HOW THE LATTICE BOLTZMANN METHOD CHANGED MY LIFE FOREVER!

by

Matthew Grasinger

BS in Civil and Environmental Engineering, University of

Pittsburgh, Pittsburgh, PA 2013

Submitted to the Graduate Faculty of

the Swanson School of Engineering in partial fulfillment

of the requirements for the degree of

Master of Science

University of Pittsburgh

2016

UNIVERSITY OF PITTSBURGH

SWANSON SCHOOL OF ENGINEERING

This thesis was presented

by

Matthew Grasinger

It was defended on

July 20, 2016

and approved by

John C. Brigham, PhD, Senior Lecturer, School of Engineering and Computing Sciences at

Durham University

Julie M. Vandenbossche, PhD, Associate Professor, Department of Civil and

Environmental Engineering

Anthony T. Iannacchione, PhD, Associate Professor, Department of Civil and

Environmental Engineering

Thesis Advisor: John C. Brigham, PhD, Senior Lecturer, School of Engineering and

Computing Sciences at Durham University

HOW THE LATTICE BOLTZMANN METHOD CHANGED MY LIFE FOREVER!

Matthew Grasinger, M.S.

University of Pittsburgh, 2016

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1.0 NUMERICAL INVESTIGATION OF THE ACCURACY, STABILITY, AND EFFICIENCY OF LATTICE BOLTZMANN MODELS IN SIMULATING NON-NEWTONIAN FLOWS

1.1 ABSTRACT

The Lattice Boltzmann method (LBM) is a computational method that can be used for simulating fluid flow. It is well-suited for complex flows such as non-Newtonian, free surface, and multiphase multicomponent flows. Non-Newtonian flows are the primary focus of this paper, as many practical engineering problems such as the flow of cement slurries and concretes, the filling of molds by molten metals and plastics, blood flows, etc., are best modeled as non-Newtonian fluids. LBM is a statistical mechanical approach for simulating fluid flow. Each time step consists of streaming particle distributions to neighboring nodes, and collisions of particle distributions at each node through a collision operator. The collision operator is of interest because it has a lot of control over the physics that are simulated, e.g. constitutive laws, interfacial dynamics, etc., and it has implications on numerical stability and computational efficiency. In this paper, various collision operators and methods for stability enhancement were examined for their suitability for simulating non-Newtonian flows in terms of accuracy, numerical stability and computational efficiency. The investigation was be carried out, not as a rigorous mathematical treatment, but instead as a numerical study looking for qualitative, yet practical, results. The numerical investigation included testing the BGK and MRT collision operators, with and without entropic filtering, as applied to Bingham plastics and power-law fluid flows. Two different benchmark problems were chosen for the flows: Poiseuille flow, and lid-driven square cavity flow.

1

1.2 INTRODUCTION

1.2.1 Applications of Non-Newtonian Flow

The dynamic viscosity of a fluid is a measure of its resistance to shear deformation. For many fluids, at a constant temperature, the dynamic viscosity can be considered as constant. These fluids are known as Newtonian fluids. Non-Newtonian fluids have an apparent dynamic viscosity, or apparent resistance to shear stress, that is variable–even at a constant temperature–and is often a function of strain-rate. There are a number of fluids in science and engineering applications that can be classified as non-Newtonian; pastes, slurries, molten plastics, polymer solutions, dyes, varnishes, suspensions, and some biomedical liquids such as blood all behave in a non-Newtonian manner [7]. Of all of the different non-Newtonian behaviors that exist, there are two models under which much of the behaviors may be idealized:

1. *Yield stress fluids.* Yield stress fluids, also known as Bingham plastics, do not flow until a threshold value of stress, referred to as its yield stress, is exceeded. Some examples of yield stress fluids are drilling muds, fresh cement slurries and concretes, and melted metal alloys [22]. Understanding the flow of cement slurries and concretes is important because they are used in the cementing of oil and gas wellbores, and in construction. If the cement job for an oil and gas wellbore is not completed successfully then gases and fluids can leak through the wellbore to contaminate ground water or the atmosphere [24]. The goal of understanding cement slurry and concrete flows during construction is to understand how construction forms can be completely filled as voids can lead to structural deficiencies [**?**]. Yield stress flow is relevant in many other disciplines and applications because of the many substances that exhibit yield stress behavior, e.g. pastes, paints, muds, molten plastics and metals, and blood [48]. As a consequence, simulating yield stress fluids can help engineers develop better pastes and paints, understand the flow and deformation of mud and clay in geotechnical engineering, understand blood circulation, and better manufacture metal and plastic parts.

