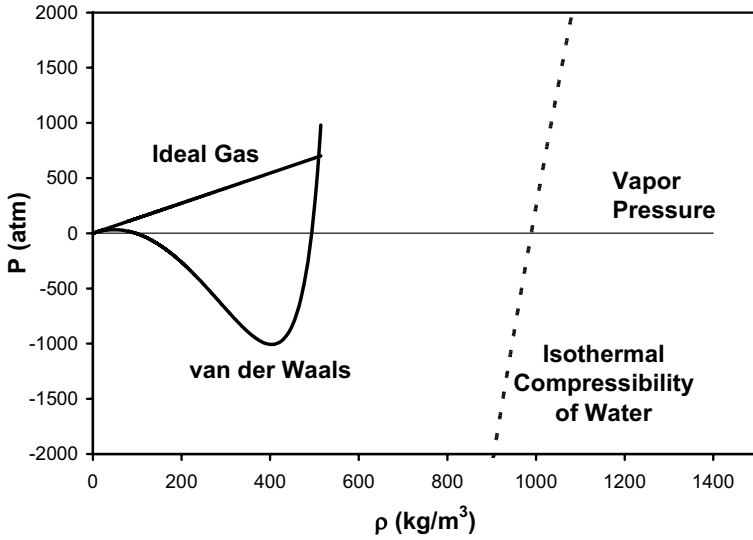
Figure 42. *P–ρ* representation of perfect and van der Waals EOS for water at 298K. There is a gross underestimation of the density of liquid water.
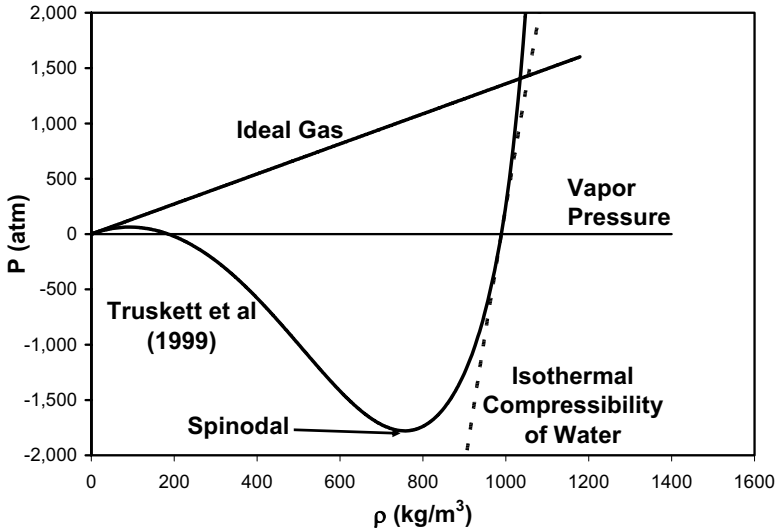
Figure 43. Truskett et al. (1999) EOS for water showing the spinodal. Incorporating directional hydrogen bonding allows good matching of the key features of water behavior.

6.2   Interparticle Forces and their Incorporation into LBM

Lattice Boltzmann models presented thus far in this book were based exclusively on streaming and collision. To simulate multiphase fluids, we need long range interactions between fluid 'particles'. For our purposes interactions with nearest neighbor particle densities f will be sufficient to simulate the basic phenomena of multiphase fluid interactions. For single component multiphase fluids (e.g., water/water vapor) an attractive (cohesive) force F between nearest neighbor fluid particles is needed and, for the D2Q9 model, is induced as follows:

$$\mathbf{F}(\mathbf{x},t) = -G\psi(\mathbf{x},t)\sum_{a=1}^{8} w_a \psi(\mathbf{x}+\mathbf{e_a}\Delta t,t)\mathbf{e_a} \qquad (60)$$

where G is the interaction strength, w_a is 1/9 for $a = \{1, 2, 3, 4\}$, is 1/36 for $a = \{5, 6, 7, 8\}$, and ψ is the interaction potential:

$$\psi(\rho) = \psi_0 \exp\left(-\rho_0 \big/ \rho\right). \qquad (61)$$

ψ_0 and ρ_0 are arbitrary constants. This interaction potential is special in that its ''…behavior is consistent with that of an isothermal process...'' (Shan and Chen 1994; see also He and Doolen 2002). According to Shan and Chen (1993), the interaction potential function must be monotonically increasing and bounded. Other forms of the interaction potential are commonly used and include for example $\psi(\rho) = \rho_0[1-\exp(-\rho/\rho_0)]$ (Shan and Chen 1993; Raiskinmäki et al. 2000 and 2002; Hyväluoma et al. 2004), $\psi(\rho) = \rho$ (Martys and Chen 1996; Pan et al. 2004), and $\psi(\rho) = g\rho_0^2\rho^2/[2(\rho_0+\rho)^2]$ (Qian et al. 1995).

Figure 44 shows the Eq. (61) interaction potential function with $\psi_0 = 4$ and $\rho_0 = 200$. These parameters were selected arbitrarily, but will be used consistently in SCMP LBM simulations in this book for convenience because the model behavior with these values has been explored more thoroughly. $G < 0$ for attraction between particles and the force is stronger when the density is higher. Thus, dense regions (liquid) experience a stronger cohesive force than vapor, which leads to surface tension phenomena.
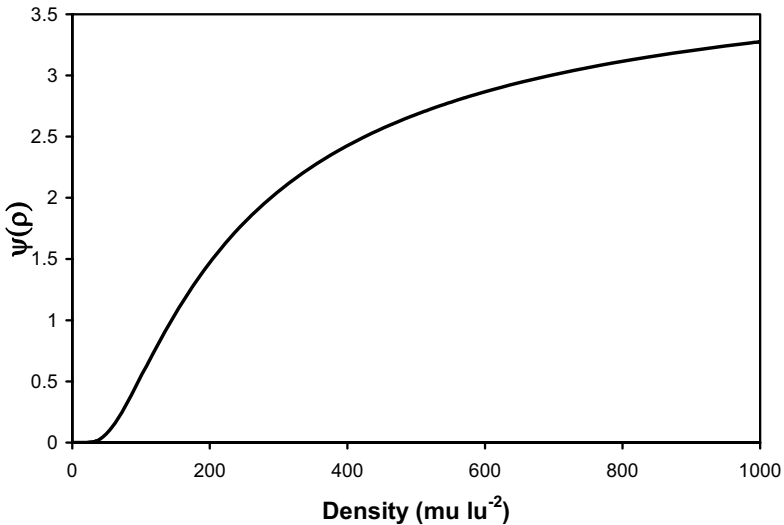
Figure 44. 'Isothermal' interaction potential function of Shan and Chen (1994) with $\psi_0 = 4$ and $\rho_0 = 200$. Other forms of the interaction potential are in common use. We use 2-D densities and pressures throughout the text.

The attractive force is included in the model the same way that gravity was incorporated earlier (Eqs. (45), (46), and (47)).

It is important to note that we have incorporated only the molecular attraction aspect of the van der Waals gas model described above; the repulsive forces that dominate the van der Waals gas model when a gas is compressed to near its 'hard sphere' volume are neglected in this simplest of SCMP models. Enskog developed theory for dense gases that has been incorporated in more advanced LBM models (e.g., He and Doolen, 2002; Martys 2001; Luo 2000; Luo, 1998). Ignoring these effects here has important ramifications for the equations of state of the simulated gases that will be introduced below. At the present time, the ability of LBM to simulate any desired EOS is severely limited. The work of He and Doolen (2002) may hold the key to resolving EOS limitations (and also allow non-isothermal flows) but requires decoupling of velocity and space discretizations and has not been explored. The approach of Zhang and Chen (2003) may be similarly profitable. Nevertheless, the simple Shan and Chen (1993, 1994) model is capable of simulating rich SCMP behaviors includ-

ing surface tension/capillarity, evaporation, condensation, and homogeneous and heterogeneous cavitation.

6.2.1 The SCMP LBM EOS

Application of Eqs. (60) and (61) leads to a non-ideal EOS for the simulated fluids (He and Doolen, 2002):

$$P = \rho RT + \frac{GRT}{2}[\psi(\rho)]^2 . \tag{62}$$

The first term on the right hand side is the ideal gas law, which applies to the single component, single phase model discussed above. The value of RT is fixed for both the SCSP and SCMP models:

$$RT = \frac{1}{3}. \tag{63}$$

The second term on the right hand side of (62) is the non-ideal part that accounts for the attractive force between the molecules and leads to a reduction in pressure (when $G < 0$) and the non-linear form of the EOS. When G is adequately negative that the EOS is subcritical (non-monotonic), phase separation can occur.

After incorporating RT, Eq. (62) becomes

$$P = \frac{\rho}{3} + \frac{G}{6}\Psi^2(\rho). \tag{64}$$

This is plotted for a series of G values in Figure 45. This EOS is qualitatively similar to the van der Waals EOS and is useful in simulating Laplace Law and capillary phenomena. The lack of a repulsive force in the model however has led to unfortunate (and generally not physically correct) behavior where the liquid phase is actually more compressible than the vapor phase. (The slope $dP/d\rho$ of the curve is lower in the liquid density region than it is in the vapor region.) This does not affect equilibrium liquid-vapor configurations but makes certain types of simulations more challenging and one always needs to be aware of it.
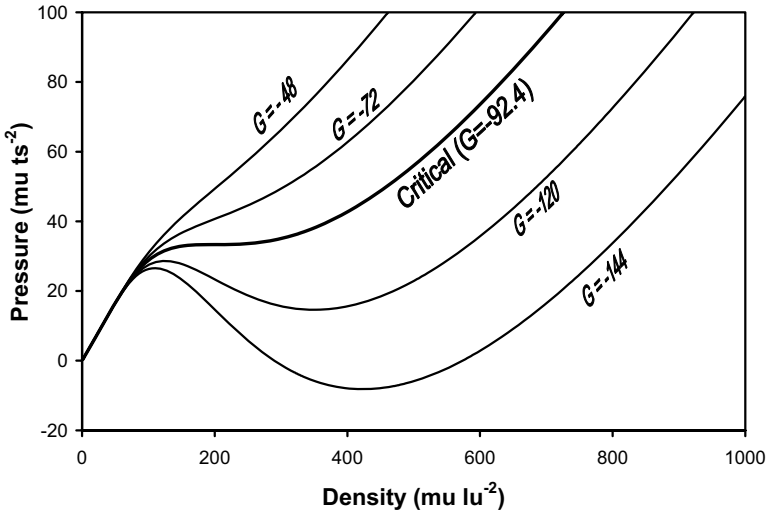
Figure 45. SCMP EOS for $\psi_0 = 4$ and $\rho_0 = 200$ at 5 values of G.

Eqs. (60) and (61) are implemented as follows:

```
// Compute psi, Eq. (61).
for( j=0; j<LY; j++)
  for( i=0; i<LX; i++)
    if( !is_solid_node[j][i])
      {
        psi[j][i] = 4.*exp( -200. / ( rho[j][i]));
      }

// Compute interaction force, Eq. (60) assuming periodic domain.
for( j=0; j<LY; j++)
{
  jp = ( j<LY-1)?( j+1):( 0   );
  jn = ( j>0    )?( j-1):( LY-1);

  for( i=0; i<LX; i++)
  {
    ip = ( i<LX-1)?( i+1):( 0   );
    in = ( i>0    )?( i-1):( LX-1);

    Fx = 0.;
    Fy = 0.;

    if( !is_solid_node[j][i])
    {
      Fx+= WM*ex[1]*psi[j ][ip];
      Fy+= WM*ey[1]*psi[j ][ip];
      Fx+= WM*ex[2]*psi[jp][i ];
```

```
      Fy+= WM*ey[2]*psi[jp][i ];
      Fx+= WM*ex[3]*psi[j ][in];
      Fy+= WM*ey[3]*psi[j ][in];
      Fx+= WM*ex[4]*psi[jn][i ];
      Fy+= WM*ey[4]*psi[jn][i ];
      Fx+= WD*ex[5]*psi[jp][ip];
      Fy+= WD*ey[5]*psi[jp][ip];
      Fx+= WD*ex[6]*psi[jp][in];
      Fy+= WD*ey[6]*psi[jp][in];
      Fx+= WD*ex[7]*psi[jn][in];
      Fy+= WD*ey[7]*psi[jn][in];
      Fx+= WD*ex[8]*psi[jn][ip];
      Fy+= WD*ey[8]*psi[jn][ip];

      Fx = -G * psi[j][i] * Fx;
      Fy = -G * psi[j][i] * Fy;
    }
  }
}
```

Beware of the assumption about a periodic domain (see the note in the comment at the top of the code snippet). When using boundary conditions, the interaction force must be computed differently at those boundaries because the density is not necessarily continuous across the periodicity then. Hence, the three terms in Eq. (60) associated with nodes across the periodic boundary are unavailable. We find that replacing those three terms with duplicates of the three terms in the other direction (towards the interior of the domain instead of across the boundary of the domain) gives good results.

6.3 Phase (Liquid-Vapor) Separation and Interface Minimization

The preceding model development is enough to simulate phase separation and its dynamics. In Figure 46, we show the results of a simulation initialized with an average density of 200 $mu\ lu^{-2}$ with a random variation incorporated via the 'static' initial condition included in our code. For the standard parameter values ($\psi_0 = 4$ and $\rho_0 = 200$) and ψ function we adopt, this initial density falls on the negatively-sloped, non-physical part of the $G = -120$ EOS (Figure 45) and hence is unstable and leads to phase separation. In this case, the phase separation ultimately leads to a single droplet in a vapor atmosphere. Whether liquid drops or vapor bubbles are formed depends on the total mass in the domain and consequently on the initial density selected.
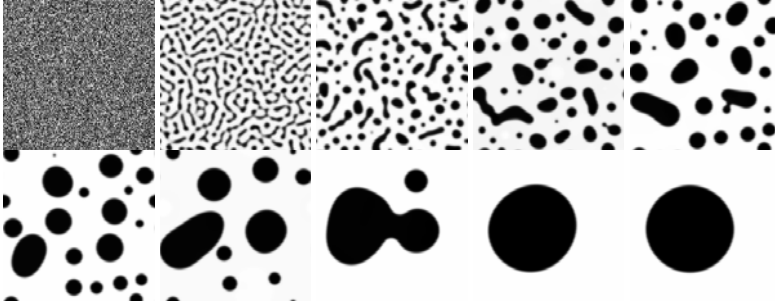
Figure 46. Time series of liquid-vapor phase separation dynamics in a 200×200 *lu²* domain. Gray scale proportional to normalized density. $G = -120$, $\psi_0 = 4$, $\rho_0 = 200$, and $\tau = 1$. Initial density 200 *mu lu⁻²* plus a random number in the interval [0,1]. Results shown at time = 0, 100, 200, 400, 800, 1600, 3200, 6400, 12800, and 25600 *ts*.

When phase separation occurs, there is a strong tendency for the interfaces formed to minimize their total area (or length in 2-D). This is a straightforward consequence of free energy minimization and occurs in part by geometric rearrangement into the minimum surface area volume (a sphere or in 2-D, a circle). Depending on the initial conditions, this rearrangement may also involve a significant amount of coalescence of 'blobs' of each phase. In liquid-vapor systems, there can also be condensation and evaporation; bubbles can simply fill in or grow at the expense of mass elsewhere in the domain. The relatively high vapor density in the simple simulations described in this book indicates a potential excess of transport in the vapor phase relative to many real liquid-vapor systems.

6.3.1 Spurious Interface Velocities

Examination of the velocity fields in SCMP simulations reveals high velocities perpendicular to phase boundaries. These are non-physical. They are present at equilibrium and there appears to be no mass exchange associated with them. Some papers that focus on or mention these velocities include Wagner (2003), Lishchuk et al. (2003), Nourgaliev et al. (2003), and Raiskinmäki et al. (2000).

6.3.2 Estimating Surface Tension

The surface tension can be estimated simply by simulating a series of drops and bubbles of various sizes and measuring their radii and inside/outside densities. The densities must be converted to pressures via the EOS and the difference ΔP computed. Then the slope of a plot of $1/radius$ vs. ΔP will be the surface tension in accordance with the Laplace law (Eq. (7)). Figure 47 presents results leading to a surface tension of 14.3 *lu mu ts*$^{-2}$. Readers are encouraged to develop such a plot on their own (see Exercises).
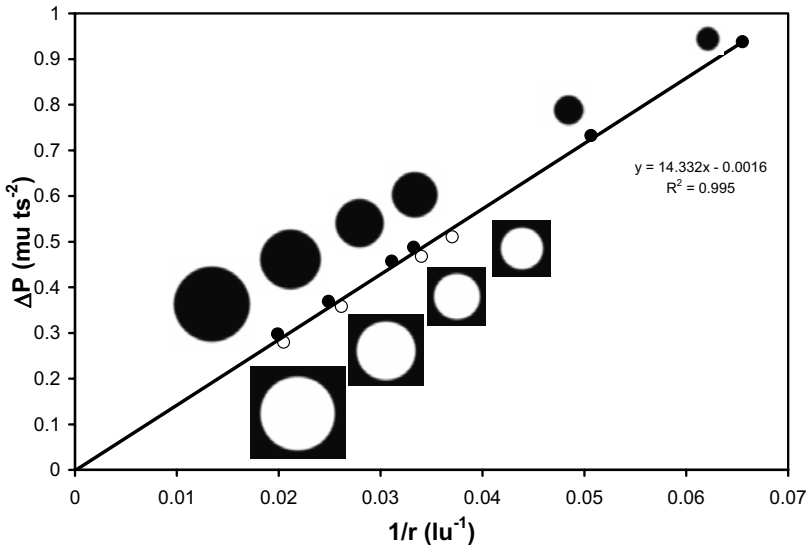
Figure 47. Plot of curvature (1/*r*) vs. pressure difference used to determine surface tension in SCMP lattice Boltzmann model. Simulated drops and bubbles shown adjacent to data points at relative scale. Bubbles consistently fall below line and drops are consistently above; this may be due to the selection of a density cutoff value for measuring radii.

6.3.3 Flat Interfaces: Maxwell Construction for SCMP LBM

Curved interfaces are common and the relationship between radii of curvature and pressure differential across them can be predicted when the surface tension is known. Flat interfaces are exceptionally important however because the vapor pressure above them at equilibrium is the saturation va-

por pressure, which is used in scaling many vapor pressure-dependent processes. At equilibrium, the pressure difference across a flat interface must be zero, which is consistent with the Laplace law, and the Maxwell Construction should apply. We have already illustrated a trial-and-error approach to solving the Maxwell Construction for the van der Waals equation of state. From more sophisticated numerical solutions of the Maxwell Construction on the SCMP LBM EOS with $\psi_0 = 4$, $\rho_0 = 200$, and $G = -120$, we compute $\rho_l = 514.64$ *mu lu^{-2}* and $\rho_v = 79.705$ *mu lu^{-2}*. These values differ appreciably from those observed in flat interface simulations that yield $\rho_l = 524.39$ *mu lu^{-2}* and $\rho_v = 85.704$ *mu lu^{-2}*. The observed vapor and liquid densities give very similar pressures of 25.5599 and 25.5605 respectively from the EOS, while the optimal Maxwell solution densities lead to pressures of 24.45172 and 24.45166. The reasons for this discrepancy are unknown and this represents a significant outstanding problem from our perspective. It may account wholly or in part for the less than quantitative results we obtain in certain types of simulations.

6.4 Cavitation

Cavitation is a catastrophic transition from liquid to vapor. Cavitation can be either 'homogeneous' when it occurs at the limit of the pure liquid's tensile strength, or 'heterogeneous' when it is nucleated by preexisting bubbles or other disruptions in the structure of the liquid. Single component multiphase lattice Boltzmann methods offer a virtual laboratory for investigation of cavitation.

Sukop and Or (2005) conducted LB simulations of cavitation in a two-dimensional geometry; they adapted Or and Tuller's (2002) discussion of three-dimensional critical bubble radius and energy cost for heterogeneous cavitation in 2-D. The energy cost for the creation of a vapor bubble is the sum of the interfacial energy needed to create the bubble and the work of negative pressure over the bubble area; that is,

$$\Delta E = 2\pi r \sigma + \pi r^2 \Delta P \tag{65}$$

with σ = (2-D) surface tension (MLT^{-2}), ΔP = (2-D) pressure (MT^{-2}), and r = bubble radius. This relationship is plotted in Figure 48. The energy cost is maximized at $r^* = -\sigma/\Delta P$. For any given tension applied to the LB system, r^* represents a critical bubble radius. A bubble with a radius less than r^* will be lost to condensation rather than result in cavitation because con-

densation is more energetically favorable. Once a bubble is large enough to overcome the energy barrier, cavitation is the more favorable outcome.
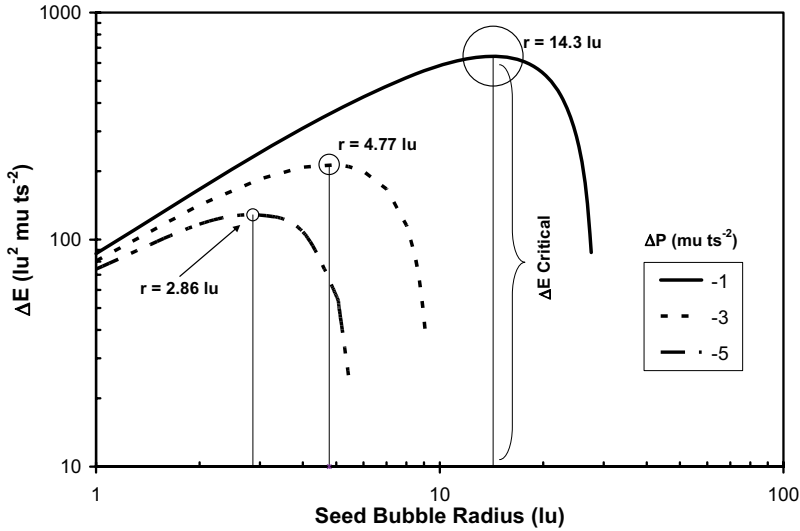
Figure 48. Energy barrier as a function of seed bubble radius for different initial liquid pressures. Circle size proportional to critical radii. $\psi_0 = 4$, $\rho_0 = 200$, and $G = -120$. (Sukop and Or, 2005)

6.4.1 Homogeneous Cavitation

Sukop and Or (2005) simulated spinodal decomposition (cavitation via homogeneous nucleation) in a 200×200 lu^2 domain. Figure 49 shows the process as the change in density at two points in the domain during the simulation. The initial density in the liquid is 400 $mu\ lu^{-2}$ (i.e., the liquid is under tension at the outset), but this simply hastens the process. The left and right boundaries are periodic, and constant velocity boundaries (0.005 $lu\ ts^{-1}$) at the top and bottom of the domain pull liquid from the domain. This results in increasing tension in the liquid with time until the liquid spinodal pressure and density are reached. Then the liquid cavitates catastrophically. Cavitation occurs as a linear band (lower left inset of Figure 49) in this simulation because there is no randomness in the pressure/density distribution in the domain and the vertical domain boundaries are periodic. In real systems, small fluctuations would result in random preferred loci of cavitation and bubbles would form. Severe den-

sity/pressure fluctuations occur in the liquid phase after the onset of cavitation; these eventually subside and liquid is present at density very close to its equilibrium, flat, free interface value of 524 *mu lu*$^{-2}$. Smaller fluctuations in the vapor phase are rapidly damped and a final density very close to the equilibrium value (85 *mu lu*$^{-2}$) is attained. The relative magnitudes of these fluctuations are consistent with the EOS and a greater compressibility in the liquid phase.
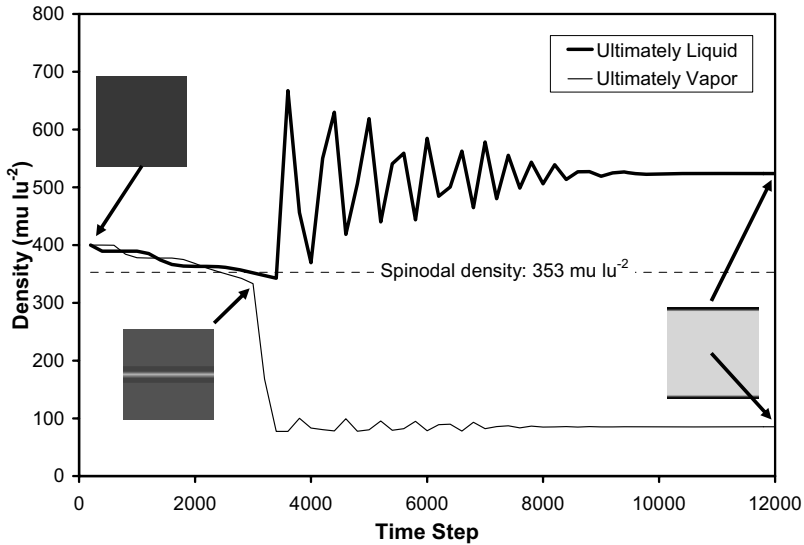
Figure 49. Density as a function of time for two points inside a 200 × 200 *lu*^{2} domain. The fine line follows density at a point in the domain (100,100) that ultimately becomes vapor, while the heavy line gives the density near the boundary at a point (100,1) that is ultimately liquid. Insets show density distributions in domain. (Sukop and Or, 2005)

6.4.2 Heterogeneous cavitation

Sukop and Or (2005) also produced results that closely match heterogeneous cavitation theory (Or and Tuller, 2002). Periodic boundaries were used on the vertical edges of the 200×200 *lu*^{2} domain and pressure boundaries were applied on the top and bottom. Based on the equation of state (Figure 45), the density of the vapor and liquid phases of the fluid can be calculated for any given pressure. A density corresponding to a liquid pressure below that of the experimental flat free interface was used on both bounda-

ries. (The difference between the boundary pressure and the pressure of a flat, free liquid-vapor interface is represented by ΔP in Figure 48.) These boundary conditions effectively stretch the fluid, creating a situation favorable for cavitation.