2. *Power-law fluids.* Power-law behavior is more commonly known as *shear-thinning*–when

the apparent viscosity decreases with increasing strain-rate–or *shear-thickening*–when the apparent viscosity increases with increasing strain-rate. Shear-thinning fluids are also known as *pseudoplastics*, and some examples include polymer mixes and molten plastics. Shear-thickening fluids are also known as *dilatants*, and some examples include quicksand, a cornstarch and water mixture, and a silica and polyethylene glycol mixture.

Developing numerical methods for approximating these non-Newtonian flows is important because of how difficult, and often impossible, it is to derive analytical, closed-form solutions.

1.2.2 Lattice Boltzmann Method for non-Newtonian Flow

Analytical solutions rarely exists for even the simplest non-Newtonian flows because of the complexity that a nonlinear constitutive relationship entails. It is generally more practical to approximate non-Newtonian flows using numerical methods. However, the nonlinear constitutive equation–typically of the form

$$\tau = \mu_{app}(\dot{\gamma})\dot{\gamma}, \tag{1.1}$$

where the apparent viscosity, μ_{app} is a function of the strain rate–results in certain challenges for numerical methods as well. Determining the apparent viscosity and shear rate of a flow will often require an iterative solution, a general Picard-type algorithm is:

1. Start with initial guess for the apparent viscosity, $\mu_k = \mu_0$.
2. Solve for the flow using the current value of apparent viscosity, $\dot{\gamma}_k = \tau_k/\mu_k$.
3. Update the apparent viscosity, $\mu_{k+1} = \mu(\dot{\gamma}_k)$.
4. Return to Step 2 until convergence is met.

Numerical solutions work by discretizing the equations that govern the physics of interest. When the solution for fluid flow problems vary in space and time, a numerical approximation requires breaking the problem up into discrete locations and discrete time steps. The significance of approximating a solution by discretizing the governing equations is that the iterative solution for the constitutive equation must be solved at each discrete location for each discrete time step, which can become computationally expensive. The lattice Boltzmann method (LBM) is a numerical method for fluid flow that has the advantage

that computing the strain rate is second-order accurate in space and local to each node [27]. This means that although an iterative solution is still required to determine the local strain rate and apparent viscosity, each iterative solution can be done in parallel, by a separate process, as they are independent of each other. Because hardware architectures have shifted from single, sequential processing systems to parallel processing systems, the local nature of the stress–strain-rate relationship in LBM gives it a distinct advantage for simulating non-Newtonian flows over some other numerical methods.

The lattice Boltzmann method has been studied and successively applied to modeling various non-Newtonian flows. Chai et al. [13], Chen et al. [15], Fallah et al. [20], Tang et al. [43], Vikhansky [46], Wang and Ho [47] developed LBM models for simulating yield stress flows. The LBM model results agreed well when compared to analytical solutions for Bingham plastic Poiseuille flow, and values from literature for lid-driven cavity flows–showing the feasibility of using LBM models for yield-stress flows. LBM models for power-law fluid flows [8, 13, 48, 49], and blood flows using the K-L, Casson, and Carreau-Yasuda constitutive relationships [1], have also been successfully developed and verified.

LBM does however, have its drawbacks. LBM can be considered as a type of finite-difference scheme for the continuous Boltzmann equation, and as such, has numerical properties in common with finite-difference schemes. One such consideration associated with this view, is the potential for numerical inaccuracies and instabilities [2, 28, 35, 36]. Stability concerns are just as prevalent, if not more prevalent, in simulating non-Newtonian flows because the nonlinear relationship between shear stress and strain-rate can lead to high nonequilibrium fluctuations. Various schemes and strategies for incorporating the physics of non-Newtonian flows and yet maintaining a stable numerical method have been developed and studied. The simplest approach for simulating a shear-rate dependent viscosity, used in [8, 15, 20, 39, 40, 43, 52], is to make the collision frequency–which is proportional to apparent viscosity–variable and dependent on the local strain rate. A potential issue with the stability of the variable relaxation time approach is that as the collision frequency, ω, approaches 2 the viscosity approaches zero and overrelaxation occurs; and alternatively, if the relaxation time is much greater than one, the accuracy and stability of the method also degrades [29]. In order to ensure that the variable collision frequencies did not approach val-

ues leading to numerical instabilities, [21, 39, 40] set upper and lower bounds on allowable collision frequencies. Although bounding the collision frequency was shown to be effective in terms of stability, it is also nonphysical and can lead to approximations that are inaccurate, not because of round-off error or numerical instability, but because the collision does not reflect the proper constitutive relationship of interest. Another scheme for incorporating non-Newtonian effects into LBM is to use a constant collision frequency, typically unity, and to instead incorporate the local shear-rate effect into equilibrium distribution functions. This means particle distribution functions will always relax toward equilibrium at the same rate, but that the definition of equilibrium is modified to represent the correct stress–strain-rate relationship. The equilibrium distribution function is derived for the specific constitutive relationship of interest using the Chapman-Enskog multiscale expansion. The equilibrium distribution function for Bingham plastic flow was derived, implemented, and verified in [47]; and the equilibrium distribution functions for power-law fluids and Carreau fluids were derived, implemented, and verified in [51]. The strategy of using an equilibrium distribution function that incorporates the local shear-rate effect has the advantage that, because the relaxation time is constant (at unity), the local relaxation time will not approach values that are too small or too large, and so there is no reason to bound the collision frequency in a way that is nonphysical. Wang and Ho [48] developed another constant-collision frequency LBM scheme for non-Newtonian flows by splitting the effects of constitutive relationship into Newtonian and non-Newtonian parts–for example, considering the plastic viscosity and yield stress for a Bingham plastic separately–then incorporating the Newtonian part in the usual way, namely scaling the collision frequency to achieve the macroscopic (Newtonian) viscosity, and incorporating the non-Newtonian part as a source of momentum, i.e. as an external forcing term, that is dependent on local shear-rate. Although the constant-collision frequency strategies present interesting alternatives, the variable collision scheme is used in the present study because of its simplicity and generality, i.e. it can be easily fitted to any constitutive equation without for example, performing Chapman-Enskog multiscale expansion.

Much work has been done (not specific to non-Newtonian flows) to investigate and develop ways to enhance the stability of lattice Boltzmann methods. Sterling and Chen [35, 36] investigated the stability of LBM with the linear Bhatnagar–Gross–Krook (BGK) collision

operator on 7-velocity and 9-velocity 2D lattices and a 15-velocity cubic lattice. The investigation was performed by linearizing all nonlinear terms about global equilibrium values and applying a von Neumann stability analysis on the resulting linearized equations. The spatial dependence of stability was analyzed by taking the Fourier transform of the linearized equations. Sterling and Chen concluded that the linear stability of LBM models depends on the mass distribution parameters, the mean velocity, the relaxation time, and the wave number. Worthing et al. [50] extended the stability analysis performed by Sterling and Chen and discovered both physical and nonphysical instabilities; and that setting the BGK collision frequency, $\omega = 1$ provides the optimal accuracy in time.

Although much work has been done to characterize the stability of LBM models, these analyses have the following limitations to their applicability:

- von Neumann stability analysis, and linear stability analysis in general, requires modifying the governing equations of the model so that they are linear. This linearization cannot be done without introducing some error into the analysis.

- Much of the analysis that exists in literature is done about a uniform flow [32, 50].

- Analyses are most often carried out assuming periodic boundary conditions. Periodic boundary conditions are advantageous for stability analysis because the analysis is generally done by a Fourier analysis on the propagation of error. The periodicity of Fourier modes allows the periodic boundary conditions to automatically be satisfied. Such an analysis, however, does not generalize well to instances when modes of the system are heavily influenced by the boundary conditions [50].

- The analyses are general and do not delve into any of the nuances of non-Newtonian flows.

To improve upon the stability of LBM models, d'Humieres [18] developed a multiple-relaxation-time (MRT) collision operator, that takes place in moment space and allows each moment to relax at a different rate. Lallemand and Luo [28] used von Neumann stability analysis to investigate the stability of the newly constructed LBM-MRT model and concluded LBM with the MRT collision operator was more stable, but with increased computational expense, than what is still often used, the BGK collision operator. Note that although this

increased computational expense was decided to be not too significant for Newtonian flows ($\approx$ 10-20% [28]), the issue may be magnified for non-Newtonian flows because an iterative solution for the constitutive equation can require that certain expensive computations be performed at each iteration (the strain rate tensor is determined from the nonequilibrium distribution function which must be mapped into momentum space when using the MRT collision operator). Both the BGK [1, 43, 48, 49] and MRT [13, 15, 20] collision operators have been used successfully to simulate non-Newtonian flows. Chen et al. concluded that the MRT collision operator was more stable for Bingham plastic flow and allowed the use of a more accurate approximation to the Bingham plastic constitutive relationship. However, what remains unclear is:

- What is the increased cost associated with the MRT collision operator when applied to non-Newtonian flows?

- Under what conditions, e.g. material parameters, physical problem, etc., for Bingham plastic fluid flows is the MRT collision operator necessary to maintain stability and/or accuracy?

- Under what conditions, e.g. material parameters, physical problem, etc, for power-law fluid flows is the MRT collision operator necessary to maintain stability and/or accuracy?