Sukop and Or (2005) inserted vapor bubbles of various sizes into their simulations. As shown in Figure 48, a bubble with radius just below the critical value cannot overcome the energy barrier and eventually condenses. However, a bubble with radius just above the critical value acts as a seed for cavitation. The LBM's proper simulation of these phenomena at two different liquid tensions is demonstrated in the following figures. In Figure 50, for $\Delta P = -1$ *mu ts*$^{-1}$, the critical radius is 13.85 *lu* and the radii of the seed bubbles are 12 and 15 *lu*.
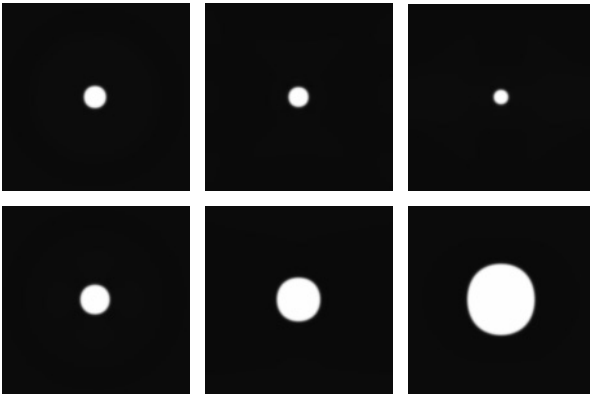
Figure 50. Effect of initial seed bubble size on evolution for $\Delta P = -1$ *mu ts*$^{-2}$. Top: $r = 12$ *lu* bubble dissipates due to condensation. Bottom: $r = 15$ *lu* bubble nucleates cavitation.

Similarly, in Figure 51, for $\Delta P = -5$ *mu ts*$^{-1}$ the critical radius is 2.77 *lu* and the radii of the seed bubbles shown are 2 and 4 *lu*. Clearly, much smaller seed bubbles can nucleate cavitation at higher liquid tensions (more negative ΔP).
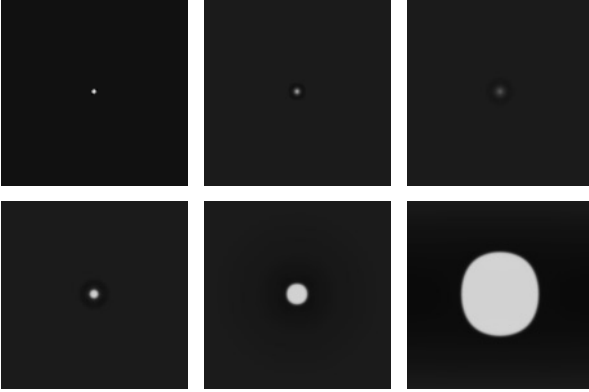
Figure 51. Effect of initial seed bubble size on evolution for ΔP = -5 *mu ts^{-2}*. Top: *r* = 2 *lu* bubble dissipates due to condensation. Bottom: *r* = 4 *lu* bubble nucleates cavitation.

6.5 SCMP LBM with Surfaces

6.5.1 Fluid-Surface Forces

In order to extend our capability to include simulation of SCMP fluids in porous media and other containers, it is essential that we incorporate an adhesive interaction between fluid particles and surfaces. The original method is due to Martys and Chen (1996) and is elegant in its simplicity. The idea is to create an analogue to the particle-particle interaction force used to induce phase separation. The only difference is that instead of summing the ψ functions of neighboring nodes (Eq. (60)), we sum an indi-cator variable denoting a solid. The strength of the force contribution is specified by a G_{ads} 'adsorption' coefficient. The equation describing this is

$$\mathbf{F}_{ads}(\mathbf{x},t) = -G_{ads}\psi(\mathbf{x},t)\sum_a w_a s(\mathbf{x} + \mathbf{e}_a\Delta t)\mathbf{e}_a , \qquad (66)$$

where *s* is a 'switch' that takes on value one (1) if the site at $\mathbf{x} + \mathbf{e}_a\Delta t$ is a solid and is zero (0) otherwise. The w_a are the same direction-dependent weighting factors used before. The C implementation of Eq. (66) is straightforward. The `is_solid_node` array is used to evaluate if a node neighboring the fluid node under consideration is a solid; if so, a contribu-tion is made to the surface force.

```
// Compute psi, Eq. (61).
for( j=0; j<LY; j++)
  for( i=0; i<LX; i++)
    if( !is_solid_node[j][i])
    {
      psi[j][i] = 4.*exp( -200. / ( rho[j][i]));
    }

// Compute interaction force, Eq. (66).
for( j=0; j<LY; j++)
{
  jp = ( j<LY-1)?( j+1):( 0   );
  jn = ( j>0   )?( j-1):( LY-1);

  for( i=0; i<LX; i++)
  {
    ip = ( i<LX-1)?( i+1):( 0   );
    in = ( i>0   )?( i-1):( LX-1);

    if( !is_solid_node[j][i])
    {
      sum_x=0.;
      sum_y=0.;

      if( is_solid_node[j ][ip]) // neighbor 1
      { sum_x = sum_x + WM*ex[1];
        sum_y = sum_y + WM*ey[1]; }
      if( is_solid_node[jp][i ]) // neighbor 2
      { sum_x = sum_x + WM*ex[2];
        sum_y = sum_y + WM*ey[2]; }
      if( is_solid_node[j ][in]) // neighbor 3
      { sum_x = sum_x + WM*ex[3];
        sum_y = sum_y + WM*ey[3]; }
      if( is_solid_node[jn][i ]) // neighbor 4
      { sum_x = sum_x + WM*ex[4];
        sum_y = sum_y + WM*ey[4]; }
      if( is_solid_node[jp][ip]) // neighbor 5
      { sum_x = sum_x + WD*ex[5];
        sum_y = sum_y + WD*ey[5]; }
      if( is_solid_node[jp][in]) // neighbor 6
      { sum_x = sum_x + WD*ex[6];
        sum_y = sum_y + WD*ey[6]; }
      if( is_solid_node[jn][in]) // neighbor 7
      { sum_x = sum_x + WD*ex[7];
        sum_y = sum_y + WD*ey[7]; }
      if( is_solid_node[jn][ip]) // neighbor 8
      { sum_x = sum_x + WD*ex[8];
        sum_y = sum_y + WD*ey[8]; }

      sforce_x[j][i] = -Gads * psi[j][i] * sum_x;
      sforce_y[j][i] = -Gads * psi[j][i] * sum_y;
    }
  }
}
```

We seem to have flexibility in the choice of the pre-sum factor $\psi(\mathbf{x},t)$; Martys and Chen (1996) originally used $\psi = \rho$, while Raiskinmäki et al. (2000; 2002) and Hyväluoma et al. (2004) used the ψ function $\psi(\rho) = 1 - \exp(-\rho)$. For consistency with the cohesion force, we use the ψ function Eq. (61) here.

Next we introduce a few studies that used the same basic Shan and Chen model we develop in this book and then give more detailed discussions of various phenomena in the following sections. Raiskinmäki et al. (2000) considered the spreading dynamics of three-dimensional droplets on surfaces with the Shan and Chen SCMP lattice Boltzmann model. They showed that the simulated rate of droplet spreading on a smooth surface was consistent with Tanner's Law, which states that the radius of the liquid/surface interface increases as a power function of time, $r \sim t^q$.

Raiskinmäki et al. (2002) investigated capillary rise dynamics using the same model. This work showed that the Washburn Equation, which predicts the rate of capillary rise, was satisfied by the SCMP LBM simulations. These authors concluded that relatively large capillary tubes – at least 30 lattice units across – would be needed to properly simulate the hydrodynamics. They also noted a discrepancy between the analytical Poiseuille velocity profile and the velocities of the liquid phase in an SCMP simulation; apparently the cohesive and adhesive forces imposed on the liquid affect its properties. Additional work needs to be done to assess this phenomenon.

Hyväluoma et al. (2004) also used the Shan and Chen SCMP LBM in an investigation of mercury intrusion porosimetry. In a similar study, Sukop and Or (2004) simulated contact angles, adsorption, capillary condensation, and wetting and drying of angular pores.

6.5.2 Contact Angles

Varying the G_{ads} parameter allows simulation of the complete range of contact angles. In Figure 52, we show SCMP simulations that yield 3 special contact angles: 0, 90, and 180 degrees.

We compute the necessary G_{ads} parameter for the three special contact angles by balancing the cohesive and adhesive forces in different ways. Assume that we are at points of either pure liquid or pure vapor. Then the

point's neighbors all have the same density as found at the point itself. From Eq. (60), the forces can then be written as

$$\mathbf{F} = -G\psi^2 \sum_{a=1}^{8} w_a \mathbf{e}_a \,, \tag{67}$$

where the ψs are all equal and can be combined. So, for the vapor and liquid phases respectively, we have

$$\mathbf{F}^v = -G\psi_v^2 \sum_{a=1}^{8} w_a \mathbf{e}_a \tag{68}$$

and

$$\mathbf{F}^l = -G\psi_l^2 \sum_{a=1}^{8} w_a \mathbf{e}_a \,. \tag{69}$$

A fluid node completely surrounded by solid surfaces would experience a different force. From Eq. (66) we obtain for vapor and liquid respectively:

$$\mathbf{F}_{ads}^v = -G_{ads}\psi_v \sum_{a=1}^{8} w_a \mathbf{e}_a \tag{70}$$

and

$$\mathbf{F}_{ads}^l = -G_{ads}\psi_l \sum_{a=1}^{8} w_a \mathbf{e}_a \,. \tag{71}$$

Note that the ψ factors are not squared here because the indicator variable s appears inside the summation rather than the ψ values.

We also consider the forces at points that have the average ψ value to represent the interface between liquid and vapor:

$$\overline{\mathbf{F}} = -G\overline{\psi}^2 \sum_{a=1}^{8} w_a \mathbf{e}_a \tag{72}$$

and

$$\overline{\mathbf{F}}_{ads} = -G_{ads}\overline{\psi}\sum_{a=1}^{8}w_a\mathbf{e}_a\,, \tag{73}$$

where

$$\overline{\psi} = \tfrac{1}{2}\left(\psi_\ell + \psi_v\right). \tag{74}$$

Note that all of the equations [(68) through (73)] contain the same summation term.

On a surface that is completely wetted by the liquid yielding a contact angle of 0 degrees, the adhesive force between the solid and the liquid is equal to the cohesive force of the liquid. Setting

$$\mathbf{F}^\ell = \mathbf{F}^\ell_{ads} \tag{75}$$

gives

$$-G\psi_l^2\sum_{a=1}^{8}w_a\mathbf{e}_a = -G_{ads}\psi_l\sum_{a=1}^{8}w_a\mathbf{e}_a\,, \tag{76}$$

which reduces to

$$G_{ads} = G\psi_l\,, \tag{77}$$

For the usual liquid density (albeit a flat free interface value) of 524.39 *mu lu^{-2}*, $\psi = 2.7316$. The cohesion parameter has been selected as $G = -120$, so $G_{ads} = -327.79$. It is easy to check if this is reasonable with simulations. Figure 52 shows the results.

Figure 52. Simulation of 0, 90, and 180 degree contact angles based on the force balance rationale discussed in the text. For the cohesive interaction parameter $G = -120$ and the ψ function parameters $\psi_0 = 4$ and $\rho_0 = 200$, $G_{ads} = -327.79$, -187.16, and -46.534 respectively.

On a surface that is wetted by the liquid to an extent exactly between completely wettable and completely non-wettable for a contact angle of 90°, the adhesive force of the solid on the interface between liquid and vapor is equal to the cohesive force at the liquid-vapor interface. Setting

$$\overline{\mathbf{F}} = \overline{\mathbf{F}}_{ads} \tag{78}$$

leads to

$$G_{ads} = G\overline{\psi} = G\frac{(\psi_l + \psi_v)}{2}, \tag{79}$$

which gives $G_{ads} = -187.16$ for $G = -120$, $\psi_l = 2.7316$, and (with $\rho_v = 85.70$ $mu\ lu^2$) $\psi_v = 0.38774$.

Finally, on a surface that is completely non-wettable by the liquid for a contact angle of 180 degrees, the adhesive force between the solid and the vapor must equal the cohesive force of the vapor. Setting

$$\mathbf{F}^v = \mathbf{F}_{ads}^v \tag{80}$$

gives $G_{ads} = -46.534$ by the same procedure used for the 0-degree contact angle except that $\psi_v = 0.38774$ is the proportionality constant based on $\rho_v = 85.70\ mu\ lu^{-2}$.

Readers should carry out similar simulations (see Exercises). Our experience suggests that it is easily possible to create zones of excessive density adjacent to surfaces with the magnitude of G_{ads} too large. This is probably another consequence of the lack of a repulsive interaction between the particles. The approach for selecting G_{ads} suggested here should make the process more rational than the trial-and-error procedure often followed. Our method is ad hoc however and questions remain on what is the most appropriate technique.

Considerable work on dynamic contact angles using LBM has also been completed (especially with the free-energy LBM; e.g., Zhang and Kwok, 2004; Raiskinmäki et al. 2002; Briant et al. 2002; Fan et al. 2001). A more fundamental basis and method for incorporating surface adsorption is needed (e.g., the various adsorption models discussed in Adamson and

Gast (1997) for example); nevertheless, reasonable results can be obtained. We demonstrate some applications in the following sections.

6.5.3 Capillary Rise

The SCMP LBM can be effectively used to simulate capillary rise in a simple capillary and by extension capillary rise phenomena in more complex porous media. The use of the dimensionless Bond number Bo – which relates capillary and gravitational forces – allows the simulation of direct analogues of real systems in the same way as the Reynolds number allowed the simulation of equivalent flow regimes.

We can adjust the surface adhesion parameter G_{ads} so that the desired contact angle is attained. For simplicity, we make the contact angle zero and we can use the 2-D Young-Laplace equation with a zero contact angle term to determine the pressure difference across a curved (2-D) interface:

$$\Delta P = \frac{\sigma}{r}. \tag{81}$$

With zero contact angle (and assuming there is no adsorbed liquid film), the interface radius of curvature is identical to the half-width of the 2-D capillary tube (a slit). For small capillaries in 3-D, Hyväluoma et al. (2004) have demonstrated that the ratio of capillary cross-sectional area to perimeter can account for discretization and is preferable to the radius for use in Eq. (81).

The hydrostatic pressure difference between the top and bottom of a column of incompressible liquid of height h in a gravitational field g is

$$\Delta P = \rho g h. \tag{82}$$

Equating the right hand sides of (81) and (82) gives

$$\rho g h = \frac{\sigma}{r}, \tag{83}$$

which can be rearranged as the capillary rise equation:

$$h = \frac{\sigma}{r\rho g}. \tag{84}$$

The dimensionless Bond number reflects the balance between gravitational and capillary forces and is

$$Bo = \frac{r^2 \rho g}{\sigma}. \tag{85}$$

Bo is effectively r/h. To illustrate its use in defining an LBM capillary rise arrangement equivalent to a real capillary system, we consider a real capillary slit 0.002 m in width ('radius' = 0.001 m) in contact with a pool of water with density 1000 $kg\ m^{-3}$ and gravity $g = 9.8\ m\ s^{-2}$. The surface tension of water is $72.13 \times 10^{-3}\ Nm^{-1}$. Solving for the capillary rise we obtain

$$h = \frac{72.13 \times 10^{-3}\ Nm^{-1}}{0.001\ m\ 10^3\ kg\ m^{-3}\ 9.8\ m\ s^{-2}} = 7.36 \times 10^{-3}\ m. \tag{86}$$

Now we can compute the Bond number for the real system: Bo = r/h = 1/7.36. Defining an analogous LBM system can begin with a domain size. Say that the maximum capillary length and rise we wish to simulate is on the order of 300 lu. From the Bond number, $r = h$ Bo, and in our specific case the radius r = 300 lu/7.36 = 41 lu.

We still need to incorporate the density of the LBM liquid and determine a gravity value to use in the simulation. Rearranging Eq. (84) we can solve for g as

$$g = \frac{\sigma}{r\rho h} \tag{87}$$

or $g = 2.222 \times 10^{-6}\ lu\ ts^{-2}$ for our model problem. We chose a domain size of 1000×600 lu^2 with a wall on the bottom and $\tau = 1$ (preliminary observations suggest that the rise might be affected by τ, but we have not investigated this). To expedite the computer run we begin with the capillary filled to the approximate rise height from an initially flat liquid/vapor interface at $y = 180\ lu$. The results are shown in Figure 53. The simulated rise of 334 lu is about 10% greater than the target value of 300 lu computed analytically above. In part this is due to the presence of a non-negligible wetting

film (Langmuir 1938) with a thickness of approximately 3 *lu* on the capillary walls. Recomputing the expected rise with Eq. (84) and a effective capillary radius of 38 *lu* together with the observed average liquid density in the capillary (523.5 *mu lu⁻²*) yields 324 *lu* – still an approximation.
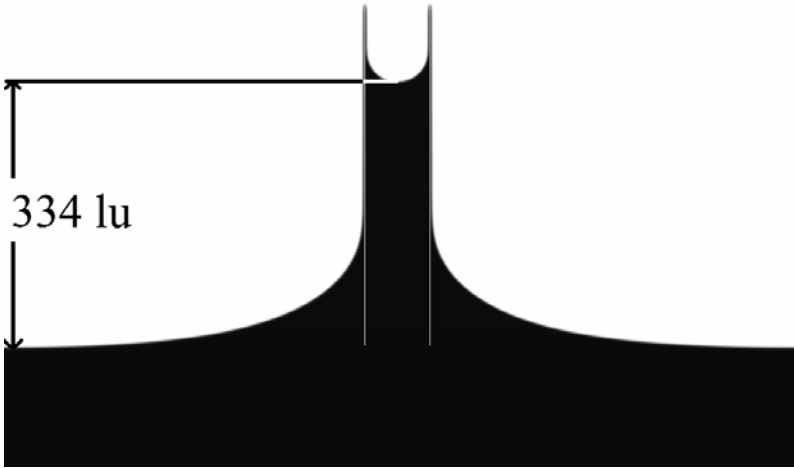
Figure 53. Capillary rise in a 1000×600 *lu²* domain at 81,000 *ts*. Density of fluid shown as gray scale. Capillary walls in white. Dark edges show wetting film.

As discussed in relation to the SCMP EOS, the model we use simulates compressible liquid and gas phases. The liquid phase is actually more compressible than the gas phase.

The hydrostatic pressure difference between the top and bottom of a column of *compressible* fluid of height h in a gravitational field *g* under isothermal conditions is approximately (Halliday and Resnick, 1978)

$$\Delta P = P_0 \left(e^{\rho_0 g h / P_0} - 1 \right). \tag{88}$$

Here ρ_0 and P_0 are the density and pressure at some reference level in the fluid. We can set the *ΔP* on the left side of Eq. (88) to the *ΔP* from the Young-Laplace equation (81) and rearrange to have a capillary rise equation for compressible fluids:

$$\frac{\sigma}{r} = P_0 \left(e^{\rho_0 gh / P_0} - 1 \right) \Rightarrow$$

$$h = \frac{P_0}{\rho_0 g} \ln \left(\frac{\sigma}{P_0 r} + 1 \right),$$

(89)

where a zero contact angle has been assumed. If the reference density is the flat, free interface value of 524.39 *mu lu^{-2}* inside the capillary slit at the level of the free liquid pool outside the slit, then the reference pressure is 24.66 from the EOS. Considering again the effective half-width of the slit due to the adsorbed liquid film, we compute a capillary rise of 329 *lu*, which is reasonably close to the simulated result.

Thus we have several possible strategies; we can ensure that g remains small and that fluid compression is negligible or we can account for the compressibility. The first approach was used by DiPietro et al. (1994) in an early lattice gas simulation of capillary phenomena. Hyväluoma et al. (2004) used an integrated density through the liquid to remove the effect of the compressibility. The above analysis incorporating the compressibility is probably more fundamental and should apply to a broader range of parameters, but we have not investigated it further.