- What are additional strategies for increasing stability and accuracy, and what are their associated computational costs?

In regards to the last question, much work has been done recently to enhance stability of LBM models beyond the MRT collision operator. Brownlee et al. [9, 10, 11], Gorban and Packwood [23], Packwood [33] have all developed and tested means for introducing artificial dissipation in order to dampen out high frequency, nonphysical oscillations–oscillations that often lead to numerical instabilities if left unchecked. Stability enhancement through artificial dissipation and entropic filtering has shown a lot of promise, but to the authors' knowledge has not been tested for use in non-Newtonian flows.

The goal of this paper is to numerically study the implications of accuracy, stability, and efficiency for some of the different strategies for simulating non-Newtonian flows using LBM. The intention of the study is to aid scientists and engineers in understanding which

strategy is best suited to their priorities and applications of interest so as to maximize the advantages LBM has in simulating non-Newtonian flow. Advantages, such as LBM's potential to scale well in parallel, can be much less realized if the collision operator is too computationally expensive, or if numerical instabilities ensue. A numerical study can help to determine approximate numerical values, domains, and boundary conditions in which one LBM scheme may be more advantageous than another so that LBM may be used in a computationally efficient and stable manner.

1.3   NUMERICAL METHODS

1.3.1   Lattice Boltzmann Method: Overview

The Lattice Boltzmann method is a numerical approach that uses statistical mechanics to represent a variety of physical processes, such as fluid flow. More specifically, LBM can be thought of as a special finite difference discretization of the lattice Boltzmann equation [14]. The length scale of LBM is unique in contrast to most common numerical methods, and is referred to as the mesoscale. In contrast to continuum based methods, LBM simulates the kinetics of microscopic particles, and so it reaches a finer length scale than the macroscopic domain of continuum mechanics; and in contrast to molecular dynamics, discrete element method, and other particle scale approaches, LBM does not deal with a complete description of the degrees of freedom for each individual particle. LBM instead relies on a statistical description of particle distributions, making LBM, in general, more computationally efficient and requiring less memory than other particle methods. In addition, the Boltzmann equation is continuous, so LBM inherits the smoothness and stability of continuum methods. Thus, LBM can be seen as a compromise between continuum and particle methods, combining strengths from each.

LBM has some advantages over other methods of CFD. For example, LBM is a computationally efficient approach for some CFD. This efficiency is a consequence of two distinct features of LBM: (1) the convective operator is linear, as opposed to the nonlinear convection

terms that appear in continuum mechanics approaches; and (2) the fluid pressure is given by an equation of state. Solving for the fluid pressure in traditional method is more computationally expensive and requires special treatment such as iteration and/or relaxation [14].

1.3.2   The Boltzmann Equation

The Boltzmann equation (BE) can be thought of as a conservation of particle distributions. The BE is given as $\frac{\partial f}{\partial t} + \boldsymbol{\xi} \cdot \boldsymbol{\nabla} f = \boldsymbol{\Omega}$, where $f = f(\mathbf{x}, \boldsymbol{\xi}, t)$ is the particle velocity distribution function, $\mathbf{x}$ is the spatial position vector, $\boldsymbol{\xi}$ is the particle velocity, $\mathbf{u}$ is the macroscopic velocity, and $\boldsymbol{\Omega}$ is the collision operator. The lattice part of LBM refers to the way in which the BE is discretized. The lattice discretizes the spatial domain with nodes that are connected to their neighbors through discrete lattice velocity vectors. The velocity vectors act as pathways for particle distributions to travel along. Each time step in LBM consists of two distinct actions:

- Streaming: particle distribution propagate to their neighbors along the lattice velocity vectors. The particles can only move along the vectors in their specified direction and can only move at a specific speed.

- Collision: particle distributions meet at a node and "collide". In LBM, collisions are not simulated in a realistic sense, meaning that each individual particle does not exist and glance off of, or interact with, one another. Instead the collision operator is formulated in such a way that particle distributions are relaxed toward equilibrium. What defines equilibrium depends on the mechanics of interest to be modeled.

The D2Q9 lattice was used in the current work (shown in Figure 1.1), which is commonly used for two-dimensional, incompressible flow simulations [38]. The lattice is two-dimensional with nine discrete velocities at each node. There is a stationary particle, there are four discrete velocities of magnitude 1, $(\pm 1, 0)$, $(0, \pm 1)$, and there are four discrete velocities of magnitude $\sqrt{2}$, $(\pm 1, \pm 1)$.