6.5.4 Adsorption/Capillary Condensation

Important phenomena that occur when a vapor phase interacts with a surface include adsorption and capillary condensation. Adsorption is the accumulation of the vapor phase chemical on the solid surface. This accumulation will commonly result in surface condensation and the formation of a liquid film. Numerous physicochemical processes can be responsible for the formation of such films, but if one limits consideration to ubiquitous van der Waals interactions (Tuller et al. 1999), then as a first approximation the thickness of the film formed is a function of the vapor pressure and a Hamaker constant A_{slv} that quantifies the interaction of the solid surface with vapor through a liquid film.

$$h(\Pi) = \sqrt[3]{\frac{A_{slv}}{6\pi\Pi}} .$$

(90)

Here Π is the disjoining pressure (P relative to flat, free interface).

When films on opposing surfaces grow towards one another – such as on opposite walls of a pore – a critical film thickness is reached beyond which the pore spontaneously fills completely. The critical thickness is approximately H/3 where H is the aperture width. The equation governing the simultaneous growth of the films on opposing surfaces is an extension of Eq. (90):

$$\Pi = \frac{A_{slv}}{6\pi h^3} - \frac{A_{lvl}}{6\pi(H-2h)^3} + \frac{A_{slv}}{6\pi(H-h)^3} \, , \tag{91}$$

where A_{lvl} is a new Hamaker constant that governs the interaction of liquid with liquid through an intervening vapor phase. It is possible to estimate the values of the Hamaker constants by fitting Eqs. (90) and (91) to simulations. With 1 or 2 adjustable parameters these are rather inflexible equations however and good fits using the current LBM are elusive. Simulations of film adsorption and capillary condensation with the SC LBM model have not been widely reported in the literature – probably because of difficulties in obtaining satisfying results. Sukop and Or (2003) presented some promising preliminary results using a different form of the adhesive force (Figure 54), but our current, presumably improved code does not seem to simulate these particular phenomena as well. More detailed analysis of the nature of the approximation of surface forces in LBM is needed.
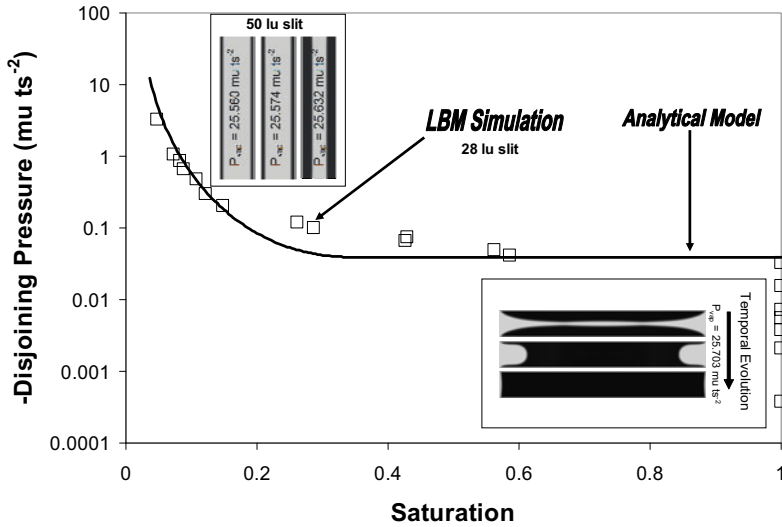
Figure 54. Simulation of film adsorption and capillary condensation from Sukop and Or (2003). The surface force Eq. (66) used the density rather than the ψ function and was not divided by the density as Eq. (47) requires.

6.5.5 Hysteretic Wetting/Drying of Porous Media

Despite challenges with the detailed surface adsorption and capillary condensation simulations, LBM has been used successfully to simulate the hysteretic wetting and drying of porous media. We refer readers to Pan et al. (2004) and Vogel et al. (2005) for 3-D examples.

6.5.6 Fluid Displacement in Porous Media

The displacement of one fluid by another in porous media leads to a rich variety of behaviors depending on the properties of the fluids, the rate of displacement, gravitational effects, and the structure of the medium. Lenormand (1988) condensed the fluid properties and displacement rate into two parameters and showed the variety of behaviors experimentally. As we have seen previously for other fluid phenomena, such observations present opportunities for validation of lattice Boltzmann methods. We present Lenormand's results and some early attempts at simulating such be-

havior with LBM here (Sukop and Or 2003); more advanced efforts should follow.
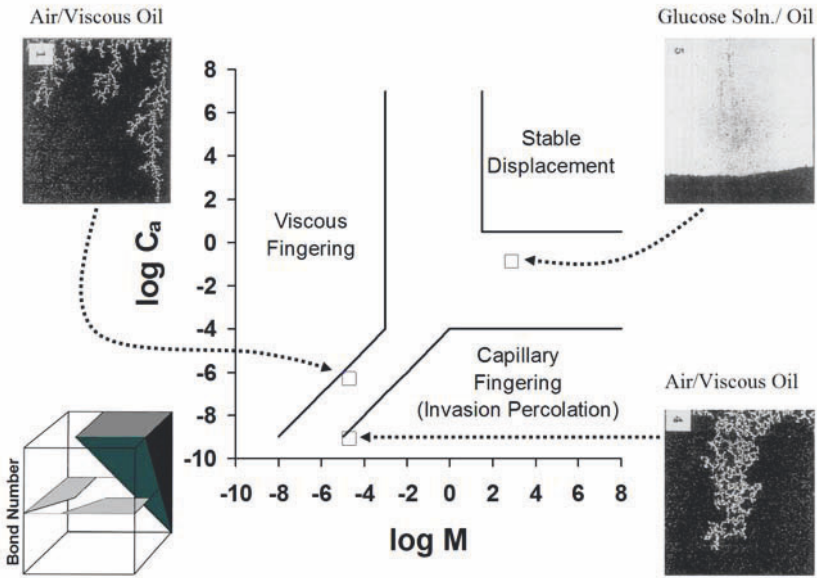
Figure 55. Fluid displacement phase diagram and displacement patterns adapted from Lenormand et al. (1988) with permission of Cambridge University Press. The displacement fronts have different characteristic shapes depending on the Capillary number Ca and the viscosity ratio *M*. The inset on the lower left shows that the 'stable displacement' field expands as the Bond number increases (Berkowitz and Ewing, 1998).

The Capillary number (Ca) is a dimensionless number that gives the relative magnitude of viscous and capillary forces. It can be stated as (Friedman, 1999):

$$Ca = \frac{u\mu}{\phi\sigma\cos\theta} \tag{92}$$

with u the inlet/outlet velocity, μ the viscosity of injected fluid, porosity ϕ, interfacial tension between fluids σ, and contact angle θ. Under many conditions, the other variables are fixed and the Capillary number can be considered a measure of velocity.

The second key variable considered by Lenormand is the ratio (M) of the viscosities of the injected and displaced fluids:

$$M = \frac{\mu_{injected}}{\mu_{displaced}} \tag{93}$$

While this ratio can be varied freely in experimental work by choosing appropriate fluid pairs, this is less straightforward in simple LBM models. For the D2Q9 SCMP model, we find

$$M = \frac{\dfrac{c^2}{3}\left(\tau - \dfrac{\Delta t}{2}\right)\rho_{injected}}{\dfrac{c^2}{3}\left(\tau - \dfrac{\Delta t}{2}\right)\rho_{displaced}} = \frac{\rho_{injected}}{\rho_{displaced}} \cong 0.1 \tag{94}$$

which indicates that the viscosity ratio is controlled by the density ratio when τ is held constant. For the parameters used for SCMP throughout this text, the ratio of vapor to liquid density is on the order of 0.1; far greater density contrasts are generally the norm in the physical world. Similarly to the non-physical relative compressibilities of the liquid and vapor phases noted above, these fixed density and viscosity ratios are not a significant drawback for the computation of equilibrium interface configurations; in fact equilibrium conditions may be attained more quickly due to significant mass transport in the vapor phase. These ratios can affect numerous other phenomena of interest however, and the desire to vary the viscosity ratio to simulate these displacement processes is a good example.

A few strategies are available for varying the viscosity ratio. Ultimately, incorporation of a real equation of state would give the true density contrast. A more immediate approach would be to simply change the EOS parameters to increase the density contrast. This can lead to numerical instabilities however. A second approach is to vary τ as a function of density (Tölke et al. 2002; Nie et al. 1998). We have not experimented with this in the context of SCMP models yet, but successfully use it to simulate the flow of immiscible fluids below.

Single component multiphase lattice Boltzmann methods were applied to invasion percolation in simplified porous media by Sukop and Or (2003). They demonstrated the invading fluid's selectivity for the largest available pore and showed how this is affected by Ca. These abilities are crucial to

simulation of behavior in larger networks. To simulate these displacement processes, Sukop and Or (2003) induced interface movement from a phase-separated initial condition with fixed velocity (u_y = constant, $u_x = 0$) boundary conditions at the vapor inlet and liquid outlet in keeping with the typical experimental practice of constant flow rate. They maintained constant equal volumetric fluxes at the inlet and outlet. Since the (2-dimensional) volume of the fluid injected per unit time is the velocity multiplied by the inlet length, equal volumetric fluxes in the vapor and liquid correspond to equal velocities in the two phases. Implementation of the constant velocity boundary conditions followed the approach proposed by Zou and He (1997).

Figure 56 shows invasion of injected vapor phase through a perforated plate with holes of varying size. In the upper time sequence of images at low Ca; only the largest pore is invaded. In contrast, the bottom sequence shows invasion of the 2 largest pores at larger Ca when viscous effects become more important. These results are similar to those obtained by An-
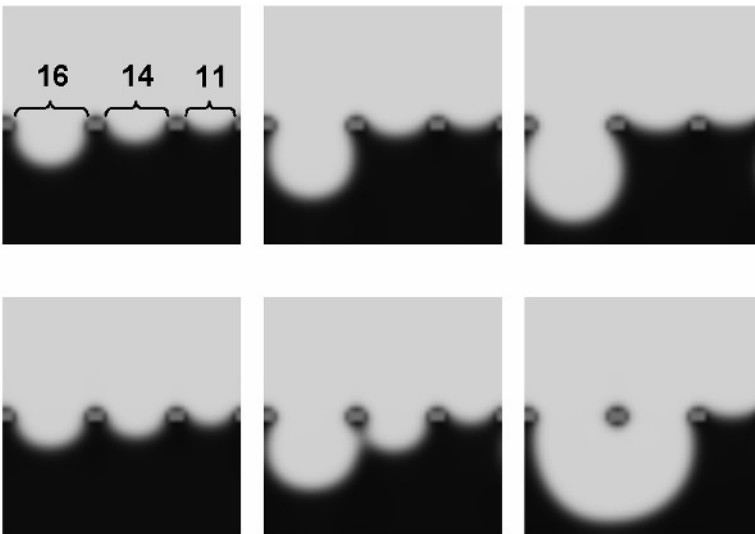
Figure 56. Time series (left to right) of invasion of vapor through plate with perforations of 16, 14, and 11 *lu*. Ca = 10^{-3} (top) and Ca = 10^{-2} (bottom). Liquid black, vapor gray, solids dark gray. (Sukop and Or, 2003) At low Capillary number, capillary forces dominate and only the largest pore is invaded by vapor. Increasing the Capillary number (displacement velocity in this case) causes a second pore to be invaded despite its smaller size.

gelopoulos, et al. (1998) with the free energy lattice Boltzmann model; selectivity for the largest pore was demonstrated but the effect of Ca was not investigated in that work.

Figure 57 illustrates the effects of changes in Ca in a network of pores. Each simulation is carried out on a 200×200 lattice with the same random arrangement of disks of three different sizes. For the disks, r = 2.5, 3.5, or 4.5 *lu*. The domains are periodic in the x direction, although disks at the left and right edges of the domain are fixed at large size and effectively bound the invasion process. Results near breakthrough for two Ca are shown. There are significant differences in the invasion patterns as a function of Ca.
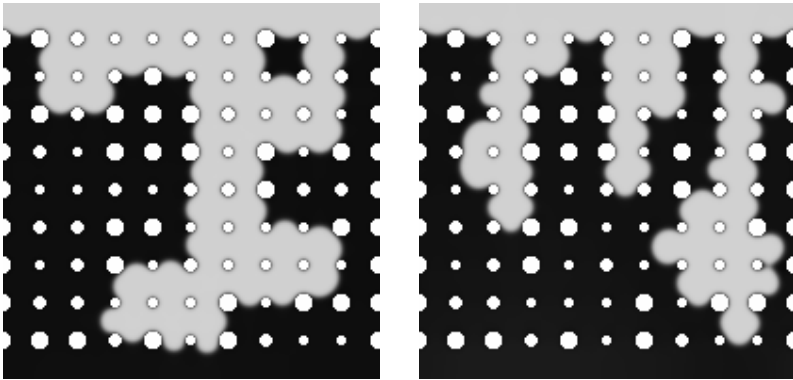
Figure 57. Vapor invasion into a porous medium consisting of random disks at two Capillary numbers. Ca = 10^{-3} (left) and Ca = 10^{-2} (right). Liquid black, vapor gray, solids white.

As noted in the caption of Figure 55, gravity can play a role in stabilizing (flattening) a fluid invasion front at certain Bond numbers when the displaced fluid is denser and tends to collect at the bottom of the domain. Figure 58 illustrates the effect.
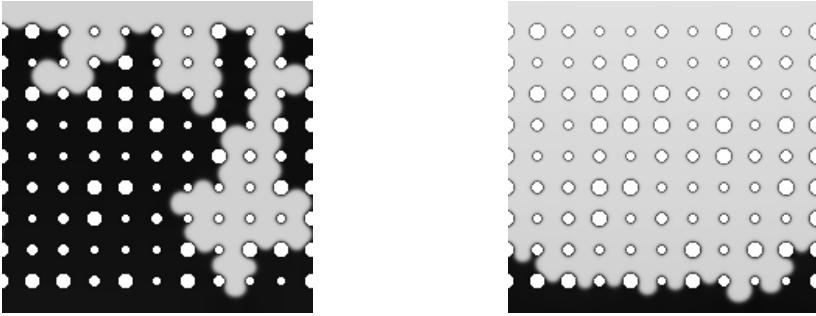
Figure 58. Effect of gravity on displacement front. The front is ramified in the absence of gravity (left). Gravity stabilizes front when the invading fluid is less dense (right). Liquid black, vapor gray, solids white.

6.6 Exercises

1. Plot a family of curves of ρ vs. P for different G (0, -80, -100, -120, and -140). Use $\psi_0 = 4$, $\rho_0 = 200$ mu lu^{-2}.

2. Use the Maxwell Construction to estimate the equilibrium vapor pressure from an equation of state with $G = -120$ and the ψ function parameters $\psi_0 = 4$ and $\rho_0 = 200$ mu lu^{-2}.

3. Simulate a flat interface in a fully periodic domain by beginning with an initial condition with density 500 *mu lu^{-2}* in half of the domain and density 80 *mu lu^{-2}* in the other half. Use the same parameters as in Exercise 2. Run the model to equilibrium and measure the equilibrium densities. Compute the pressure in each phase using the EOS and comment on your results.

4. Simulate drops and bubbles of various sizes in a fully periodic domain by starting with different initial densities in the unstable portion of the EOS (seeds or randomness in the initial density distribution may be necessary to prevent a metastable situation). Measure the drop and bubble diameters and the inside and outside densities. Convert the densities to pressures using the EOS. Plot the pressure difference (high – low) as a function of the inverse radii (the curvature), fit a line and estimate the surface tension. Track and report all units.

5. Compute the critical radius for heterogeneous cavitation under an initial fluid tension of -2 *mu ts^{-1}*. Use the LBM model to simulate the fate of bubbles of size greater than and less than the critical radius.

6. Simulate different contact angles by adding a surface and varying G_{ads}. It is helpful to start with a liquid 'blob' near the surface. Plot the density profile through the vapor, liquid, and solid.

7. The water-water vapor surface tension is 72.13×10^{-3} *N m^{-1}* at 25C. Compute the capillary rise in a clean glass slit of radius 0.001 *m* in the Earth's gravitational field. We choose a slit to reduce the problem to a 2 dimensional one, in which there is only one possible radius of curvature. Assume the contact angle is zero. Show all units.

8. Compute the Bond number for the capillary rise problem above.

9. Propose and run a lattice Boltzmann model illustrating capillary rise at the Bond number found in Exercise 8. Remember that r should be at least 5 or so lattice units and that the 'pool' of liquid that the capillary will be immersed into should be wide enough that a pool height unaffected by capillary rise can serve as a reference level for measuring the liquid rise in your capillary. What is the expected capillary rise in your model? Show all calculations and units. Remember to use a surface adhesion parameter that corresponds closely to the desired contact angle; too big or too small may cause problems. Plot the steady-state result of your model. What is the observed capillary rise? Compare the ratio h/r for the case on Earth, the predicted LB case, and the simulated case.

7 Multicomponent Multiphase (MCMP) LBM

Multicomponent, Multiphase (MCMP) LBM is shown at the bottom right corner of the chart in Figure 38. Multiple chemical components (two for our purposes) such as oil and water are combined with interaction forces between them to yield immiscible fluids. The nature of the interaction forces is repulsive, which is opposite to the attractive forces that bring about van der Waals-like phase separation in SCMP; in MCMP, the phase separation can be thought of as a hydrophobic interaction.

MCMP fluids are of tremendous economic importance because petroleum is often found with water and of considerable environmental importance because non-aqueous phase liquids (NAPLs) in the subsurface often act as long-lived sources of ground water contamination. Our treatment is brief as this topic has not yet been a research area for us and it has been covered in more detail than SCMP in the open literature (e.g., Buckles et al. 1994; Soll et al. 1994; Martys and Chen 1996; Langaas and Papatzacos 2001; Martys and Douglas 2001).

In much of the LBM algorithm, adding a second component requires only introducing a new index to the relevant arrays and a new loop to traverse both components. We call this the 'substance' or `subs` loop and, following standard C, we index the first substance by 0 and the second by 1 in the code but generally refer to them as first and second in the text. An important exception to simply including a `subs` loop is the equilibrium distribution function computation. The equilibrium distribution function is computed from a composite macroscopic velocity

$$\mathbf{u}' = \frac{\sum\limits_{\sigma} \frac{1}{\tau_\sigma} \sum\limits_a f_a^\sigma \mathbf{e}_a}{\sum\limits_{\sigma} \frac{1}{\tau_\sigma} \rho_\sigma}, \tag{95}$$

which differs from the macroscopic uncoupled velocities $\mathbf{u}_\sigma$

$$\mathbf{u}_\sigma = \frac{1}{\rho_\sigma} \sum_{a=0}^{8} f_a^\sigma \mathbf{e}_a \tag{96}$$

of the individual fluids. The densities ρ_σ for each component are as usual

$$\rho_\sigma = \sum_{a=0}^{8} f_a^\sigma. \tag{97}$$

The composite velocity $\mathbf{u}'$ represents the flow of the bulk fluid and, as such, is the more meaningful velocity to view and analyze for the overall fluid flow of the system.

Here is pseudo-code for Eqs. (95), (96), and (97):

```
// Compute density, Eq. (97), and the sums used (below)
// in the velocities.
for( subs=0; subs<NUM_FLUID_COMPONENTS; subs++)
  for( j=0; j<LY; j++)
    for( i=0; i<LX; i++)
    {
      rhoij[subs] = 0.;
      u_xij[subs] = 0.;
      u_yij[subs] = 0.;

      if( !is_solid_node[j][i]) {
        for( a=0; a<9; a++) {
          rhoij[subs] += ftemp_ij[a];
          u_xij[subs] += ex[a]*ftemp_ij[a];
          u_yij[subs] += ey[a]*ftemp_ij[a]; } }
    }

// Compute the composite velocity and individual velocities.
for( j=0; j<LY; j++)
{
  for( i=0; i<LX; i++)
  {
    if( !is_solid_node[j][i])
    {
      ux_sum =  u_xij[0]/tau0 + u_xij[1]/tau1;
      uy_sum =  u_yij[0]/tau0 + u_yij[1]/tau1;

      if( rhoij[0] + rhoij[1] != 0)
      {
        // Composite velocity, Eq. (95).
        uprime_x = ( ux_sum) / ( rhoij[0]/tau0 + rhoij[1]/tau1);
        uprime_y = ( uy_sum) / ( rhoij[0]/tau0 + rhoij[1]/tau1);
      }
      else { uprime_x = 0.; uprime_y = 0.; }
```

```
        // Individual velocities, Eq. (96), x-direction.
        if(  rhoij[0] != 0) { u_xij[0] = u_xij[0] / rhoij[0]; }
        else                { u_xij[0] = 0.;                  }
        if(  rhoij[1] != 0) { u_xij[1] = u_xij[1] / rhoij[1]; }
        else                { u_xij[1] = 0.;                  }

        // Individual velocities, Eq. (96), y-direction.
        if(  rhoij[0] != 0) { u_yij[0] = u_yij[0] / rhoij[0]; }
        else                { u_yij[0] = 0.;                  }
        if(  rhoij[1] != 0) { u_yij[1] = u_yij[1] / rhoij[1]; }
        else                { u_yij[1] = 0.;                  }

      }
    }
  }
```

Note that we allow the macroscopic density to be zero, though both components cannot have zero density simultaneously at the same location. This is discussed in more detail in the next section.

As in the single component case, we add terms for gravity and interaction forces to $\mathbf{u}'$ to obtain $\mathbf{u}^{eq}$. The interaction force term is different, as shown in the following section.

7.1 Interparticle Forces

The force on fluid component σ is

$$\mathbf{F}_\sigma(\mathbf{x}) = -G\psi_\sigma(\mathbf{x},t)\sum_a w_a \psi_{\bar{\sigma}}(\mathbf{x}+\mathbf{e}_a\Delta t,t)\mathbf{e}_a , \qquad (98)$$

where $\bar{\sigma}$ indicates the other fluid component. ψ_σ and $\psi_{\bar{\sigma}}$ are commonly taken as the densities, $\psi_\sigma = \rho_\sigma$ and $\psi_{\bar{\sigma}} = \rho_{\bar{\sigma}}$. The interaction force term $\tau \mathbf{F}_\sigma$ is added to the momentum $\rho\mathbf{u}'$ as before to obtain the velocity for use in the computation of f_σ^{eq}:

$$\rho_\sigma\mathbf{u}_\sigma^{eq} = \rho_\sigma\mathbf{u}'+\tau_\sigma\mathbf{F}_\sigma , \qquad (99)$$

or

$$\mathbf{u}_\sigma^{eq} = \mathbf{u}'+\frac{\tau_\sigma\mathbf{F}_\sigma}{\rho_\sigma} . \qquad (100)$$

The magnitude of G and the fluid densities will determine the magnitude of $\mathbf{F}_\sigma$, which in turn will determine the magnitude of the velocity increment in Eq. (100), which needs to be kept small. This requirement places limits on the value of G while the desire for sharp, non-diffusing interfaces calls for G as large as possible.

In the case of two fluid components, pseudocode for the interaction forces looks like

```
// Compute fluid-fluid interaction force, equation (98),
// (assuming periodic domain).
//
// We begin by computing psi even though in this implementation
// it is the same as rho. A different function of rho could
// be substituted here.
for( subs=0; subs<NUM_FLUID_COMPONENTS; subs++)
  for( j=0; j<LY; j++)
    for( i=0; i<LX; i++)
      if( !is_solid_node[j][i])
        {
          psi[subs][j][i] = rho[subs][j][i];
        }

// Compute the summations in Eq. (98).
for( subs=0; subs<NUM_FLUID_COMPONENTS; subs++)
{
  for( j=0; j<LY; j++)
  {
    jp = ( j<LY-1)?( j+1):( 0   );
    jn = ( j>0   )?( j-1):( LY-1);

    for( i=0; i<LX; i++)
    {
      ip = ( i<LX-1)?( i+1):( 0   );
      in = ( i>0   )?( i-1):( LX-1);

      Fxtemp = 0.;
      Fytemp = 0.;

      if( !is_solid_node[j][i])
      {
        if( !is_solid_node[j ][ip]) // neighbor 1
        { Fxtemp = Fxtemp + WM*ex[1]*psi[subs][j ][ip];
          Fytemp = Fytemp + WM*ey[1]*psi[subs][j ][ip]; }
        if( !is_solid_node[jp][i ]) // neighbor 2
        { Fxtemp = Fxtemp + WM*ex[2]*psi[subs][jp][i ];
          Fytemp = Fytemp + WM*ey[2]*psi[subs][jp][i ]; }
        if( !is_solid_node[j ][in]) // neighbor 3
        { Fxtemp = Fxtemp + WM*ex[3]*psi[subs][j ][in];
          Fytemp = Fytemp + WM*ey[3]*psi[subs][j ][in]; }
        if( !is_solid_node[jn][i ]) // neighbor 4
        { Fxtemp = Fxtemp + WM*ex[4]*psi[subs][jn][i ];
          Fytemp = Fytemp + WM*ey[4]*psi[subs][jn][i ]; }
```

```
        if( !is_solid_node[jp][ip]) // neighbor 5
        { Fxtemp = Fxtemp + WD*ex[5]*psi[subs][jp][ip];
          Fytemp = Fytemp + WD*ey[5]*psi[subs][jp][ip]; }
        if( !is_solid_node[jp][in]) // neighbor 6
        { Fxtemp = Fxtemp + WD*ex[6]*psi[subs][jp][in];
          Fytemp = Fytemp + WD*ey[6]*psi[subs][jp][in]; }
        if( !is_solid_node[jn][in]) // neighbor 7
        { Fxtemp = Fxtemp + WD*ex[7]*psi[subs][jn][in];
          Fytemp = Fytemp + WD*ey[7]*psi[subs][jn][in]; }
        if( !is_solid_node[jn][ip]) // neighbor 8
        { Fxtemp = Fxtemp + WD*ex[8]*psi[subs][jn][ip];
          Fytemp = Fytemp + WD*ey[8]*psi[subs][jn][ip]; }

      } /* if( !is_solid_node[j][i]) */

      Fx[subs][j][i] = Fxtemp;
      Fy[subs][j][i] = Fytemp;

    } /* for( i=0; i<LX; i++) */
  } /* for( j=0; j<LY; j++) */
} /* for( subs=0; subs<NUM_FLUID_COMPONENTS; subs++) */

// Compute the final interaction forces of Eq. (98) using
// the summations computed above.
for( j=0; j<LY; j++)
{
  for( i=0; i<LX; i++)
  {
    if( !is_solid_node[j][i])
    {
      Fxtemp = Fx[1][j][i];
      Fx[1][j][i] = -G*psi[1][j][i]*Fx[0][j][i];
      Fx[0][j][i] = -G*psi[0][j][i]*Fxtemp;

      Fytemp = Fy[1][j][i];
      Fy[1][j][i] = -G*psi[1][j][i]*Fy[0][j][i];
      Fy[0][j][i] = -G*psi[0][j][i]*Fytemp;
    }
  }
}
```

Based on Eqs. (96), (95), and (100), it appeared that zero density of either component was inadmissible; however simple checking for the condition and defaulting to zero-valued velocities and force terms when either density is zero seems to work very well.

As in the SCMP cohesion force, beware of the assumption about a periodic domain (see the parenthetical note in the comment at the top of the code snippet). When using boundary conditions, the interaction force must be computed differently at those boundaries because the density is not necessarily continuous across the periodicity then. Hence, the three terms in Eq. (98) associated with nodes across the periodic boundary are unavailable. We find that replacing those three terms with duplicates of the three

terms in the other direction (towards the interior of the domain instead of across the boundary of the domain) gives good results.

7.2 Phase (Fluid-Fluid) Separation

The Laplace law (6) is also satisfied by two-fluid MCMP systems. Figure 59 shows several LBM data points and a linear fit. The interfacial tension is given by the slope of the line.

The chart shows ΔP (mu ts^{-2}) on the y-axis versus $1/r$ (lu^{-1}) on the x-axis. Fit equation: $y = 2.003x - 3E\text{-}05$, $R^2 = 0.9983$.
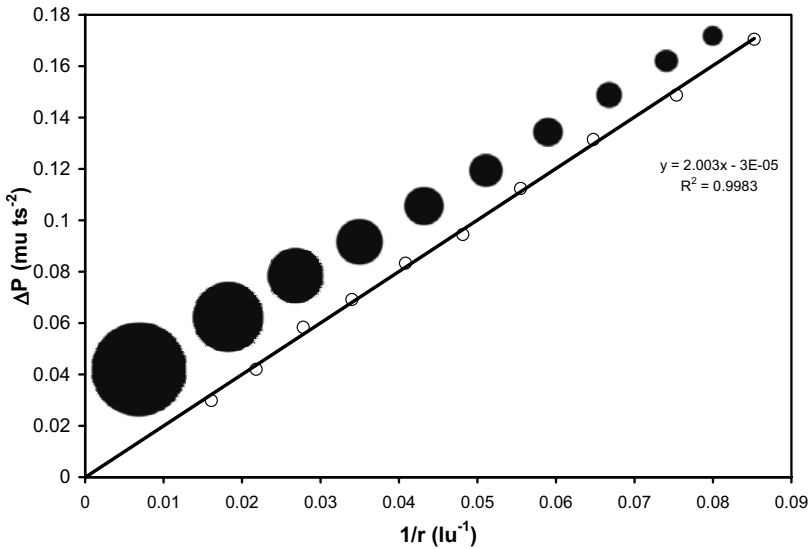
Figure 59. Pressure difference between inside and outside of drops as a function of 1/*r* for immiscible fluids. Simulated drops of component 1 inside component 2 shown adjacent to data points at relative scale. Linear trend confirms simulation agrees with Laplace law. Interfacial tension given by slope.

Our Laplace results for MCMP fluids are based on fairly low resolution runs. These runs take a long time to equilibrate, even when the domain is initialized with a round drop. The relatively low resolution (about $200{\times}200$ lu^2) means that the thickness of the interface is non-negligible relative to the diameter of the drop. This leads to the question of how to distinguish a drop from the surrounding fluid. In other words, what density should be used as the cut-off at the edge of the drop? The most naive choice is half-way between the density inside the drop and the density in the bulk of the

surrounding fluid. This gives a non-zero Laplace plot intercept on the order of 0.001 in our experiments. A more intelligent choice might be the point at which the density profiles of the two fluids cross – the point where, when moving from the interior of the drop towards the exterior, the density of the drop component becomes lower than the density of the surrounding fluid. This does not substantially improve the results however. Another idea is to consider the drop to extend all the way to its lowest density; the area of the drop corresponds to the entire region where the drop component density is higher than its density in the bulk of the surrounding fluid component. This improves the Laplace intercept dramatically in our experiments to an order of about 0.00001. We find this choice mildly distasteful because it includes as part of the drop a region where the density of the ambient fluid is higher than that of the drop.

Another issue related to low resolution experiments is that larger drops can begin to interfere with themselves across the periodic domain boundaries. This would be expected to affect data points corresponding to larger radii in our measurements where the domain size was held constant. However, data points associated with larger radii follow the trend of the other data points closely.

7.3 Metastable States

The fluids can be initialized as a metastable emulsion; any tiny perturbation will lead to catastrophic phase separation. The phase separation proceeds via an exsolution-like process (Figure 60).
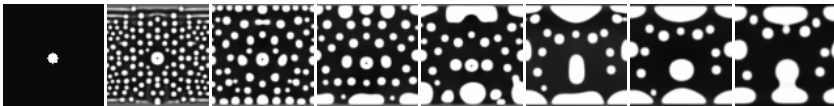
Figure 60. Catastrophic phase separation initiated by a seed of separated fluid (left) in an emulsified binary mixture that is metastable in the absence of perturbations.

Quasi-metastable (kinetically frustrated) states can also be obtained by initializing a desired condition.

7.4 MCMP LBM with Surfaces

Each fluid component can interact with solids independently via separate interaction parameters. This is as expected considering that there are interfacial tension values between the two fluids (σ_{12}) and between each fluid and the surface (σ_{S1} and σ_{S2}) that appear in Young's equation,

$$\cos\theta = \frac{\sigma_{S2} - \sigma_{S1}}{\sigma_{12}}, \tag{101}$$

which determines the contact angle measured in fluid 1 (Adamson and Gast 1997).

Surface forces are incorporated into the multicomponent model much the same way as in the single component model (Eq. (66)) except that now separate forces are computed for each component and ρ is used directly because $\psi(\rho) = \rho$:

$$\mathbf{F}_{ads}^{\sigma}(\mathbf{x},t) = -G_{ads}^{\sigma}\rho(\mathbf{x},t)\sum_{a} w_a s(\mathbf{x} + \mathbf{e}_a \Delta t)\mathbf{e}_a . \tag{102}$$

The following pseudocode illustrates how the surface adhesion forces for each fluid substance (`subs` index 0 or 1 in the code) are computed. Note that the multiplication by $\rho(\mathbf{x},t)$ is omitted because incorporation of the forces into the model by Eq. (100) requires division by ρ. This also makes the `sforce` below independent of time so it needs to be computed just once.

```
for( j=0; j<LY; j++)
{
  jp = ( j<LY-1)?( j+1):( 0   );
  jn = ( j>0    )?( j-1):( LY-1);

  for( i=0; i<LX; i++)
  {
    ip = ( i<LX-1)?( i+1):( 0   );
    in = ( i>0    )?( i-1):( LX-1);

    if( !is_solid_node[j][i])
    {
      sum_x=0.;
      sum_y=0.;
```

```
        if( is_solid_node[j ][ip])  // neighbor 1
        { sum_x = sum_x + WM*ex[1];
          sum_y = sum_y + WM*ey[1];}
        if( is_solid_node[jp][i ])  // neighbor 2
        { sum_x = sum_x + WM*ex[2];
          sum_y = sum_y + WM*ey[2];}
        if( is_solid_node[j ][in])  // neighbor 3
        { sum_x = sum_x + WM*ex[3];
          sum_y = sum_y + WM*ey[3];}
        if( is_solid_node[jn][i ])  // neighbor 4
        { sum_x = sum_x + WM*ex[4];
          sum_y = sum_y + WM*ey[4];}
        if( is_solid_node[jp][ip])  // neighbor 5
        { sum_x = sum_x + WD*ex[5];
          sum_y = sum_y + WD*ey[5];}
        if( is_solid_node[jp][in])  // neighbor 6
        { sum_x = sum_x + WD*ex[6];
          sum_y = sum_y + WD*ey[6];}
        if( is_solid_node[jn][in])  // neighbor 7
        { sum_x = sum_x + WD*ex[7];
          sum_y = sum_y + WD*ey[7];}
        if( is_solid_node[jn][ip])  // neighbor 8
        { sum_x = sum_x + WD*ex[8];
          sum_y = sum_y + WD*ey[8];}

        for( subs=0; subs<NUM_FLUID_COMPONENTS; subs++)
        {
          sforce_x[subs][j][i] = -Gads[subs]*sum_x;
          sforce_y[subs][j][i] = -Gads[subs]*sum_y;
        }
      }
    }
  }
}
```

A rationale similar to that applied above in the section on contact angles for SCMP models can also be fruitfully applied to MCMP. Suppose we wish to ensure that fluid A near a solid surface experiences a force environment similar to the one it feels when it is surrounded by like fluid. This will prevent excess compression by surface forces for example. Considering Eq. (98), we have $F^A = \rho_A \rho_B G$ for fluid A surrounded by itself, while Eq. (102) leads to $F_{ads}{}^A = \rho_A G_{ads}{}^A$ for the fluid surrounded by solids on all sides. Equating these gives

$$G_{ads}^A = \rho_B G \qquad\qquad (103)$$

and by symmetry

$$G_{ads}^B = \rho_A G . \qquad\qquad (104)$$

The significance of these relationships is that when the complementary density is low, G_{ads} should be small relative to G. As discussed below however, it is other constraints on the G, G_{ads}^1 and G_{ads}^2 values that determine the relative wetting of the fluids.

It is expected that there is a direct relationship between the interfacial tension values in Young's equation (101) and the G, G_{ads}^1 and G_{ads}^2 values. When the densities of each fluid substance are identical and $G_{ads}^1 = G_{ads}^2$, it is reasonable to expect $\sigma_{S1} = \sigma_{S2}$ and a 90° contact angle. Figure 61 shows the results of such a simulation.

Figure 61. Multicomponent fluids interacting with a surface when $G = 0.1$ and $G_{ads}^1 = G_{ads}^2 = -0.01$. The density of the first substance (with index 0) is show in gray scale. Contact angle is 90°. Simulation domain 100×100 lu^2. Image represents 6000 time steps from an initial condition of a 20×20 lu^2 square of the first substance (with density 1 and complementary density 0) surrounded by the second substance. $\tau = 1$ for both fluids.

Examining Young's equation (101) further and assuming that we can simply replace the interfacial tension values σ_{12}, σ_{S1} and σ_{S2} by G, G_{ads}^1 and G_{ads}^2, we can rearrange Eq. (101) as

$$G \cos \theta = G_{ads}^2 - G_{ads}^1 = \Delta G_{ads}. \tag{105}$$

This makes it clear that it is the difference in the adhesion strength parameters that determines the relative wetting. Consider the following example in which we desire a 45° contact angle. In this case, $\Delta G_{ads} = \sqrt{2}/2\,G$ from Eq. (105). If $G = 0.1$, then $\Delta G_{ads} \approx 0.0707$. And, if $G_{ads}^1 = -0.02$, then $G_{ads}^2 = \Delta G_{ads} + G_{ads}^1 = 0.0507$. Note that the wetting fluid (here 1) must have the lower G_{ads}. The result of applying these values is shown in Figure 62.

Figure 62. Multicomponent fluids interacting with a surface when $G = 0.1$, $G_{ads}^{1} = -0.02$, and $G_{ads}^{2} = 0.0507$. The density of the first substance (with index 0) is show in gray scale. Line indicates 45° contact angle. Simulation domain $1000 \times 200 \; lu^2$. Image represents 38,000 time steps from an initial condition of a $200 \times 50 \; lu^2$ rectangle of the first substance (with density 1 and complementary density 0) surrounded by the second substance. $\tau = 1$ for both fluids.

7.5 Two-Phase Flow

Ginzburg (1994) presented simulations and analytical solutions for two phase flow in a channel where the phases have different viscosities. These types of flows can be simulated with LBM by setting separate τ for each fluid. Figure 63 shows results for immiscible fluids with τ for substance 0 = 1.2 and τ for substance 1 = 0.6. The gravitational acceleration applied to each substance is $-10^{-6} \; lu \; ts^{-2}$. The fluid-fluid interaction parameter is given a value of $G = 0.05$ and the domain is periodic in y and bounded by walls at $x = 0$ and $x = 99$. The initial densities are 1 and 0 $mu \; lu^{-2}$ and the domain is initialized with substance 0 on the left half and substance 1 on the right. Figure 63 shows the velocity profile. There is some noise presumably related to the spurious interface velocities at the interface between the fluids, but overall the profile is as expected.

In accordance with Eq. (2), more viscous substance 0 presents a lower overall $d\mathbf{u}/dx$. It is interesting that substance 0 has been compressed and substance 1 has been rarefied so that the interface is found at $x = 39$ instead of its initial position of $x = 50$. Further, the concentration of substance 0 'dissolved' in substance 1 has increased on the right hand side of the domain. All of these phenomena deserve further exploration.
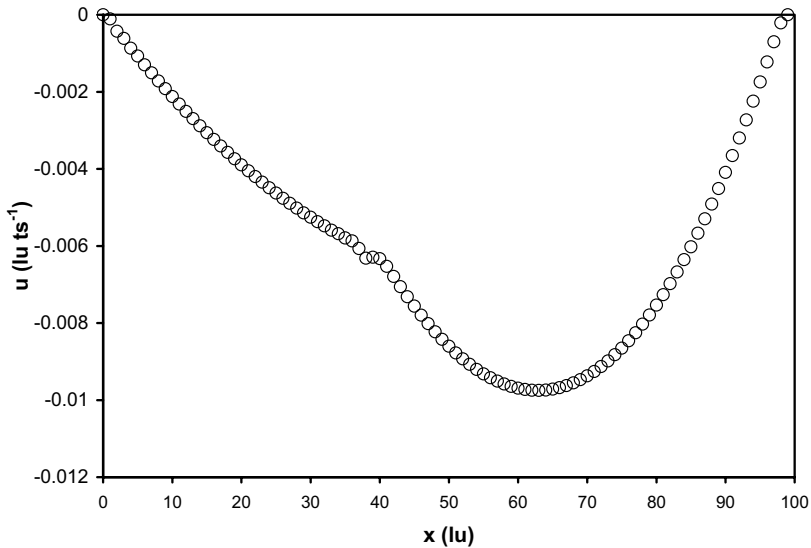
Figure 63. Velocity profile for flow of two immiscible fluids in a 100 lu channel. More viscous fluid is on left. Downward flow driven by gravity.

7.6 Exercises

1. Run a series of simulations to determine the interfacial tension between two fluids.

2. Set up a poised metastable system and demonstrate its persistence. Seed the same domain and demonstrate phase separation.

3. In the MCMP contact angle example presented above, $\Delta G_{ads} \approx 0.0707$, $G_{ads}^{1} = -0.02$, and $G_{ads}^{2} = 0.0507$. Run a model to test if this is equivalent to $G_{ads}^{1} = -\Delta G_{ads}/2$ and $G_{ads}^{2} = \Delta G_{ads}/2$.

4. Simulate the flow of two immiscible fluids between parallel plates. Vary the viscosity for the two fluids from where they are equal to where they differ by a factor of 7 or more. Plot the velocity and density profiles for each case.

8 Solute Transport

Lattice Boltzmann methods (LBM) offer an exceptionally powerful tool for the computation of advective and diffusive transport in complex porous media. Mass and energy transport can be simulated. Unlike traditional models that rely on space-averaged properties that lose meaning at the pore scale (such as porosity, hydraulic conductivity, and dispersivity), LBM can compute detailed solutions to the Navier-Stokes equations for fluid flow and solute diffusion considering the intricacies of porous medium geometry. Dispersion is an outcome of solute advection and diffusion in this detailed flow field and is not bound to any particular macroscopic transport model or conceptualization.

The simulation of 2-D transport of dilute solutes does not require the use of all 9 velocity directions in the D2Q9 model; 4 perpendicular velocities provide adequate symmetry for the simulation of 2-D diffusive phenomena (Wolf-Gladrow 2000) and 6 are adequate for 3-D simulations (Stockman et al. 1998). For the simplicity that comes from consistency with our previous models, we use all nine velocities.

This chapter is organized as follows: first we provide a brief overview of selected applications of LBM to solute transport. Then two different approaches for modeling solutes are introduced. Detailed boundary conditions are provided for the 'passive' solute approach and the modifications required to simulate solute-induced buoyancy are given. The remainder of the chapter is dedicated to examples involving comparisons with analytical solutions to diffusion and convection-diffusion/dispersion problems, including diffusion in infinite and bounded domains, diffusion on a propagating front, Taylor dispersion, and Péclet number correlations.

8.1 Selected Review of Previous Applications of LBM to Solute Transport

Since the early 1990s, LBM has been used to simulate solute transport. Examples range from simple diffusion to buoyant mixing of dense solutions in 3 dimensions. Some papers and theses specifically addressing LBM solute transport modeling include Zhang and Ren (2003), Yoshino and Inamuro (2003), Zhang et al. (2002a-c), Drazer and Koplik (2001), Grubert (1997), and Noble (1997). Diffusion simulation capabilities of LBM are discussed in Flekkøy (1993), Shan and Doolen (1996), and Wolf-Gladrow (2000) among others. Mixing (dispersion) due to density variations and resulting buoyant effects between the tracer plume and unaffected solvent (Stockman 1997; Jalbert et al. 2000) can readily be simulated with LBM. Low viscosity fluids and/or high velocity flow conditions can lead to high Reynolds number flows characterized by eddy formation and the additional dispersion that results. Yoshino and Inamuro (2003) recently computed solute transport under such flows in simple porous media. A number of earlier papers used lattice gas methods to investigate solute transport and provide useful background for LBM studies; examples include Gutfraind et al. (1995), Perea-Reeves and Stockman (1997), Stockman (1997), and Stockman et al (1997a,b).

Stockman et al. (1997b) used lattice gas methods and LBM (following Flekkøy (1993)) to investigate mixing at fracture intersections at varying Péclet numbers. Stockman et al. (2001) used a 3-D LBM in a similar study. Stockman et al. (1998) investigated 3-D dispersion in fractures and double-diffusive fingering. Zhang et al. (2002a) presented a means of solving the 2-D advection-dispersion equation with anisotropic dispersion coefficients and spatially variable velocities (at least variable in one dimensional flow). Otherwise, explicit consideration of the flow field (and simultaneous computation of the flow field in a porous medium for example) was not presented. Later papers by this group (Zhang et al. 2002b and 2002c; Zhang and Ren, 2003) present boundary conditions applicable to their model, a 3-D model that can be applied to unsaturated soils (or ground water flow), and a 1-D kinetic sorption/transport model. The strategy presented in these papers is useful as an alternative to more traditional means of solving the anisotropic advection-dispersion equation. Our focus in this chapter however is on simultaneous solution of the advective and diffusive components of transport in specific geometric models of porous media. Rather than being an input parameter, the dispersion is an output of the model under these conditions, and long-standing questions regarding dispersion can be ad-

dressed free from the assumption that the traditional advection-dispersion equation is a correct representation of the process. Perfect and Sukop (2001) presented a review of pore geometry-based transport models that offers potential test cases for this type of modeling. In our opinion, the attributes of LBM – particularly its ability to simulate transport at much higher Reynolds numbers than are commonly encountered in 'normal' porous media – make it the ideal tool for use well beyond the pore scale. Our own interests include simulating flows in caves and conduits systems in karstic aquifers for instance; to relegate LBM strictly to pore scale modeling would be unfortunate. Thus, our use of the term 'pore' in the solute transport context should be broadly interpreted.

There are two common ways to simulate solute transport in discrete pore/solid systems with LBM. The second component of normal MCMP can be treated as a solute by eliminating or grossly reducing the non-local interaction between the fluids and initializing them with complementary densities (lower left-hand corner of Figure 38). Alternatively, a modified version of the second component that simulates a passive solute (or any scalar quantity like temperature) can be devised (e.g., Yoshino and Inamuro 2003). In the first approach, the solute component is coupled with the fluid component as usual for MCMP, albeit with small or zero non-local interaction. In the second approach, the solute component is passive, having no velocity of its own, and is carried along (advected) by the background fluid. We discuss and present examples from each approach but emphasize the passive component approach.