The discretized version of the Boltzmann equation, or the lattice Boltzmann equation (LBE), is given as: $f_i(\mathbf{x} + \boldsymbol{\xi}_i, t + \Delta t) = f_i(\mathbf{x}, t) + \Omega_i(f)$, where, for D2Q9, $i = 0, 1, ..., 8$.
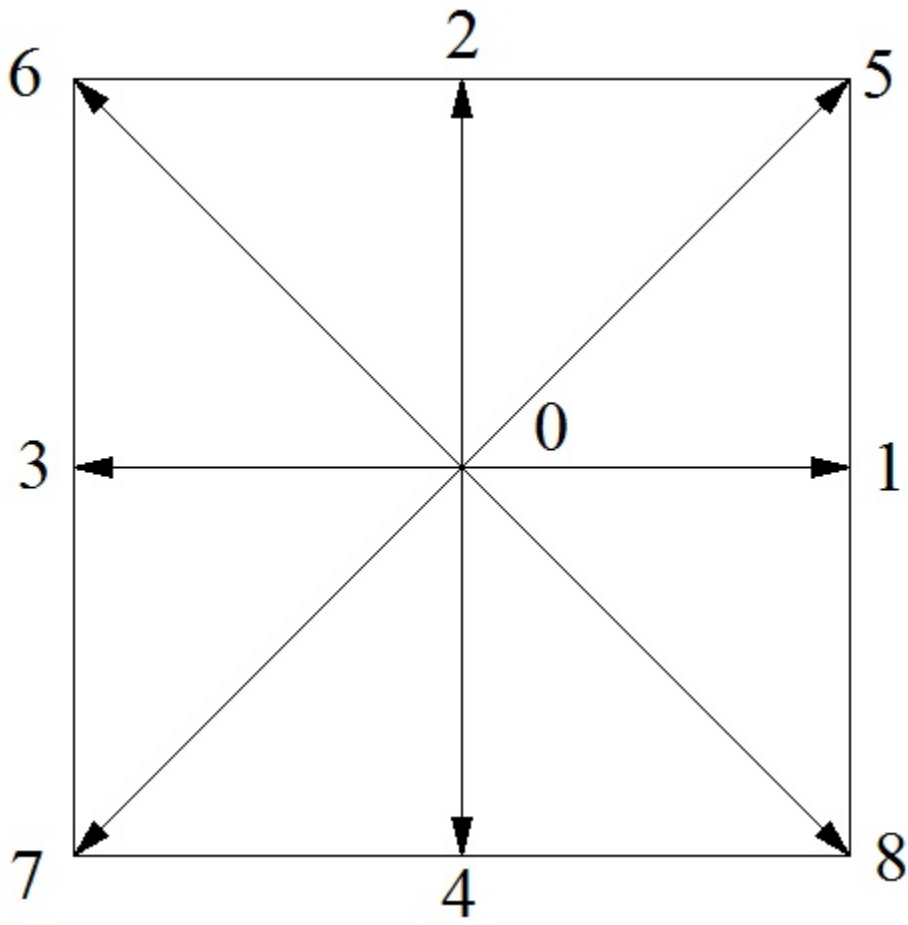
Figure 1.1: D2Q9 lattice

The macroscopic variables of interest can be calculated from the particle distribution functions, $f(\mathbf{x}, \boldsymbol{\xi}, t)$, by integrating moments of f over velocity space. Due to the discrete nature of velocity in LBM, the integrals simply become summations. The mass density is given by the sum of the particle distributions and the momentum density is given by the first moment of the particle distributions over the velocity space:

$$\rho(\mathbf{x}, t) = \sum_i f_i(\mathbf{x}, t), \tag{1.2}$$

$$\mathbf{j}(\mathbf{x}, t) = \rho(\mathbf{x}, t)\mathbf{u}(\mathbf{x}, t) = \sum_i \boldsymbol{\xi}_i(\mathbf{x}, t)f_i(\mathbf{x}, t) \tag{1.3}$$

The fluid pressure is related to macroscopic density through an equation of state:

$$p(\mathbf{x}, t) = \rho(\mathbf{x}, t)c_s^2, \tag{1.4}$$

where c_s is the lattice speed of sound ($c_s = \frac{1}{\sqrt{3}}$, for D2Q9).

1.3.3 Boundary Conditions

No slip, or zero velocity, which is commonly imposed at walls in a domain, is accomplished by simulating the particle distributions as "bouncing back" at the walls in the opposite direction from which they stream. For example, for a particle distribution streaming in the direction of a south wall, $f_2 = f_4$, $f_5 = f_7$, and $f_6 = f_8$. For velocity or pressure boundary conditions, the method proposed by Zou and He [53] can be used. The particle distributions that are missing after the streaming step are solved for by assuming a bounceback of the nonequilibrium distribution in the direction normal to the boundary; e.g., for a south inlet or outlet, $f_2 - f_2^{eq} = f_4 - f_4^{eq}$.