8.2 Active Solute Component

Shan and Doolen (1996) gave a thorough analysis of the use of 'normal' multicomponent models (those capable of phase separation) for the simulation of diffusion. In these models, the densities of the solute and solvent are 'complementary' (Figure 64) in that an increase of solute concentration is accompanied by a decrease in solvent concentration.

We present comparisons of diffusion simulations using this active solute approach with analytical solutions in Section 8.5.1 below.
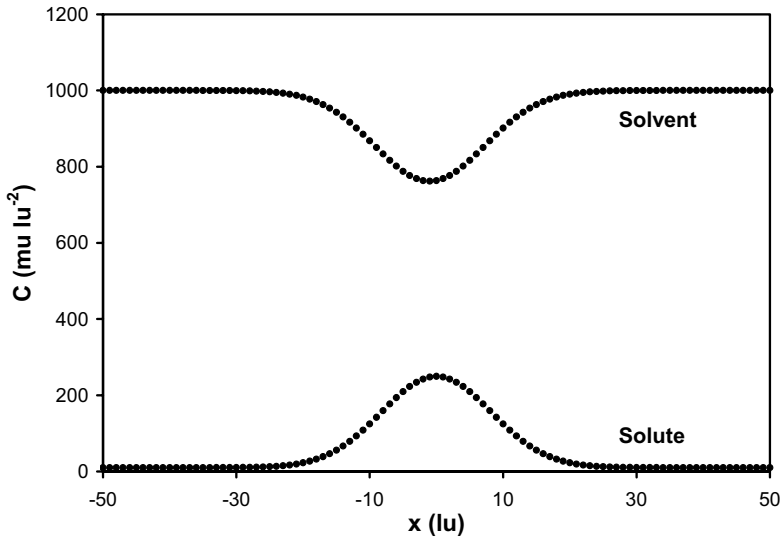
Figure 64. Complementary densities in MCMP solute model.

8.2.1 Boundary Conditions

Boundary conditions for the active solute component are identical to those for any fluid: periodic, bounceback, specified velocity, and specified pressure are available and have been detailed above.

8.3   Passive Solute Component

In the passive solute approach, solute is simulated by a second distribution f_σ called the solute or 'σ' component. This corresponds closely to the normal fluid distribution function except with a simpler equilibrium distribution

$$f_{\sigma,a}^{eq} = w_a \rho_\sigma (1 + 3\mathbf{e}_a \cdot \mathbf{u}). \tag{106}$$

The concentration ρ_σ

$$\rho_\sigma = \sum_{a=0}^{8} f_{\sigma,a} \qquad (107)$$

is analogous to fluid density.

The diffusion coefficient D_σ is given by

$$D_\sigma = \tfrac{1}{3}\left(\tau_\sigma - \tfrac{1}{2}\right) \qquad (108)$$

and is analogous to viscosity.

8.3.1 Boundary Conditions

Yoshino and Inamuro (2003) define concentration boundary conditions on the surfaces of solids. Concentration boundary conditions can also be applied at inflow/outflow boundaries. We give them for vertical and horizontal walls/boundaries here, although they can be generalized to solids of any shape.

We derive constant concentration boundaries, constant flux boundaries, and zero diffusive flux boundaries in the following sections. These derivations are related in spirit to the constant velocity and constant pressure boundary derivations presented in Chapter 4: unknown direction-specific densities that arise after streaming are solved on the basis of known values and specified conditions.

Constant concentration boundaries

After the streaming step, there are unknowns pointing out of walls and side (non-periodic) boundaries. These unknowns can be solved for in a way that maintains a specified constant concentration $\overline{\rho}_\sigma$ at these boundary nodes.

Due to symmetry, we will only consider one case, a north boundary, in detail. The other three will be stated but not derived.

Figure 65 shows the unknowns at a north boundary after streaming. This could be at the north side of the full domain or at a horizontal solid surface on the north side of a fluid region inside the domain. If it is the north boundary of the domain, it could be either a wall or an in-

flow/outflow. The mechanism for defining the constant concentration is the same in all cases.
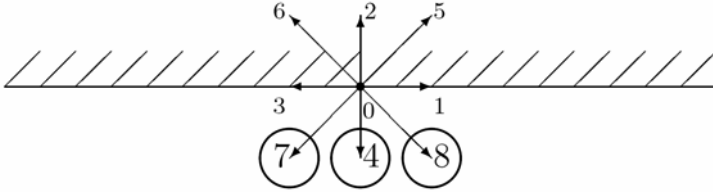

Figure 65. Solute direction-specific density unknowns (circled) after streaming at a north surface/boundary.

We require the concentration

$$\sum_a f_{\sigma,a} \qquad (109)$$

to be equal to a prescribed value denoted by $\overline{\rho}_\sigma$. The contributions to $\sum_a f_{\sigma,a}$ from $f_{\sigma,a}$ for a in $\{0,1,2,3,5,6\}$ are known after streaming because they come from regions inside the model domain.

Motivated by the weighting factors w_a on the directional densities in the equilibrium distribution function, we assume that the unknown directional densities $f_{\sigma,a}$ for a in $\{4,7,8\}$ are of the form $f_{\sigma,a} = w_a \rho'_\sigma$, where ρ'_σ denotes the residual amount of concentration needed to satisfy the specified concentration condition $\overline{\rho}_\sigma$ at the current lattice node.

Then from

$$\overline{\rho}_\sigma = \sum_a f_{\sigma,a}$$
$$= f_{\sigma,0} + f_{\sigma,1} + f_{\sigma,2} + f_{\sigma,3} + f_{\sigma,5} + f_{\sigma,6} + w_4\rho'_\sigma + w_4\rho'_\sigma + w_4\rho'_\sigma \qquad (110)$$

we can compute the residual concentration ρ'_σ as

$$\rho'_\sigma = \frac{\overline{\rho}_\sigma - (f_{\sigma,0} + f_{\sigma,1} + f_{\sigma,2} + f_{\sigma,3} + f_{\sigma,5} + f_{\sigma,6})}{w_4 + w_7 + w_8} . \tag{111}$$

Finally, under the assumption on the form of the unknown directional densities, we solve for them as

$$f_{\sigma,4} = w_4 \rho'_\sigma = \tfrac{1}{9} \rho'_\sigma , \tag{112}$$

$$f_{\sigma,7} = w_7 \rho'_\sigma = \tfrac{1}{36} \rho'_\sigma , \tag{113}$$

and

$$f_{\sigma,8} = w_8 \rho'_\sigma = \tfrac{1}{36} \rho'_\sigma . \tag{114}$$

To summarize the procedure, we compute the residual concentration ρ'_σ needed to add to the contribution from the known directional concentrations so that the total sums to the specified condition $\overline{\rho}_\sigma$. Then this residual concentration is distributed among the unknown directional concentrations according to the weighting used in the computation of the equilibrium distribution function.

Here is a pseudocode implementation of constant concentration boundaries on all four sides (where, on the north side, for example, the 6 factor comes from $1/(w_4 + w_7 + w_8)$):

```
// Constant concentration on north side.
for( i=0; i<LX; i++)
{
  fi = ftemp[LY-1][i];
  rho_sigma_prime = 6.*( rho_sigma_bar
                     - ( fi[0] + fi[1] + fi[3]
                       + fi[2] + fi[5] + fi[6]));
  fi[4] = (1./ 9.)*rho_sigma_prime;
  fi[7] = (1./36.)*rho_sigma_prime;
  fi[8] = (1./36.)*rho_sigma_prime;
}

// Constant concentration on south side.
for( i=0; i<LX; i++)
{
  fi = ftemp[0][i];
  rho_sigma_prime = 6.*( rho_sigma_bar
```

```
                                - ( fi[0] + fi[1] + fi[3]
                                  + fi[7] + fi[4] + fi[8]));
    fi[2] = (1./ 9.)*rho_sigma_prime;
    fi[5] = (1./36.)*rho_sigma_prime;
    fi[6] = (1./36.)*rho_sigma_prime;
  }

  // Constant concentration on east side.
  for( j=0; j<LY; j++)
  {
    fj = ftemp[j][LX-1];
    rho_sigma_prime = 6.*( rho_sigma_bar
                          - ( fj[0] + fj[2] + fj[4]
                            + fj[1] + fj[5] + fj[8]));
    fj[3] = (1./ 9.)*rho_sigma_prime;
    fj[7] = (1./36.)*rho_sigma_prime;
    fj[6] = (1./36.)*rho_sigma_prime;
  }

  // Constant concentration on west side.
  for( j=0; j<LY; j++)
  {
    fj = ftemp[j][0];
    rho_sigma_prime = 6.*( rho_sigma_bar
                          - ( fj[0] + fj[2] + fj[4]
                            + fj[3] + fj[6] + fj[7]));
    fj[1] = (1./ 9.)*rho_sigma_prime;
    fj[5] = (1./36.)*rho_sigma_prime;
    fj[8] = (1./36.)*rho_sigma_prime;
  }
```

Constant flux boundaries

As pointed out by van Genuchten and Wierenga (1986), maintenance of a true constant concentration boundary is challenging under laboratory conditions. Most often it is the chemical flux that is held constant; a solution of known concentration C_0 is pumped into a domain at constant flow rate. If the diffusion or dispersion coefficient is high, the velocity is low, or there is a strong concentration gradient near the inlet, the diffusive/dispersive flux at the inlet cannot be ignored and the proper boundary condition is

$$\left(-D\frac{\partial C}{\partial x} + vC \right)\Bigg|_{x=0+} = vC_0, \tag{115}$$

where the $x = 0+$ refers to just inside the domain where C denotes concentration which is equivalent to ρ_σ as used in the context of the Yoshino and Inamuro (2003) boundary conditions elsewhere in this chapter.

Eq. (115) can be rearranged and applied as a finite difference to the boundary and first interior discrete nodes of the model as follows:

$$-D\frac{C_1 - C}{1} + vC = vC_0 \Rightarrow C = \frac{vC_0 + DC_1}{D + v}.$$

(116)

C_1 is the concentration at the first interior node and the difference $C_1 - C$ is the gradient since the lattice step is 1 lu. C is the concentration applied at the boundary.

Zero diffusive flux boundaries

Solute fluxes can arise from concentration gradients in accordance with Fick's Law. Zero concentration gradient boundaries eliminate pure diffusive fluxes into or out of a domain. Chemical engineers, hydrogeologists, and soil physicists use a Fick's Law approach to treat the anisotropic dispersion that arises from velocity variations. In this case, zero concentration gradient eliminates the dispersive flux but leaves the advective flux (Cv) unaffected.

After the streaming step, there are unknowns pointing out of walls and side (non-periodic) boundaries. These unknowns can be solved for in a way that maintains a zero solute diffusive flux in the direction normal to the boundary. Due to symmetry, we will only consider one case, a north boundary, in detail. The south, west, and east conditions will be stated but not derived.

Figure 65 shows the unknowns at a north boundary after streaming. This could be at the north side of the full domain or at a horizontal solid surface on the north side of a fluid region inside the domain. If it is the north boundary of the domain, it could be either a wall or an inflow/outflow. The zero diffusive flux concentration procedure is the same.

We require the diffusive solute mass flux normal to the boundary to be zero:

$$\sum_a f_{\sigma,a}\mathbf{e}_a \cdot \mathbf{n} = 0.$$

(117)

Here **n** is the unit vector normal to the surface. The contributions from $f_{\sigma,a}$ for a in $\{0,1,2,3,5,6\}$ are already known.

Motivated by the weighting factors w_a on the directional densities in the equilibrium distribution function, we assume that the unknown directional densities $f_{\sigma,a}$ for a in $\{4,7,8\}$ are of the form $f_{\sigma,a} = w_a \rho'_\sigma$, where ρ'_σ denotes the residual amount of concentration needed to satisfy the zero concentration gradient condition at the current lattice node.

Next, we note that directions $a = 1$ and 3 are perpendicular to the surface normal vector **n** and therefore that the dot product $\mathbf{e}_a \cdot \mathbf{n}$ will not contribute to the sum Eq. (117). The same is true of the zero velocity, $a = 0$ case. We also have $\mathbf{e}_a \cdot \mathbf{n} = 1$ for $a = \{4,7,8\}$ as these vectors have components of magnitude 1 that are parallel to n. Finally, $\mathbf{e}_a \cdot \mathbf{n} = -1$ for $a = \{2,5,6\}$, where the components are parallel but opposite in direction. Then we can expand Eq. (117) as

$$
\begin{aligned}
\sum_a f_{\sigma,a} \mathbf{e}_a \cdot \mathbf{n} = \\
0 f_{\sigma,0} + 0 f_{\sigma,1} - f_{\sigma,2} + 0 f_{\sigma,3} - f_{\sigma,5} - f_{\sigma,6} \\
+ w_4 \rho'_\sigma + w_7 \rho'_\sigma + w_8 \rho'_\sigma = 0.
\end{aligned}
\tag{118}
$$

Solving for the residual concentration ρ'_σ we obtain

$$
\rho'_\sigma = \frac{f_{\sigma,2} + f_{\sigma,5} + f_{\sigma,6}}{w_4 + w_7 + w_8}.
\tag{119}
$$

Finally, under the assumption on the form of the unknown directional densities, we solve for them as

$$
f_{\sigma,4} = w_4 \rho'_\sigma = \tfrac{1}{9} \rho'_\sigma,
\tag{120}
$$

$$
f_{\sigma,7} = w_7 \rho'_\sigma = \tfrac{1}{36} \rho'_\sigma,
\tag{121}
$$

and

$$
f_{\sigma,8} = w_8 \rho'_\sigma = \tfrac{1}{36} \rho'_\sigma.
\tag{122}
$$

To summarize the procedure, we compute the residual concentration ρ'_σ needed to add to the contribution from the known directional concentrations so that the zero diffusive mass flux condition is satisfied in the direction normal to the boundary. Then this residual concentration is distributed among the unknown directional concentrations according to the weighting used in the computation of the equilibrium distribution function.

It is also possible to make this boundary condition a constant diffusive flux condition simply by including a prescribed flux J_0 on the right hand side of Eq. (117). We show it that way in the following pseudocode:

```
// Constant diffusive flux boundary on north side.
for( i=0; i<LX; i++)
{
  fi        = ftemp[LY-1][i];
  rho_prime = 6.*( J0 + fi[2] + fi[5] + fi[6]);
  fi[4]     = (1./ 9.)*rho_prime;
  fi[7]     = (1./36.)*rho_prime;
  fi[8]     = (1./36.)*rho_prime;
}

// Constant diffusive flux boundary on south side.
for( i=0; i<LX; i++)
{
  fi        = ftemp[0][i];
  rho_prime = 6.*( J0 + fi[7] + fi[4] + fi[8]);
  fi[2]     = (1./ 9.)*rho_prime;
  fi[5]     = (1./36.)*rho_prime;
  fi[6]     = (1./36.)*rho_prime;
}

// Constant diffusive flux boundary on east side.
for( j=0; j<LY; j++)
{
  fj        = ftemp[j][LX-1];
  rho_prime = 6.*( J0 + fi[1] + fi[5] + fi[8]);
  fj[3]     = (1./ 9.)*rho_prime;
  fj[7]     = (1./36.)*rho_prime;
  fj[6]     = (1./36.)*rho_prime;
}

// Constant diffusive flux boundary on west side.
for( j=0; j<LY; j++)
{
  fj        = ftemp[j][0];
  rho_prime = 6.*( J0 + fi[3] + fi[6] + fi[7]);
  fj[1]     = (1./ 9.)*rho_prime;
  fj[5]     = (1./36.)*rho_prime;
  fj[8]     = (1./36.)*rho_prime;
}
```

More testing of these various conditions is needed.

8.4   Solute-induced Buoyancy

The rich behaviors of solute- and temperature-induced buoyancy are easily produced in these LBM simulations. This was illustrated with lattice gases by Perea-Reeves and Stockman (1997) and Stockman (1997). Solute concentration and temperature can readily be exchanged (e.g., Zhou et al. 2004). The magnitude of the body force/gravity term acting on the main fluid component is modified to produce buoyancy as follows:

$$\mathbf{u} = \frac{1}{\rho}\sum_{a=0}^{8} f_a \mathbf{e}_a + \tau \mathbf{g} \frac{\rho + \rho_\sigma}{\rho}. \tag{123}$$

At lattice nodes where the concentration ρ_σ of solute is larger, the body force/gravity term is larger, thereby contributing more to the momentum of particles in the direction of the body force/gravity at that node. This induces buoyancy effects based on density differences due to solute and/or temperature. We present some examples below. Clearly, factors relating concentration and/or temperature changes to density changes ($\partial\rho/\partial C$, $\partial\rho/\partial T$) will be necessary as factors multiplying the concentration in Eq. (123) to insure the correct scaling of the effect.

8.5   Examples

8.5.1 Diffusion

The simplest test of any solute transport model is the ability to simulate diffusion. For one-dimensional diffusion, Fick's 2nd Law (also known as the Heat or Diffusion Equation) in 1 dimension is

$$\frac{dC}{dt} = D\frac{d^2 C}{dx^2}, \tag{124}$$

where C is the concentration, t is time, D is the diffusion coefficient, and x is the space coordinate. Numerous analytical solutions are available for various boundary and initial conditions; many were compiled by Crank (1975). We present comparisons of several analytical solutions with LBM simulations below.

The D2Q9 LBM diffusion coefficient is known to be

$$D = \tfrac{1}{3}(\tau_\sigma - \tfrac{1}{2})$$ (125)

and its value is $D = 1/6$ $lu^2 ts^{-1}$ for $\tau_\sigma = 1$. This is directly analogous to the kinematic viscosity for single phase fluids (see Eq. (18)). The analytical solutions are of course dimensionally consistent, so we can use lattice units in them and compare the results directly to the LBM simulations. We illustrate MCMP-type simulations here, but comparable results can be obtained with the σ-component approach. The LBM simulations seem to match the analytical solutions well, though we have not conducted any formal error analysis.

Unbounded domain, plane instantaneous source

The analytical solution for diffusion in an infinite domain with a planar Dirac initial condition at the origin is (Crank 1975)

$$C = C_i + \frac{M_0}{2\sqrt{\pi D t}} e^{\frac{-x^2}{4Dt}},$$ (126)

with uniform initial concentration C_i and initial mass M_0. Figure 66 shows the analytical solution and the results of the lattice Boltzmann simulation for this problem at three different times. The agreement is excellent.
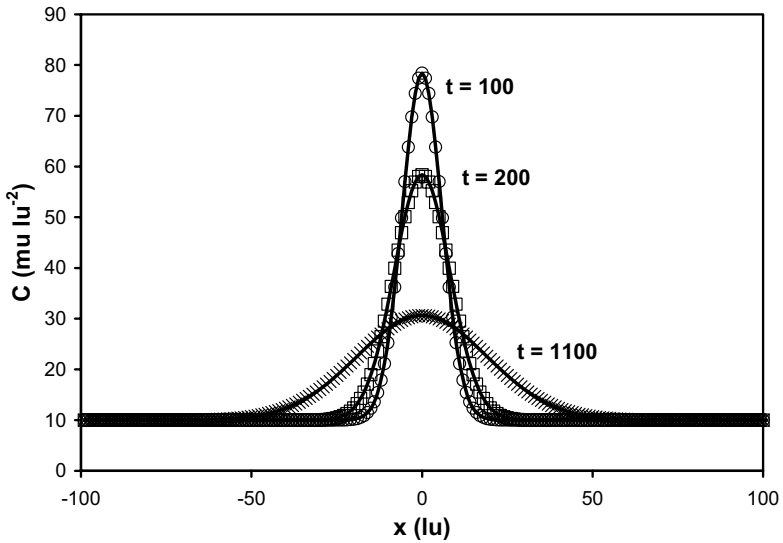
Figure 66. Diffusion from a plane instantaneous source in a 1-dimensional domain. Analytical solution shown as solid line, LBM solution as open symbols.

Extended initial condition

An analytical solution is also available for an initial condition that extends a distance ±h from the origin in an infinite domain (Crank 1975):

$$C = C_i + \frac{C_0}{2}\left[erf\left(\frac{h+x}{\sqrt{4Dt}}\right) + erf\left(\frac{h-x}{\sqrt{4Dt}}\right)\right], \qquad (127)$$

where *h* is half the width of the source. Figure 67 shows the LBM results and the analytical solution.
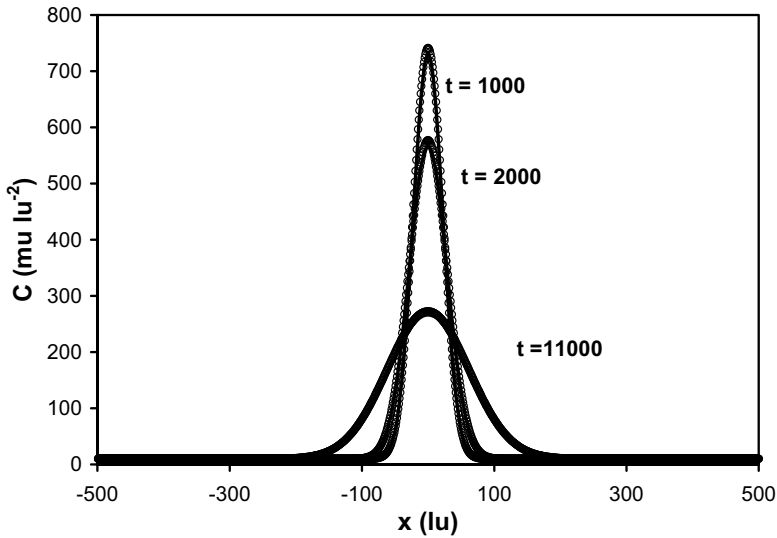
Figure 67. Diffusion from spatially extended initial condition in infinite 1-dimensional domain. Analytical solution shown as solid line, LBM solution as open symbols.

Bounded domain

Diffusion in a bounded, finite domain of length $2l$ subjected to an initial condition of width $2h$ centered at the origin has the following analytical solution (Crank 1975):

$$C = C_i + \frac{C_0}{2} \sum_{n=-\infty}^{\infty} \left[erf\left(\frac{h + 2nl - x}{\sqrt{4Dt}} \right) + erf\left(\frac{h - 2nl + x}{\sqrt{4Dt}} \right) \right]. \qquad (128)$$

The infinite series represents a superposition of the original diffusion process and reflections at the boundaries. Figure 68 presents a comparison of the analytical solution with the LBM simulation. This simulation indicates that bounceback alone is a suitable no flux boundary condition for diffusive processes. This is significant for the simulation of diffusive processes in porous media because it indicates that special solute boundary conditions are not needed at solid surfaces and the simplicity of bounceback is retained.
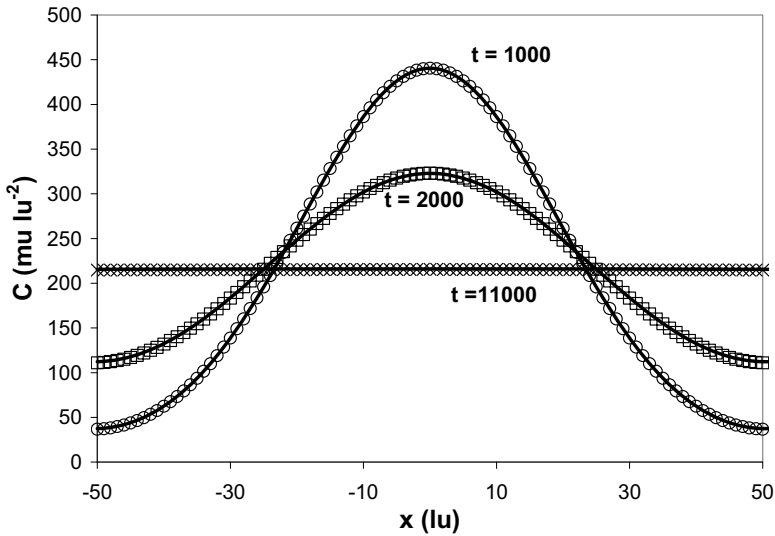
Figure 68. Diffusion in a bounded domain. Analytical solution shown as solid line, LBM solution as open symbols. Bounceback boundaries were applied to both 'fluids'. The results indicate that these are adequate boundaries for simulating solute transport.

2D test

As a final demonstration of the LBM's capabilities for simulating diffusion, we simulate 2-dimensional diffusion from an instantaneous point source in an infinite domain and compare the results with the analytical solution. The analytical solution is (Crank, 1975):

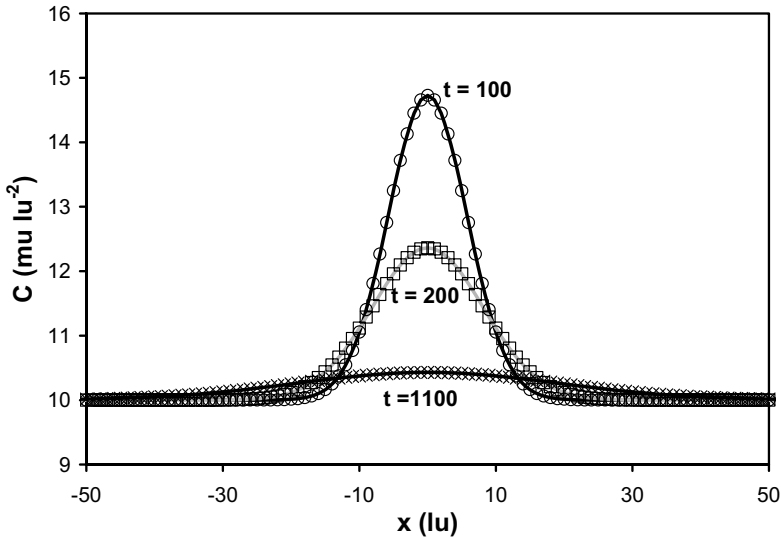$$C = C_i + \frac{M_0}{4\pi Dt} e^{-r^2/4Dt} . \tag{129}$$

Figure 69. Diffusion from an instantaneous point source in an infinite 2-dimensional domain. Analytical solution shown as solid line, LBM solution as open symbols.

8.5.2 Convection-Diffusion/Dispersion Equation (CDE)

The Convection-Diffusion/Dispersion Equation (CDE) governs the movement of solutes in moving fluids. LBM solute transport models generally solve a form of the convection-diffusion equation consistent with the model's dimensionality. The full CDE can be written

$$\frac{\partial C}{\partial t} = -v \nabla C + \nabla D \nabla C \qquad (130)$$

where v ($= v(x,y,z)$ for 3D), and D is the diffusion coefficient. We rely on the LBM's ability to solve this equation for most of our transport simulations in complex flows.

For a few simplified conditions, analytical solutions of the CDE are available (e.g., Hunt 1978; van Genuchten and Wierenga 1986; Leij et al. 1991). The solutions to the diffusion equation presented above represent one example of this; when velocity is zero (and the diffusion coefficient is constant), the 1-D CDE reduces to the diffusion equation (124).

In a one dimensional system where velocity v is single valued, the convection diffusion equation for constant D is

$$\frac{\partial C}{\partial t} = -v\frac{\partial C}{\partial x} + D\frac{\partial^2 C}{\partial x^2} \tag{131}$$

In one dimensional transport in simple Poiseuille flow and in porous media, the velocity is the mean channel or mean pore water velocity and D is the longitudinal dispersion coefficient, which quantifies variations from the mean water velocity and the magnitude of their effect in spreading solute concentrations. In porous media the mean pore water velocity v is given by the fluid flux q divided by the volumetric fluid content θ.

The breakthrough curve $C(x,t)$ is the temporal and spatial variation of concentration. In experimental practice often aimed at determining D for use in the CDE, it is common to use either a step change in concentration or a pulse of solute at the inflow. Clearly, more complex initial and boundary conditions often apply in engineered, natural, and 'accidental' systems. Figure 70 schematically illustrates a laboratory column system subjected to step and pulse changes in boundary concentrations and their breakthrough curves.

It is also common in experimental work to nondimensionalize time as 'pore volumes' $T = vt/L$, where L is the column length. This works in both fully and partially saturated porous media.
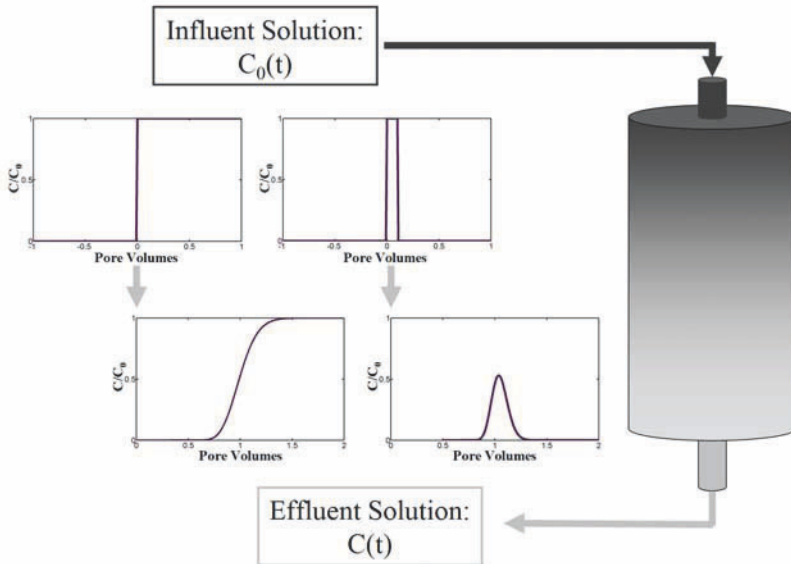
Figure 70. Typical experimental approach for measurement of temporal breakthrough curves. On the upper left a step change in relative concentration from 0 to 1 occurs at time 0 (0 pore volumes relative to solute injection) and longitudinal dispersion transforms this into a sigmoidal breakthrough curve (lower left) that elutes around the time one pore volume has been injected. The upper right shows a pulse injection that transforms into a breakthrough curve shaped approximately like a Gaussian probability density function.

It is useful to introduce the dimensionless Péclet number Pe = vL/D_m, where v is the velocity of the flow, L is a characteristic length, and D_m is the molecular diffusion coefficient. The Péclet number is a measure of the relative importance of transport by flow and diffusion. High Pe signifies the dominance of convection in the transport process while low Pe indicates that diffusion is dominant. The characteristic length is usually the grain size in packed bed studies.

A somewhat different Péclet number has also been applied to dispersive transport through porous media. We refer to this as the column Péclet number or the Brenner number Br. It is given by

$$Br = \frac{vL}{D} \tag{132}$$

where now L is the transport distance (e.g., the column length or other distance from source to measurement point) and D is the dispersion coefficient. The impact of the Brenner number on the shape of breakthrough curves is shown in Figure 71.
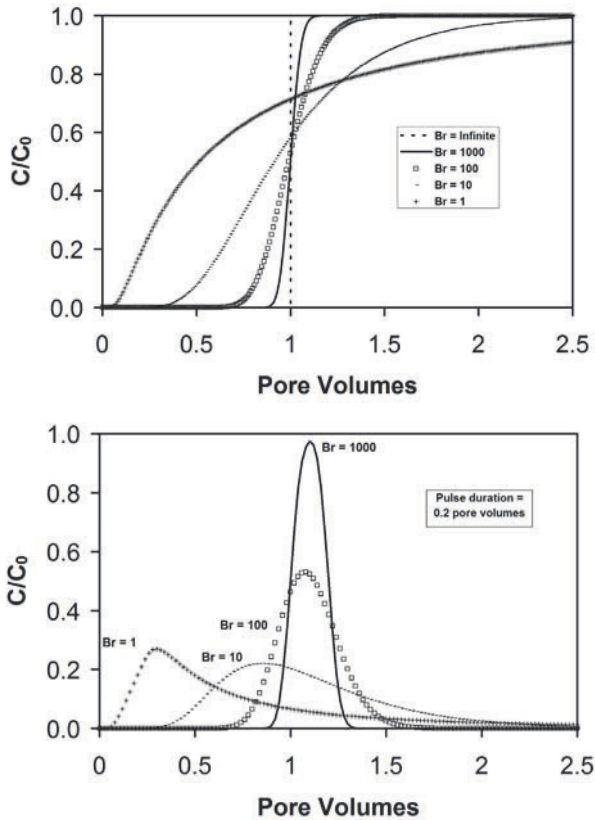

Figure 71. Effect of Brenner number on shape of breakthrough curves for step (top) and pulse (bottom) input functions.

Considerable effort has been devoted to evaluation of the behavior of different analytical solutions to the CDE at low Br where they tend to diverge (van Genuchten and Wierenga 1986). This is of significant practical interest for the use of short laboratory columns for example and for analysis of 'pre-asymptotic' solute transport in the field. Much like entry length effects for fluid flow in many applications, such effects may be more the rule rather than the exception.

There are also different methods of determining concentration when the diffusive/dispersive flux is appreciable (Kreft and Zuber 1978). This will occur when there is a high concentration gradient, when the diffusion/dispersion coefficient is high, and when the advective flux is low. Because the diffusive flux (and at least conceptually the dispersive flux) is driven by the concentration gradient dC/dx, there will be a flux in addition to the convective flux that cannot be accounted for by the normal 'resident' concentration C_r = (solute mass)/(fluid volume). A flux concentration C_f defined as (solute mass flux)/(fluid flux) can incorporate the diffusive/dispersive flux. If we sum the diffusive/dispersive and advective fluxes, C_f can be defined by

$$C_f v = -D \frac{\partial C}{\partial x} + C_r v, \qquad (133)$$

which implies

$$C_f = -\frac{D}{v} \frac{\partial C}{\partial x} + C_r. \qquad (134)$$

All indications are that the macroscopic CDE is inadequate for completely describing solute transport in porous media (it fails to reproduce observed heavy 'tails' in breakthrough curves and the apparent change in dispersion coefficient with travel time and distance). LBM and its ability to solve the convection-diffusion equation in complex velocity fields will probably make substantial contributions to the development of new theories such as the Fractional Advection Dispersion Equation (FADE) and Continuous Time Random Walk (CTRW) models for solute transport.

To explore LBM's abilities in solute transport, first we consider the case of a single-valued convective velocity and diffusion of the solute front. Next, we examine the case of Poiseuille flow, which leads to well known Taylor dispersion. Finally, we demonstrate transport in a simplified porous me-

dium for which empirical correlations allow dispersion to be predicted. These results establish the LBM's ability to simulate solute transport very well.

8.5.3 Propagation of a Diffusing Front

In a 1-dimensional uniform flow field, solute spreading occurs due to diffusion only. These simulations are interesting in that, with no walls or other obstacles, the fluid merely needs to be set into motion by setting the initial velocity; no driving force is needed to maintain a fixed velocity. The solute boundary conditions can be constant concentration or constant flux, which for small Br will significantly affect the measured breakthrough curve and will require the use of different analytical solutions for comparison.

For our example, we set the LBM simulation parameters to $v = 0.0066$ *lu ts*$^{-1}$, $D_m = 0.166$ *lu*^{2} *ts*$^{-1}$, and $L = 25$ *lu*, which leads to Br $= vL/D_m = 1$. We simulate four different combinations of boundary conditions and domain types corresponding to analytical solutions A1 through A4 of van Genuchten and Wierenga (1986): A1 and A2 are semi-infinite domains with constant concentration and constant flux boundary conditions respectively at $x = 0$, while A3 and A4 are finite domains of length L with $dC/dx|_{x=L} = 0$ (no diffusive flux BC described above) and constant concentration and constant flux boundary conditions respectively at $x = 0$ as in A1 and A2.

The results of the simulations are plotted against the appropriate analytical solutions in Figure 72. As in the comparison of the diffusion simulations to their analytical solutions above, these are not fitted results: the simulation parameters are known and are used directly in the analytical solutions. C_f is measured directly based on Eq. (134) in the form of Eq. (116). Agreement between the simulations and analytical solutions is excellent.
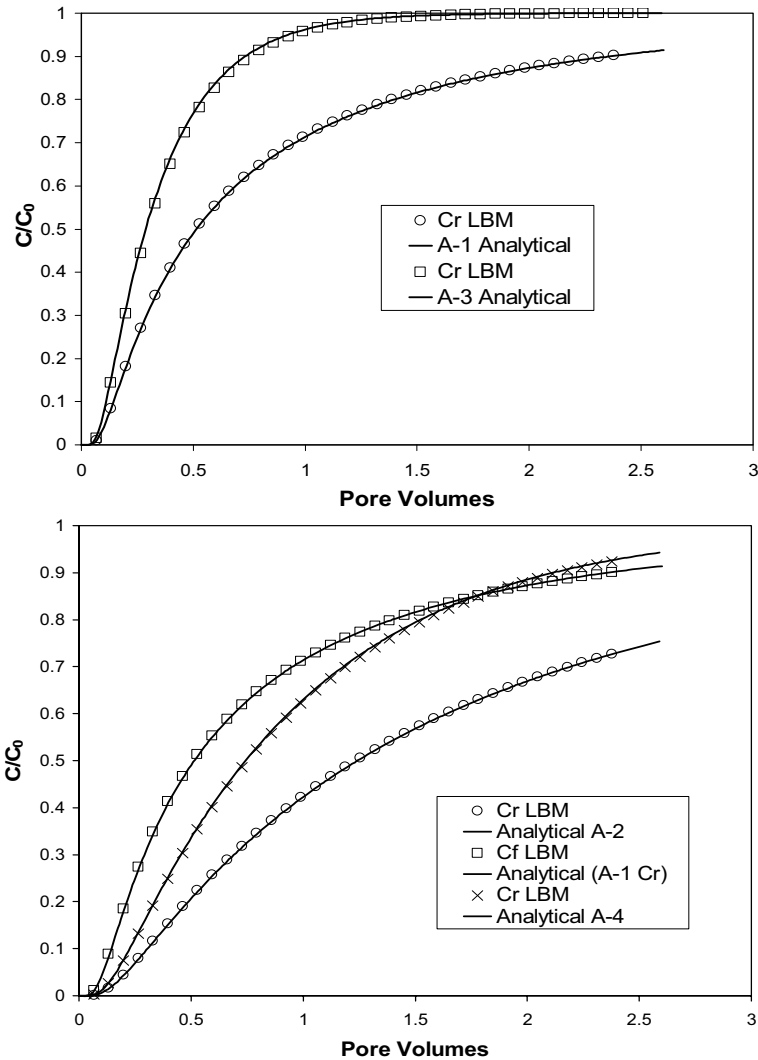
Figure 72. Breakthrough curves for diffusion at a propagating front for four different combinations of boundary conditions. LBM simulation results shown as open symbols, analytical solutions as solid lines. A1-A4 correspond to analytical solutions investigated by van Genuchten and Wierenga (1986). Both C_f and C_r are shown for the A2 boundary conditions, where the A1 solution gives the resident concentration C_r.

8.5.4 Taylor Dispersion

The next simplest demonstration of the capabilities of LBM for solute transport in porous media is simulation of Taylor-Aris dispersion (Taylor 1953, 1954; Aris 1956). This is dispersion in a straight-walled channel under Poiseuille flow as illustrated for a pipe in Figure 73.
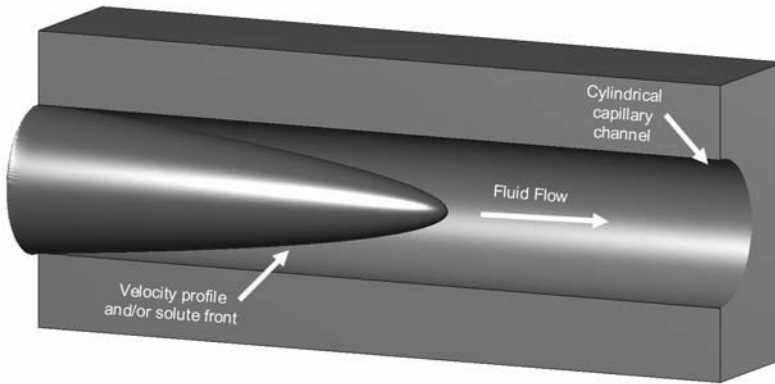

Figure 73. Taylor dispersion. Assuming no diffusion, the velocity profile and the solute front have identical parabolic shape. (Figure from Sukop and Perfect 2005.)

Here the velocity is a function of distance perpendicular to the flow direction (i.e., radial distance r in a pipe or y-coordinate in a slit oriented with the x-axis). Dispersion occurs as a result of the velocity variations across the tube or slit and due to diffusion. Taylor dispersion is of particular interest because we can compute the expected dispersion coefficient *a priori*.

Stockman (1997) demonstrated the ability of lattice gas models to simulate Taylor dispersion. We illustrate similar results here with LBM. The equation for the dispersion coefficient in a slit is (Stockman, 1997):

$$D = D_m + \frac{(2a)^2 v^2}{210D_m} \tag{135}$$

and the variables are the width of the channel *2a*, the average velocity of the flow *v*, and the molecular diffusion coefficient D_m. Note that the ratio $(2a)^2v^2/D_m^2 = Pe^2$ so Eq. (135) can be written as $D/D_m = 1 + Pe^2/210$. The divisor 210 is replaced by 48 for the cylindrical capillary case shown

above (Figure 73); under those conditions, the equation applies only between the limits $\sqrt{48}$ <<Pe<<L/a, but for the slit geometry we simulate here, the limits are $\sqrt{210}$ <<Pe<<L/a.

We exploit the LBM's tunable diffusion coefficient and noise-free nature to successfully compute dispersion in much smaller channels than possible in Stockman (1997). Figure 74 shows the analytical model Eq. (135) and the dispersion estimated by fitting LBM-computed breakthrough curves to the CDE (Toride et al. 1995). The trend and magnitude of the analytical solution are well represented.

The figure shows a plot with the analytical solution:

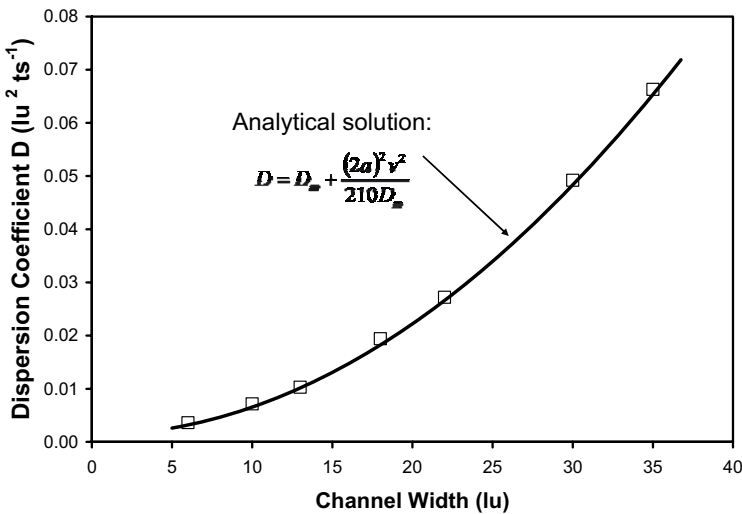$$D = D_m + \frac{(2a)^2 v^2}{210 D_m}$$

Figure 74. Results of Taylor-Aris dispersion simulations: Lattice Boltzmann (open symbols) versus analytical solution (solid line). Simulations were conducted in channels from 50 to 1500 *lu* in length and 6 to 35 *lu* in width with gravity-driven flow. Average velocity was held fixed at ≈ 0.0038 *lu* ts^{-1} by adjusting gravity. Reynolds numbers were <1, D_m ≈ 0.0013 *lu^2* ts^{-1}, Pe>17>$\sqrt{210}$, Br>15.

8.5.5 Dispersion in Packed Beds

Numerous papers present empirical correlations between the Péclet number and the dimensionless relative dispersion coefficient (D/D_m) (e.g.,

Pfannkuch 1963; Fried and Combarnous 1971; Bijeljic et al. 2004). One compilation of such data is presented in Figure 75. We can use such results to predict the dispersion coefficient based on grain size, flow velocity, and molecular diffusion coefficient.
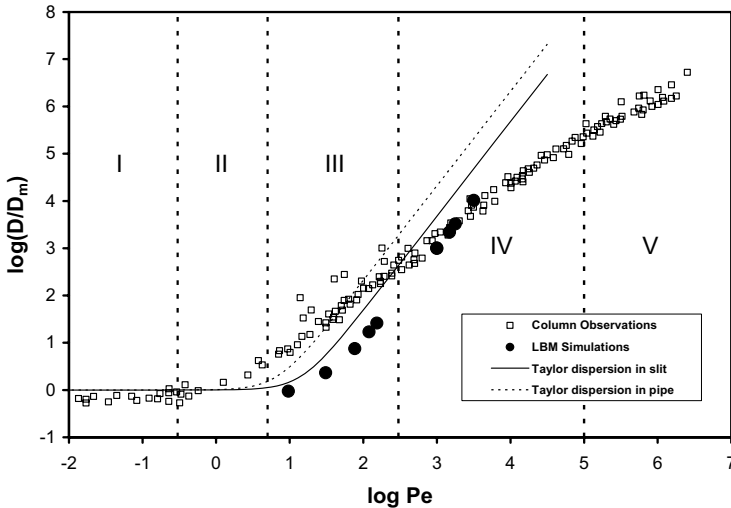

Figure 75. Empirical correlation between Péclet number and dimensionless dispersion coefficient (Modified from Fried and Combarnous, 1971, with permission). Regions I through V represent ranges of different behavior resulting from increasing relative importance of mechanical dispersion over diffusion. Observations mostly from packed columns, the results of LBM simulations as described below, and the Taylor-Aris theory predictions for slits and pipes are shown.