1.3.4 Applied Forces

Incorporating external forces, such as gravity, pressure gradients, etc., is done by adding a source of particle distributions in the direction of the force. The increase in particle

distributions leads to the desired macroscopic result–an increase in momentum. The LBE with external forces is:

$$f_i(\mathbf{x} + \boldsymbol{\xi}_i, t + \Delta t) = f_i(\mathbf{x}, t) + \Omega_i(f) + \frac{w_i \Delta t}{c_s^2}\mathbf{F} \cdot \boldsymbol{\xi}_i \qquad (1.5)$$

where $\mathbf{F}$ is the body force vector.

1.3.5 Collision Operator

1.3.5.1 Bhatnagar-Gross-Krook (BGK) The collision operator, in the case of the continuous BE, attempts to describe the change in particle momentums and trajectories due to pairwise particle collisions (based on their respective momentums and trajectories just prior to collision) [12]. In LBM, the collision operator causes particle distributions to relax toward a quasiequilibrium. This equilibrium is determined by the macroscopic physical behavior of interest. In the case of incompressible flow, the quasiequilibrium particle distribution, f_i^{eq}, is often given by:

$$f_i^{eq} = w_i \rho \left[1 + \frac{\boldsymbol{\xi}_i \cdot \mathbf{u}}{c_s^2} + \frac{(\boldsymbol{\xi}_i \cdot \mathbf{u})^2}{2c_s^2} - \frac{\mathbf{u}^2}{2c_s^2} \right], \qquad (1.6)$$

where w_i is the weight in the ith direction, and

$$w_i = \begin{cases} \frac{4}{9}, & i = 0 \\ \frac{1}{9}, & i = 1, 2, 3, 4 \\ \frac{1}{36}, & i = 5, 6, 7, 8 \end{cases} \qquad (1.7)$$

Due to its simplicity and computational efficiency, the most common collision operator is the Bhatnagar–Gross–Krook (BGK) operator. BGK consists of a single relaxation time and is a linear relaxation of particle distributions toward equilibrium. The BGK is expressed as:

$$\Omega_i = -\omega(f_i(\mathbf{x}, t) - f_i^{eq}), \qquad (1.8)$$

where ω is the collision frequency [4]. The collision frequency can be related to macroscopic constitutive properties through Chapman-Enskog multiscale analysis. For incompressible

Newtonian flow, the relaxation time is related to the kinematic viscosity by $\nu = c_s^2(\frac{1}{\omega} - \frac{1}{2})$. From this relationship it is clear that $\omega \in [0.0, 2.0]$, otherwise the viscosity would be negative.

Despite the utility of the BGK collision operator, it does have a few drawbacks. For example, in low viscosity flows the BGK operator results in an over relaxation of particle distributions toward quasiequilibrium. It is well known that when large nonequilibrium distributions exist in the LBM approximation that over relaxation can result in nonphysical oscillations that are slow to decay [10, 17]. To illustrate this, consider the example shown in Figure 1.2. If $a \gg 0$, then when $f_5(x, y)$ propagates to $f_5(x + 1, y + 1)$ during the streaming step (where $\mathbf{x} = [x, y]$, $\boldsymbol{\xi}_5 = [1, 1]$, and $\Delta t = 1$), $f_5(x + 1, y + 1)$ will be "far" from $f_5^{eq}(x + 1, y + 1)$ for the subsequent collision step. If $\nu \to 0$ then $\omega \to 2$ and (1.8) results in $f_5(x + 1, y + 1)$ still being "far" from $f_5^{eq}(x + 1, y + 1)$ but on the "other side". If, as in Figure 1.2, the LBM macroscopic variables dictate that f_i^{eq} remain mostly constant (or, at the very least, far from f_i) in the direction of propagation for f_i, then this leads to oscillations in the particle distribution, f_i. An example of the nonphysical oscillations that would occur for f_5 in Figure 1.2 can be seen in Figure 1.3. Considering the effect oscillations of particle distributions will have on macroscopic variables and, consequently, local quasiequilibriums, positive feedback loops can occur causing the system to "blow up" or "pollute" the system enough to make the results highly nonphysical and altogether useless [23].

In contrast to low viscosity flows, the challenge associated with high viscosity flows (using the BGK collision operator) is that particular distributions may never relax as "far" toward quasiequilibrium as is physical because as $\nu \to \infty$, $\omega \to 0$, resulting in extreme underrelaxation to the point of being negligible.