It is of considerable interest to be able to generate comparable results from lattice Boltzmann simulations, although we are not aware of efforts that are completely successful at this. The results of Noble (1997) using a 2-D FHP-type (triangular lattice) LBM model with specialized boundary conditions on the solids are perhaps the most notable; they show a considerable shift towards higher Pe as porosity increases and very good agreement with the observations from packed columns up to Pe$\approx 10^2$. Here we work with the domain of periodic disks shown in Figure 76 and standard bounceback boundaries. The porosity is 0.63. The flow boundaries are set to fully periodic and the flow is driven with gravity so that we effectively

simulate an infinite flow domain. We allow the flow to equilibrate for 1800 time steps and then initiate a step change in concentration. We compute the breakthrough curves at the downstream end of the domain. The results of a series of simulations like those in Figure 76 are shown on Figure 75 as large circles where the dispersion coefficients were estimated by inverse fitting to the CDE (Toride et al. 1995). The results follow the basic trends of the observations and the prediction of the Taylor theory for slits but fall short of being fully satisfactory. There are many potential reasons for this that need to be evaluated more fully. One major possibility is that 3-D simulations are necessary; these would allow lower porosities and more realistic flow fields to be simulated. Also, based on the results of Noble (1997), we anticipate that simulations at lower porosity would more closely approach the theoretical and observed curves.
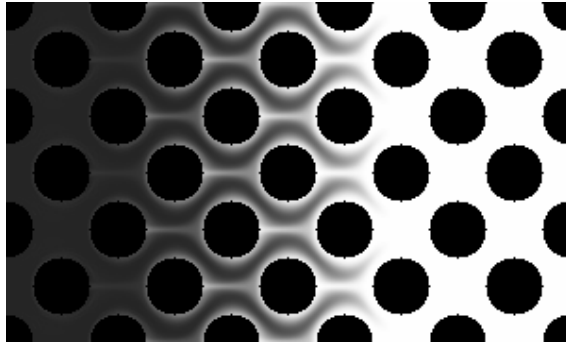

Figure 76. First 300 *lu* of a 1200×60 *lu²* simulation of solute transport after 12000 *ts*. Solute concentration proportional to gray scale. Three copies of the simulation have been tiled along the periodic boundaries at the top and bottom of the domain in this figure. The value of τ_σ is 0.503918 resulting in a small molecular diffusion coefficient of $D_m = 0.0013$ *lu² ts⁻¹* and the 'fingering' of the solute front.

8.5.6 Solute-induced Buoyancy

Dissolved substances can affect the density of fluids. As discussed above (Eq. (123)), these density effects can readily be incorporated into LBM by accounting for the density in the gravity term. We have applied this approach to the simulation of spontaneous mixing in a gravitational field (Figure 77) and to the solution of a classic thermohydrodynamic problem (Elder 1967) that has been widely adapted and used as a benchmark problem for models that simulate density-dependent ground water flow (Thorne

and Sukop 2004). Heroic numerical approaches using continuum equations have not resolved this problem unambiguously (eg. Frolkovič and De Schepper (2001)) and far simpler LBM may offer considerable advantages; this has yet to be explored.

Figure 77. Solute induced buoyancy. A fluid containing solute is placed above a solute-free fluid in a gravitational field.

8.6 Exercises

1. Use the MCMP 'active' solute component model to compute diffusion in a bounded domain. Set the interaction parameter to zero. Compare the results with the analytical model Eq.(128).

2. Repeat Exercise 1 with the 'passive' solute model. Discuss any differences between the models and their agreement with the analytical solution.

3. Use the passive solute model to solve a Taylor dispersion problem and compare the results to an analytical solution of the CDE. Insure that the Pèclet number is $> \sqrt{210}$ to satisfy Taylor's criteria and that the Brenner number is greater than 20 so that the choice of analytical solution is not critical.

4. Solve a density-driven mixing problem like that shown in Figure 77 except start with an initial condition that has the solute on the left half of the domain.

9 LBM for Macroscopic Porous Media

Although LBMs represent a significant advance in our ability to simulate the details of flow and transport in the complex pore spaces of natural and engineered channels, conduits, and porous media, the overall size of porous media that can be treated remains quite small at this time (See Succi (2001) for a discussion). LBM as typically applied may lead to improved understanding of fundamental aspects of fluid flow and solute transport in porous media, but our ability to solve large-scale problems using this type of pore scale LBM is limited. Dardis and McCloskey (1998a, 1998b) proposed a method that can surmount this difficulty by parameterizing a porous medium in terms of its solid density $n_s(\mathbf{x})$, or equivalently its porosity since $n_s = 1 - \phi$. Freed (1998) proposed a similar method and Balasubramanian et al. (1987) first applied the same concept to lattice gas models.

The method entails specifying a solid density value at each lattice node and simulating the porous medium by a 'probabilistic' bounce-back based on solid density. This provides a means of transcending the pore scale. Porosity can be varied arbitrarily so that assigning 0 solids density, to fractures and macropores for example, means that flow in those areas can still be simulated using full Navier-Stokes solutions. The more complex solute mixing phenomena that can arise in these features such as unsteady flows at higher Reynolds numbers and eddy diffusion can still be captured. Flow governed by Darcy's Law will simultaneously – and in a fully-coupled manner – occur in adjacent regions where the porosity is lower. We can think of the method in comparison to the standard LBM approach for porous media where we can take a binary image of a medium that is segregated into pore space and solids; in the Dardis and McCloskey method, we can use a gray scale image that reflects the porosity.

It is well known that the porosity is not the sole factor controlling the permeability of a medium. The Kozeny-Carman equation gives the permeability as a function of particle diameter and porosity for example. In many porous media the permeability is anisotropic and hence must be controlled by additional details of grain orientation, etc. It is likely that anisotropy could easily be incorporated into the method proposed by Dardis and

McCloskey, but this has not been done to our knowledge. Finally, we are unaware of work other than Thorne and Sukop (2004) that considers solute transport under this method with LBM. In that paper we used the method to simulate the benchmark Elder problem for solute-induced, density dependent transport in a porous medium. More testing of this method is needed. Yang et al. (1998, 1999) did transport simulations using probabilistic bounceback with a lattice gas model that probably will provide guidance for similar LBM work.

To implement the LBM for macroscopic porous media, consider the traditional collision step as a second intermediate step after streaming, denoted by f^{**}

$$f_a^{**}(\mathbf{x},t+\Delta t) = f_a^{*}(\mathbf{x},t) + \frac{1}{\tau}\left(f_a^{eq}(\mathbf{x},t) - f_a^{*}(\mathbf{x},t)\right) \tag{136}$$

for $0 \leq a \leq 8$. Then the porous medium step has the form

$$\begin{aligned} f_a(\mathbf{x},t+\Delta t) = \\ f_a^{**}(\mathbf{x},t+\Delta t) + n_s\left[f_{a'}^{**}(\mathbf{x}+\mathbf{e}_a\Delta t,t+\Delta t) - f_a^{**}(\mathbf{x},t+\Delta t)\right] \end{aligned} \tag{137}$$

for $0 \leq a \leq 8$, where a' is the index of the direction opposite $\mathbf{e}_a$.

Note that for $n_s = 0$ we recover the normal free-fluid model and for $n_s = 1$, we have a bounce-back-like condition that effectively makes the medium impermeable. For values of n_s between 0 and 1 we have partial or 'probabilistic' bounce-back (Figure 78).
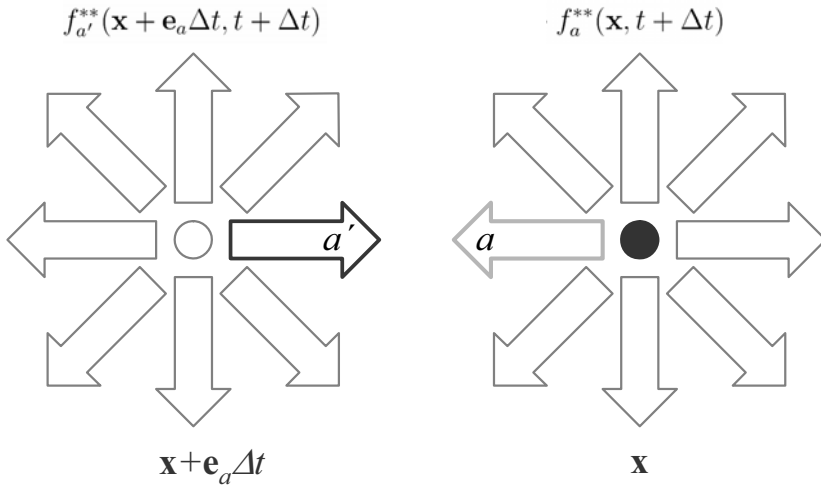
Figure 78. Probabilistic bounceback. A fraction n_s of the density difference between f_a'–f_a is added to the results of the standard collision.

Following is pseudocode for computing and applying the n_s term. It is performed right after the core collision/bounceback loop.

```
// Macroscopic porous media mechanism.
// Compute the solid density term (nsterm).
for( j=0; j<LY; j++)
{
  jp = ( j<LY-1)?( j+1):( 0    );
  jn = ( j>0    )?( j-1):( LY-1);

  for( i=0; i<LX; i++)
  {
    ip = ( i<LX-1)?( i+1):( 0    );
    in = ( i>0    )?( i-1):( LX-1);

    if( !is_solid_node[j][i])
    {
      nstermij[1] = ns*( f[j ][ip][3] - fij[1]);
      nstermij[2] = ns*( f[jp][i ][4] - fij[2]);
      nstermij[3] = ns*( f[j ][in][1] - fij[3]);
      nstermij[4] = ns*( f[jn][i ][2] - fij[4]);
      nstermij[5] = ns*( f[jp][ip][7] - fij[5]);
      nstermij[6] = ns*( f[jp][in][8] - fij[6]);
      nstermij[7] = ns*( f[jn][in][5] - fij[7]);
      nstermij[8] = ns*( f[jn][ip][6] - fij[8]);
    }
  }
}

// Apply the solid density term (nsterm).
for( j=0; j<LY; j++)
```

```
for( i=0; i<LX; i++)
  if( !is_solid_node[j][i])
    for( a=1; a<9; a++) { fij[a] += nstermij[a];}
```

9.1 Analytical Solutions

If a damping term proportional to the velocity is introduced into the Navier Stokes equation, we can write (Balasubramanian et al. 1987):

$$v \frac{d^2u}{dx^2} - \alpha u = \frac{1}{\rho} \frac{dP}{dy} \; . \tag{138}$$

For boundary conditions $u(-a) = u(a) = 0$, there is an analytical solution related to the Poiseuille equation (4) for the velocity profile in a channel containing a porous medium (at least at low solid 'scatterer' density). Under these conditions,

$$u(x) = -\frac{G^*}{\alpha\rho}\left[1 - \frac{\cosh[r(x-a)]}{\cosh(ra)}\right], \tag{139}$$

where α is a damping coefficient proportional to the scatterer density, G^* is the pressure gradient, and $r = \sqrt{\alpha/v}$ (Balasubramanian et al. 1987). Although it is not clear from inspection of Eq. (139), $\alpha \to 0$ returns the standard Poiseuille equation. Figure 79 compares LBM simulations using the macroscopic porous medium approach with Eq. (139). Similar results can be found in Balasubramanian et al. (1987) and Dardis and McCloskey (1998a).
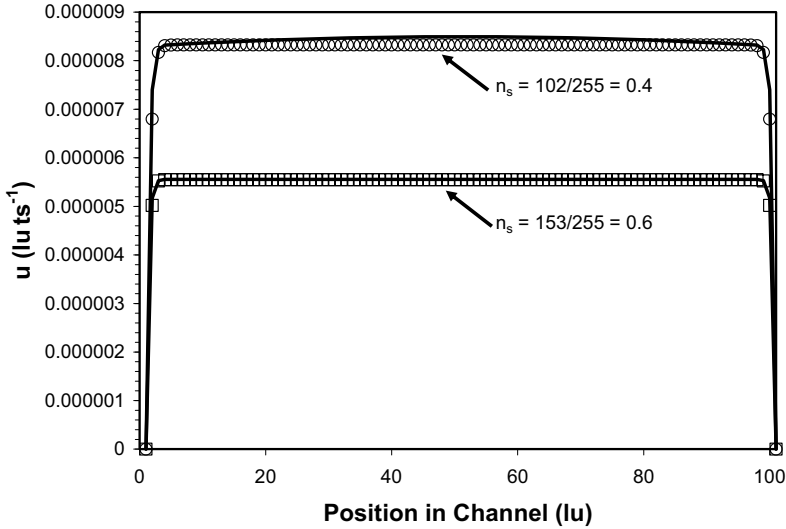
Figure 79. Simulations of flows in a walled channel filled with porous media of two different porosities in a 1001×101 lu^2 domain. The flow is driven by gravity and $g = 6.67×10^{-6}$ lu ts^{-2}. The density is $\rho = 1$ mu lu^{-2}. Open symbols show simulations. Analytical solutions from Eq. (139) shown as solid lines.

It is also possible to solve Eq. (138) for zero velocity gradient conditions throughout the domain: $d^2u/dx^2 = 0$; that is, there are no walls at which the fluid velocity must adjust to become zero. The solution is almost trivial as u is now a constant determined only by the pressure or head gradient across the domain, the damping coefficient α and the fluid density. The solution is $u = -G^*/(\alpha\rho)$, which can be related to Darcy's law. It still remains to link α with the solids density. This can be done theoretically and yields the result $\alpha = 2n_s$ (Balasubramanian et al. 1987). The simple analytical solution $u = -G^*/(\alpha\rho)$ also applies at adequate distance from a wall. So for example, examining the results for $n_s = 0.6$ in Figure 79, we can compute the velocity away from the walls as $u = -(-6.67×10^{-6}$ mu ts^{-2} $lu^{-1})/(2*0.6 *1$ mu $lu^{-2}) = 0.00000564$ lu ts^{-1}, which agrees with the results in the figure and we use the 2-D pressure and density. The dimensions of α must be T^{-1} in accordance with Eq. (138).

9.2   Relation to Darcy's Law

In the context of this simple analytical solution, u is the familiar Darcy velocity that is obtained from Darcy's law: $u = -Kdh/dx$. Writing this in terms of pressure gradient we have $u = -K/(\rho g)dP/dx$ and finally, replacing the conductivity K with the intrinsic permeability $K = kg/v$, we obtain $u = -k/(v\rho) \, dP/dx$. Equating this with the simple analytical solution above gives $k = v/\alpha = v/2n_s$. Readers should verify this by simulating flows with different porosity/scatterer densities and demonstrating agreement with Darcy's law (see Exercises).

9.3   Application of Percolation Theory

Figure 80 shows a plot of this theoretical permeability as a function of the scatterer density ($n_s = 1-\phi$) along with some simulation results. This function should apply only at low n_s. For higher n_s and for gravity-driven flows Dardis and McCloskey (1998a) showed that this method of simulating porous media fit the expectations of percolation theory very well. In a 2-D domain, randomly placing obstacles on a lattice leads to a situation in which 'percolation' (pore connectivity between opposite sides of the domain) exists until the solids density becomes too great (the porosity becomes too small). For 'site' percolation, the percolation threshold is reached when the number of blocked sites reaches ½ of the total number, corresponding to a percolation threshold porosity ϕ_c of 50 percent. Near the percolation threshold, the permeability changes rapidly with the porosity according to

$$k \propto \left(\phi - \phi_c\right)^{1.3} \qquad (140)$$

where the exponent 1.3 is a universal value for conductivity controlled by percolation (Stauffer and Aharony 1994). Using a gravity-driven lattice Boltzmann model on a triangular lattice, Dardis and McCloskey (1998a) observed a percolation threshold critical porosity ϕ_c of 0.5 and demonstrated agreement with this exponent.

Our own results with the D2Q9 lattice also seem to fit a percolation model but do not appear to show a percolation threshold appreciably different from $\phi_c = 0$. That is, unlike Dardis and McCloskey (1998a), we can simulate arbitrarily low porosities. Fundamental differences in the lattice struc-

ture may be responsible for this contrast, but this result seems intuitively reasonable to us; unlike the random closing of potential paths by obstacles in a standard percolation model, in this method of simulating porous media the porosity is reduced uniformly and the fluid's motion is damped but nowhere blocked until n_s approaches 1.

The significant difference we observed between gravity- and pressure gradient-driven flows in this macroscopic porous media LBM was surprising but may be related to the interaction of gravity with the n_s mechanism in the code or the compressibility of the LBM fluid; the density and velocity are uniform in the gravity-driven case but they vary inversely with each other in the pressure-driven case; this is necessary to maintain steady mass flux. It seems clear that there is a need for considerable additional work on this topic.
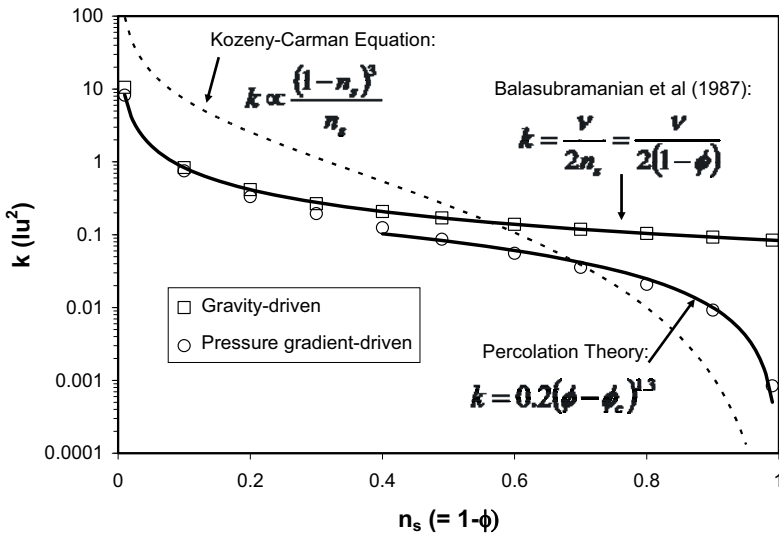

Figure 80. Theoretical intrinsic permeability following Kozeny-Carman, Balasubramanian et al. (1987) and percolation theory (Dardis and McCloskey 1998a) along with simulation results. Open circles show pressure gradient-driven results. A pressure gradient of 10^{-6} was applied across a 1001×101 lu^2 domain (periodic in y) with $\tau = 1$. Open squares show gravity-driven results. Gravity was $g = 10^{-6}$.

9.4 Dual Continuum Models

Perhaps the most exciting aspect of this type of LBM modeling is the potential for close coupling of Navier Stokes solutions in large open spaces such as macropores, fractures, conduits, and caves with Darcy's Law solutions in adjacent porous materials. Very little appears to have been done with LBM in this realm so far. Dardis and McCloskey (1998a) solved for the flow in a single fracture between two porous walls and determined the permeability tensor by rotating the domain relative to the pressure gradient.

Figure 81 shows a simple conduit traversing a porous medium. A pres-

Figure 81. Pressure field in fractured porous medium analog. Porosity set to 0.8 to accentuate behavior of porous medium. $\tau = 0.54$. The conduit (gray) plays an important role in determining the pressure field but there are significant gradients across the porous domain.

sure gradient is applied across the domain from left to right. The conduit exerts a strong influence on the pressure field inside the domain. In general, there are zones of reduced pressure behind edges that protrude into the flow and elevated pressures where fluid impinges on a corner.

In Figure 82, we show the fluid velocity vectors for a portion of the domain. The flow is laminar with Re approaching 2.
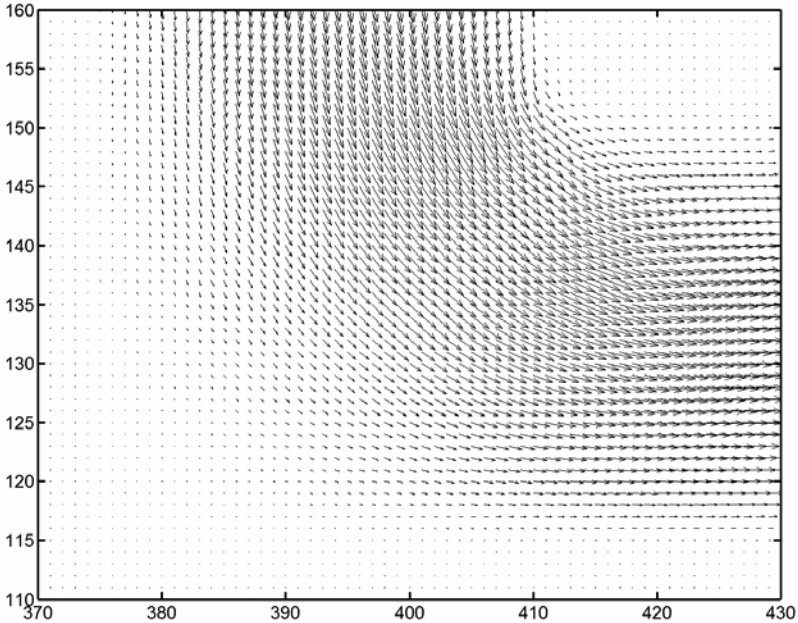
Figure 82. Fluid velocity vectors from the lower right part of the domain in Figure 81.

Finally, Figure 83 provides an alternative view of the the flow domain that emphasizes flow in the porous medium. Flow that has crossed nearly the entire domain converges on the lower left corner, while high velocities develop across a thin protrusion that has a pressure shadow behind it at the upper right.
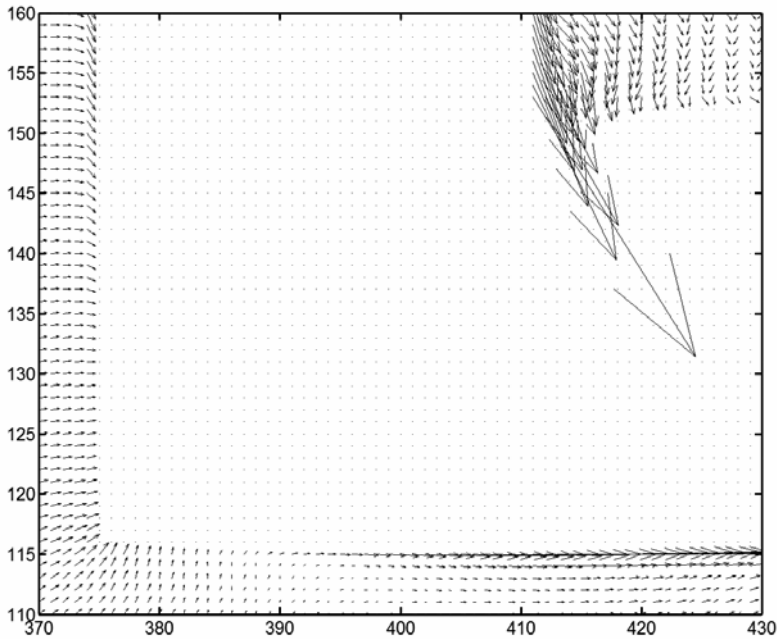
Figure 83. Same area as Figure 82 with conduit velocities masked to emphasize flows in the porous medium. High velocities develop across the protruding edge due to the low pressure behind it and the thinness of the medium at that point. Note the flows converging on the lower left corner; comparison with Figure 81 shows that this fluid has traversed most of the domain inside the porous medium.

9.5 Exercises

1. Plot Eq. (139) for a series of α values along with the Poiseuille equation (Eq. (4)) and demonstrate graphically that the results of the equations are the same as α approaches 0.

2. Simulate Darcy velocities in a series of porous media with different porosities. Compare the results to Darcy's law predictions.

3. Use the macroscopic porous media LBM model to solve for the velocity field in a domain that has a region with $n_s = 3/255$ (= 0.0117) in its

upper portion, a channel (where $n_s = 0$) of width 10 *lu* in the middle, and a region where $n_s = 6/255$ ($= 0.0234$) in the lower portion. Use $g = 5{\times}10^{-6}$ and $\tau = 1$. (The 255 factor accounts for the 8-bit gray scale images used to load the variable n_s data. In Microsoft Paint®, you can use Colors, Edit Colors, Define Custom Colors, choose 0 saturation, use the slider bar to choose the gray scale level, and then Add to Custom Colors. Finally, select the new custom color and click OK. This will make the custom color the current color and you can paint it into the domain. Return to the Colors options to select another custom color.)

10 Conclusions

It is our hope that this book has provided a useful starting point for those interested in applying lattice Boltzmann models. The variety of fluid behaviors that can be simulated within a single conceptual framework is remarkable. Enhancements of the basic methods and more applications are continually appearing and will appear in the future. LBM occupies an important niche between molecular dynamics and traditional continuum approaches in the realm of simulation models. In our opinion, LBM has an important advantage of simplicity. The continuing growth of computational power suggests that the future of LBM is bright; we wish our readers the best of success in their efforts to apply the methods and look forward to seeing the results of those efforts.

References

Adamson AW, Gast AP (1997) Physical Chemistry of Surfaces. John Wiley & Sons, Inc New York

Angelopoulos AD, Paunov VN, Burganos VN, Payatakes AC (1998) Lattice Boltzmann simulation of nonideal vapor-liquid flow in porous media. Phys Rev E 57:3237-3245

Ansumali S, Karlin IV (2002) Kinetic boundary conditions in the lattice Boltzmann method. Phys Rev E 66:026311

Appert C, Zaleski S (1990) Lattice gas with a liquid-gas transition. Phys Rev Lett 64:1-4

Aris R (1956) On the dispersion of a solute in a fluid flowing through a tube. Proc R Soc London A 235:67-77

Atkins PW (1978) Physical Chemistry. WH Freeman and Company, San Francisco

Balasubramanian K, Hayot F, Saam WF, 1987. Darcy's law from lattice-gas hydrodynamics. Phys Rev A 36:2248-2253

Berkowitz B, Ewing RP (1998) Percolation theory and network modeling applications in soil physics. Surveys in Geophys 19:23-72

Bijeljic B, Muggeridge AH, Blunt MJ (2004) Pore-scale modeling of longitudinal dispersion. Wat Resour Res 40:W11501 doi:10.1029/2004WR003567

Boltzmann L (1964/1995) Lectures on gas theory (translated by SG Brush). Dover edition, Dover Publications New York. Originally published 1964, University of California Press Berkeley

Briant AJ, Papatzacos P, Yeomans JY (2002) Lattice Boltzmann simulations of contact line motion in a liquid-gas system. Phil Trans R Soc Lond 360:485-495

Buckles JJ, Hazlett RD, Chen S, Eggert KG, Grunau DW, Soll WE (1994) Toward improved prediction of reservoir flow performance. Los Alamos Science 22:112-121

Cates ME, Stratford K, Adhikari R, Stansell P, Desplat JC, Pagonabarraga I, Wagner AJ, (2004) Simulating colloid hydrodynamics with lattice Boltzmann methods. J Phys Condensed Matter 16:S3903-S3915

Chapman S, Cowling TG (1990) The mathematical theory of non-uniform gases: an account of the kinetic theory of viscosity, thermal conduction, and diffusion in gases, 3rd ed. prepared in co-operation with D. Burnett. Cambridge University Press Cambridge New York. Originally published 1970, Cambridge University Press, Cambridge.

Chen H (1993) Discrete Boltzmann systems and fluid flows. Computers in Phys 7:632-637

Chen S, Doolen GD (1998) Lattice Boltzmann method for fluid flows. Annual Rev Fluid Mech 30:329-364

Chen S, Martínez D, Mei R (1996) On boundary conditions in lattice Boltzmann methods. Phys Fluids 8:2527-2536

Chopard B, Dupuis A (2003) A mass conserving boundary condition for lattice Boltzmann models. Int J Modern Phys B 17:103-107

Cook BK, Noble DR, Williams JR (2004) A direct simulation method for particle-fluid systems. Engineering Computations 21:151-168

Crank J (1975) The Mathematics of Diffusion, 2nd ed Clarendon Press Oxford

Dardis O, McCloskey J (1998a) Permeability porosity relationships from numerical simulations of fluid flow. Geophys Res Let 25:1471-1474

Dardis O, McCloskey J (1998b) Lattice Boltzmann scheme with real numbered solid density for the simulation of flow in porous media. Phys Rev E 57, 4834-4837

Di Pietro LB, Melayah A, Zaleski S (1994) Modeling water infiltration in unsaturated porous media by interacting lattice gas-cellular automata. Wat Resour Res 30:2785-2792

Douglas JF, Gasiorek JM, Swaffield JA (2001) Fluid Mechanics, 4th edn. Prentice Hall, Harlow England.

Drazer G, Koplik J (2001) Tracer dispersion in two-dimensional rough fractures. Phys Rev E 63:056104-1-056104-11

Dupin MM, Spencer TJ, Halliday I, Care CM (2004) A many-component lattice Boltzmann equation simulation for transport of deformable particles. Phil Trans Royal Soc London A 362:1885-1914

Elder JW (1967) Steady free convection in a porous medium heated from below J Fluid Mech 27:29-48

Fan L, Fang H, Lin Z (2001) Simulation of contact line dynamics in a two-dimensional capillary tube by the lattice Boltzmann model. Phys Rev E 63: 051603, 6 pp

Fang H, Lin Z, Wang Z (1998) Lattice Boltzmann simulation of viscous fluid systems with elastic boundaries. Phys Rev E 57:R25-R28

Flekkøy EG (1993) Lattice Bhatnagar-Gross-Krook models for miscible fluids. Phys Rev E 47:4247-4257

Freed DM, 1998, Lattice Boltzmann method for macroscopic porous media modeling, Int J Modern Phys C 9:1491-1503

Fried JJ, Combarnous MA (1971) Dispersion in porous media. Adv Hydrosci 7:169-282

Friedman SP (1999) Dynamic contact angle explanation of flow rate-dependent saturation-pressure relationships during transient liquid flow in unsaturated porous media. J Adhesion Sci Technol 13:1495-1581

Frisch U, Hasslacher B, Pomeau Y (1986) Lattice-gas automata for the Navier-Stokes equation. Phys Rev Let 56:1505-1508G

Frolkovič P, De Schepper H (2001) Numerical modelling of convection dominated transport coupled with density driven flow in porous media. Adv Wat Res 24:63-72.

Gallivan MA, Noble DR, Georgiadis JG, Buckius RO (1997) An evaluation of the bounce-back boundary condition for lattice Boltzmann simulations. Int J Num Meth Fluids 25:249-263

Ginzbourg I (1994) Les problèmes de conditions aux limites dans les méthodes de gaz sur réseaux à plusieurs phases. These de doctorat de l'Universite Paris 6

Ginzbourg I, d'Humieres D (1996) Local second-order boundary methods for lattice Boltzmann methods. J Stat Phys 84:927-971

Ginzbourg I, d'Humieres D (2003) Multireflection boundary conditions for lattice Boltzmann models. Phys Rev E 68: 066614

Grubert D (1997) Using the FHP-BGK-Model to get effective dispersion constants for spatially periodic model geometries. Int J Mod Phys C 8:817-825

Gutfraind R, Ippolito I, Hansen A (1995) Study of tracer dispersion in self-affine fractures using lattice gas automata. Phys Fluids 7:1938-1948

Halliday D, Resnick R (1978) Physics, 3rd Ed. J. Wiley and Sons, New York

Harris S (1971) An introduction to the theory of the Boltzmann equation. Holt, Rienhart and Winston, Inc., New York

He X, Doolen GD (2002) Thermodynamic foundations of kinetic theory and lattice Boltzmann models for multiphase flows. J Stat Phys 107:309-328

Hirschfelder JO, Curtiss CF, Bird RB (1954/1965) Molecular theory of gases and liquids. Wiley, New York

Hunt B (1978) Dispersive sources in uniform ground-water flow. ASCE J Hydr Div 104(HY-1):75-85

Hyväluoma J, Raiskinmäki P, Jäsberg A, Koponen A, Kataja M, Timonen J (2004) Evaluation of a lattice-Boltzmann method for mercury intrusion porosimetry simulations. Future Generation Computer Systems 20:1003-1011

Inamuro T, Yoshino M, Ogino F (1995) A non-slip boundary condition for lattice Boltzmann simulations. Phys Fluids 7:2928-2930

Jalbert M, Dane JH, Abriola LM, Pennell KD (2000) Nondimensional evaluation of tracer sensitivity to density effects. Ground Water 38: 226-233

Kittel C (1958/2004) Elementary statistical physics. Dover Publications Inc., Mineola NY

Kreft A, Zuber A (1978) On the physical meaning of the dispersion equation and its solutions for different initial and boundary conditions. Chem Eng Sci 33:1471-1480

Ladd AJC (1993) Short-time motion of colloidal particles: numerical simulation via a fluctuating lattice-Boltzmann equation. Phys Rev Lett 70:1339-1342

Ladd AJC (1994a) Numerical simulations of particulate suspensions via a discretized Boltzmann equation: Part 1. Theoretical foundation. J Fluid Mech 271:285-309

Ladd AJC (1994b) Numerical simulations of particulate suspensions via a discretized Boltzmann equation. Part 2. Numerical results. J Fluid Mech 271:311-339

Ladd AJC, Verberg R (2001) Lattice-Boltzmann simulations of particle-fluid suspensions. J Stat Phys 104:1191-1251

Langaas K, Papazacos P (2001) Numerical investigations of the steady state relative permeability of a simplified porous medium. Transport in Porous Media 45:241-266

Langmuir I (1938) Replusive forces between charged surfaces in water, and the cause of the Jones-Ray effect. Science 88, No. 2288:430-432

Leij FJ, Skaggs TH, van Genuchten MTh (1991) Analytical solutions for solute transport in three-dimensional semi-infinite porous media. Wat Resour Res 27:2719-2733

Lenormand R, Touboul E, Zarcone C (1988) Numerical models and experiments on immiscible displacements in porous media. J Fluid Mech 189:165-187

Lishchuk SV, Care CM, Halliday I (2003) Lattice Boltzmann algorithm for surface tension with greatly reduced microcurrents. Phys Rev E 67:1-5

Luo L-S (2000) Some recent results on discrete velocity models and ramifications for the lattice Boltzmann equation. Comp Phys Comm 129:63-74

Luo L-S (1998) Unified Theory of Lattice Boltzmann Models for Nonideal Gases Phys Rev Lett 81:1618-1621

Maier RS, Bernard RS, Grunau DW (1996) Boundary conditions for the lattice Boltzmann method. Phys Fluids 8:1788-

Martys NS (2001) A classical kinetic theory approach to lattice Boltzmann simulation. Int. J. Modern Phys. C 12:1169-1178

Martys NS, Chen H (1996) Simulation of multicomponent fluids in complex three-dimensional geometries by the lattice Boltzmann method. Phys Rev E 53:743-750

Martys NS, Douglas JF (2001) Critical properties and phase separation in lattice Boltzmann mixtures. Phys Rev E 63:1-18

Mattis, DC 2003. Statistical mechanics made simple: a guide for students and researchers, World Scientific Publishing Co. Pte. Ltd., Singapore

Nie X, Doolen GD, Chen S (1998) Lattice Boltzmann simulations of fluid flows in MEMS. arXiv:comp-gas /9806001 v1

Noble DR (1997) Lattice Boltzmann study of the interstitial hydrodynamics and dispersion in steady inertial flows in large randomly packed beds. PhD dissertation, University of Illinois at Urbana-Champaign, Urbana, Illinois

Noble DR, Chen S, Georgiadis JG, Buckius RO (1995) A consistent hydrodynamic boundary condition for the lattice Boltzmann method. Phys Fluids 7:203-209

Nouar C, Oldrouis M, Salem A, Legrand J (1995) Developing laminar flow in the entrance region of annuli-review and extension of standard resolution methods for the hydrodynamic problem. Int J Engng Sci 33:1517-1534

Nourgaliev RR, Dinh TN, Theofanous TG, Joseph D (2003) The lattice Boltzmann equation method: theoretical interpretation, numerics and implications. Int J Multiphase Flow 29:117-169

Or D, Tuller M (2002) The role of cavitation in porous media desaturation under tension. Wat Resour Res 38: doi: 10.1029/2001WR000282.

Pan C, Hilpert M, Miller CT (2004) Lattice-Boltzmann simulation of two-phase flow in porous media. Wat Resour Res 40: doi: 10.1029/ 2003WR002120

Perea-Reeves SJ, Stockman HW (1997) A lattice-gas study of dispersion in alveolated channels. Chem Eng Sci 52:3277-3286

Perfect E, Sukop MC (2001) Models relating solute dispersion to pore space geometry: a review. In: Physical and chemical processes of water and solute transport/retention in soils. Sparks DL, Selim HM (eds) Soil Sci Soc Am Special Pub. 56, Madison, Wisconsin, pp 77-146

Pfannkuch HO (1963) Contribution a l' etude des deplacements de fluiedes miscibles dans un milieu poreux. Rev Inst Fr Pet 18:215-270

Qian YH, Succi S, Orszag SA (1995) Recent advances in lattice Boltzmann computing. Ann Rev Comp Phys 30:195-242

Qian, YH, d'Humieres D, Lallemand P (1992) Lattice BGK models for Navier-Stokes equation. Europhys Lett 17:479-484

Raiskinmäki P, Shakib-Manesh A, Jäsberg, Koponen A, Merikoski J, Timonen J (2002) Lattice-Boltzmann simulations of capillary rise dynamics. J Stat Phys 107:143-157

Raiskinmäki P, Koponen A, Merikoski J, Timonen J (2000) Spreading dynamics of three-dimensional droplets by the lattice Boltzmann method. Comp Materials Sci 18, 7-12

Rothman DH (1988) Cellular-automaton fluids: a model for flow in porous media. Geophysics 53:509-518

Rothman DH, Zaleski S (1997) Lattice-gas cellular automata. Cambridge University Press, Cambridge New York Melbourne

Shan X, Chen H (1993) Lattice Boltzmann model for simulating flows with multiple phases and components. Phys Rev E 47:1815-1819

Shan X, Chen H (1994) Simulation of nonideal gases and liquid-gas phase transitions by the lattice Boltzmann equation. Phys Rev E 49:2941-2948

Shan X and Doolen G (1996) Diffusion in a multicomponent lattice Boltzmann equation model. Phys Rev E 54:3614-3620

Skordos PA (1993) Initial and boundary conditions for the lattice Boltzmann method. Phys Rev A 48:4823-4842

Smith AMO (1960) Remarks on transitions in a round tube. J Fluid Mech 7:565-576

Soll WE, Chen SY, Eggert KG, Grunau DW, Janecky DR (1994) Application of the lattice Boltzmann/lattice gas technique to multi-fluid flow in porous media. Computational Methods in Water Resources X:991-999

Soll WE, Eggert KE, Grunau DW, Schafer-Perini al. (1994) A study of multiphase flow in fractured porous media using a microscale lattice Boltzmann approach. Computational Methods in Water Resources X:693-700

Stauffer D, Aharony A (1994) Introduction to Percolation Theory. CRC Press, Boca Raton London New York Washington D.C.

Stockman HW (1997) A lattice-gas study of retardation and dispersion in fractures: assessment of errors from desorption kinetics and buoyancy. Wat Resour Res 33:1823-1831

Stockman HW, Johnson J, Brown SR (2001) Mixing at fracture intersections: influence of channel geometry and the Reynolds and Péclet numbers. Geophys Res Lett 28:4299-4302

Stockman HW, Glass RJ, Cooper C, Rajaram H (1998) Accuracy and computational efficiency in 3D dispersion via lattice Boltzmann: models for dispersion in rough fractures and double-diffusive fingering. Int J Modern Phys C 9:1545-1557

Stockman HW, Cooper C, Li C, Perea-Reeves SJ (1997a) Practical application of lattice-gas and lattice Boltzmann methods to dispersion problems. InterJournal Complex Systems paper 90 (http://www.interjournal.org/)

Stockman H, Li C, Wilson J (1997b) A lattice-gas and lattice Boltzmann study of mixing at continuous fracture junctions: importance of boundary conditions. Geophys Res Lett 24:1515-1518

Succi S (2001) The lattice Boltzmann equation for fluid dynamics and beyond. Clarendon Press, Oxford

Sukop MC, Or D (2005) Lattice Boltzmann method for homogeneous and heterogeneous cavitation. Phys Rev E 71:046703

Sukop MC, Or D (2004) Lattice Boltzmann method for modeling liquid-vapor interface configurations in porous media. Wat Resour Res 40:W01509, doi: 10.1029/2003WR002333

Sukop MC, Or D (2003) Invasion percolation of single component, multiphase fluids with lattice Boltzmann models. Physica B 338:298-303

Sukop MC, Perfect E (2005) Solute Transport. In: Hillel D, Rosenzweig C, Powlson D, Scow K, Singer M, Sparks DL (eds) Encyclopedia of soils in the environment, Elsevier Ltd., Oxford, U.K., pp 521-531

Sutera SP, Skalak R (1993) The history of Poiseuille's Law. Annu Rev Fluid Mech 25:1-19

Swift MR, Orlandini E, Osborn WR, Yeomans JM (1996) Lattice Boltzmann simulations of liquid-gas and binary fluid systems. Phys Rev E 54:5041-5052

Taneda S (1956) Experimental investigation of the wake behind cylinders and plates at low Reynolds numbers. J Phys Soc Japan 11:302-307

Taneda S (1979) Visualization of separating Stokes flow. J Phys Soc Japan 46:1935-1942

Taylor G (1954) Conditions under which dispersion of a solute in a stream of solvent can be used to measure molecular diffusion. Proc R Soc London Ser A 225:473-477

Taylor G (1953) Dispersion of soluble matter in solvent flowing slowly through a tube. Proc R Soc London Ser A 219:186-203

Thorne DT, Sukop MC (2004) Lattice Boltzmann model for the Elder problem. In Miller CT, Farthing MW, Gray WG, Pinder GF (eds) Computational methods in water resources, Proceedings of the XVth international conference on computational methods in water resources, June 13-17 2004, Chapel Hill, NC, USA. Elsevier, Amsterdam, pp 1549-1557

Tölke J, Krafczyk M, Shultz M, Rank E (2002) Lattice Boltzmann simulations of binary fluid flow through porous media. Phil Trans R Soc Lond 360:535-545

Toride N, Leij FJ, van Genuchten M Th (1995) The CXTFIT code for estimating transport parameters from laboratory or field tracer experiments. Research Report No. 137, U.S. Salinity Laboratory, USDA-ARS, Riverside, CA.

Tritton DJ (1988) Physical Fluid Dynamics, 2nd Ed. Oxford University Press, Oxford New York

Truskett TM, Debenedetti PD, Sastry S, Torquato S (1999) A single-bond approach to orientation-dependent interactions and its implications for liquid water. J Chem Phys 111:2647-2656

Tuller M, Or D, Dudley LM (1999) Adsorption and capillary condensation in porous media: liquid retention and interfacial configurations in angular pores. Wat Resour Res 35:1949-1964

van Genuchten MTh, Wierenga PJ (1986) Solute dispersion coefficients and retardation factors. In Klute A (ed.) Methods of soil analysis, Part I, 2nd ed. Agron. Monogr. 9, Agronomy Society of America and Soil Science Society of America, Madison WI, pp 1025-1054

Vogel H –J, Tölke J, Schultz VP, Krafczyk M, Roth K (2005) Comparison of a lattice-Boltzmann model, a full-morphology model, and a pore network model for determining capillary pressure-saturation relationships. Vadose Zone J 4:380-388

Verberg R, Ladd AJC (2000) Lattice-Boltzmann model with sub-grid-scale boundary conditions. Phys Rev Lett 84:2148-2151

Wagner AJ (2003) The origin of spurious velocities in lattice Boltzmann. Int J Modern Phys B 17:193-196

Wolf-Gladrow DA (2000) Lattice-gas cellular automata and lattice Boltzmann models: an introduction. Springer-Verlag, Berlin Heidelberg New York

Wolfram S (2002) A new kind of science. Wolfram Media, Champaign, IL

Wolfram S (1986) Theory and Applications of Cellular Automata. World Scientific Publishing Co. Ltd., Singapore

Yang D, Udey N, Spanos TJT (1999) Thermodynamic automaton simulations of fluid flow and diffusion in porous media. Transport in Porous Media 35:37-47

Yang D, Udey N, Spanos TJT (1998) Automaton Simulations of Dispersion in Porous Media, Transport in Porous Media 32:187–198.

Yoshino M, Inamuro T (2003) Lattice Boltzmann simulations for flow and heat mass transfer problems in three-dimensional porous structure. Int J Numerical Meth Fluids 43:183-198

Yu D, Mei R, Luo L-S Shyy W (2003) Viscous flow computations with the method of lattice Boltzmann equation. Progress in Aerospace Sciences 39:329-367

Zhang J, Kwok DY (2004) Lattice Boltzmann study on the contact angle and contact line dynamics of liquid-vapor interfaces. Langmuir 20:8137-8141

Zhang R, Chen H (2003) Lattice Boltzmann method for simulations of liquid-vapor thermal flows. Phys Rev E 67:066711-1 - 066711-6

Zhang X, Ren L (2003) Lattice Boltzmann model for agrochemical transport in soils. J Contam Hydrol 67:27-42

Zhang X, Bengough AG, Crawford JW, Young IM (2002a) A lattice BGK model for advection and anisotropic dispersion equation. Adv Water Resour 25:1-8

Zhang X, Bengough AG, Deeks LK, Crawford JW, Young IM (2002b) A novel three-dimensional lattice Boltzmann model for solute transport in variably saturated porous media. Wat Resour Res 38: doi: 10.1029/2001WR000982

Zhang X, Crawford JW, Bengough AG, Young IM (2002c) On boundary conditions in the lattice Boltzmann model for advection and anisotropic dispersion equation. Adv Wat Resour 25:601-609

Zhou JG (2004) Lattice Boltzmann methods for shallow water flows. Springer-Verlag, Berlin Heidelberg New York

Zhou Y, Zhang R, Staroselsky I, Chen H (2004) Numerical simulation of laminar and turbulent buoyancy-driven flows using a lattice Boltzmann based algorithm. Int J Heat Mass Tran 47:4869-4879

Ziegler DP (1993) Boundary conditions for lattice Boltzmann simulations. J Stat Phys 71:1171-1177

Zou Q, He X (1997) On pressure and velocity boundary conditions for the lattice Boltzmann BGK model. Phys Fluids 9:1591-1598

Index