Concerns with sharp gradients, overrelaxation, and underrelaxation are particularly relevant in non-Newtonian flows because of the nonlinear constitutive relationship between shear stress and strain-rate. The nonlinear constitutive relationship can lead to sharp gradients, and depending on the form of $\dot{\gamma} \mapsto \mu_{app}$, the apparent viscosity, $\mu(\dot{\gamma}(\mathbf{x}))$, may result in overrelaxation in certain parts of the domain and extreme underrelaxation in others.

Due to the instabilities associated with the collision frequency being too high, e.g. approaching 2, or too low, e.g. approaching 0, a natural, albiet nonphysical, approach to using the BGK collision operator for non-Newtonian flows is to simply put bounds on the values in
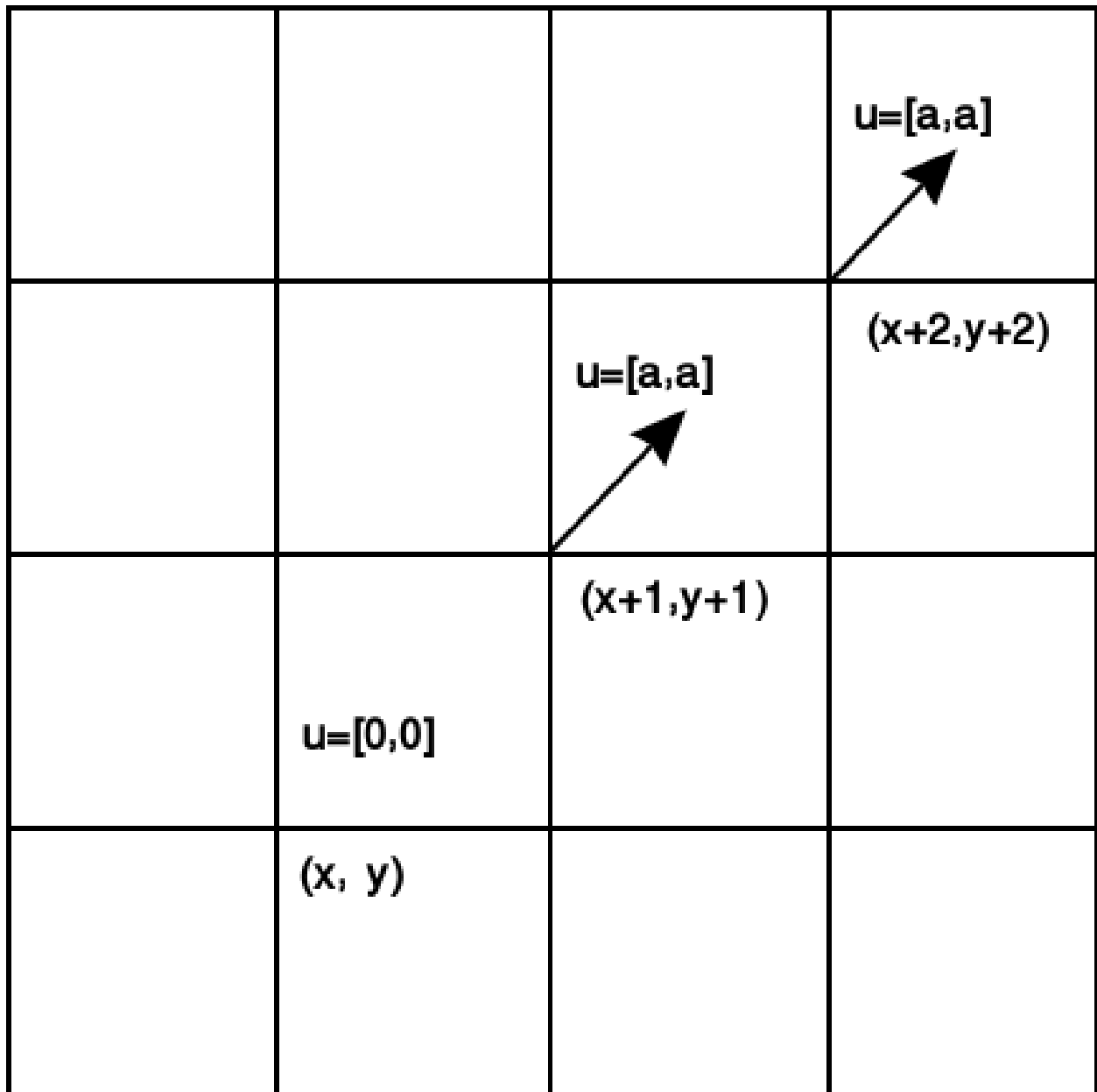
Figure 1.2: Example of an LBM approximation with a sharp gradient in the macroscopic flow velocity
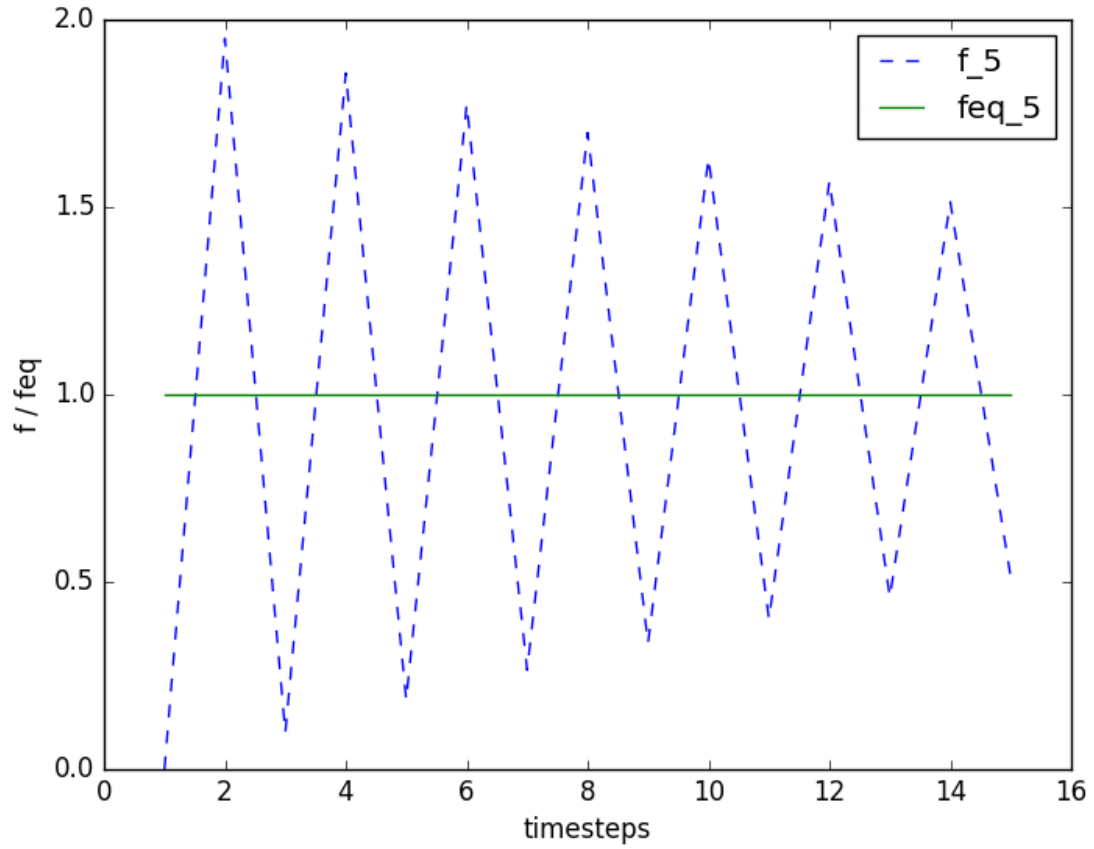
Figure 1.3: Example of oscillations that can occur as a result of overrelaxation (this example is related to the example shown in Figure 1.2); the relaxation time, $\omega = 1.95$ and $\nu \approx 0.0043$

which the collision frequency may attain. This simple methodology for increasing stability will henceforth be referred to bounded-relaxation time BGK, or BGK-BRT.

1.3.5.2 Multiple-relaxation Time (MRT) An alternative to the BGK collision operator is the multiple-relaxation-time (MRT) collision operator. In the LB-MRT scheme, one constructs a space based on the particle velocity, $\boldsymbol{\xi}$, moments of $\mathbf{f}$, herein referred to as the "moment space". The collision is then performed in the moment space. There a few reasons why it is advantageous to perform the collision in the moment space as opposed to the particle distribution space:

1. Physical processes in fluids can be approximately described by coupling or interacting among modes; and hydrodynamic modes are directly related to, i.e. linear combinations of, mass and momentum moments [28].

2. For the D2Q9 lattice, there are nine distribution functions, $f_0, f_1, ..., f_8$, but only six variables that affect the intended hydrodynamics on a macroscopic scale, namely: ρ, $\mathbf{u}$, and $\boldsymbol{\Pi}$, where $\boldsymbol{\Pi}$ is the momentum flux tensor [17]. Of the nine relaxation rates available, the three that correspond to the extra variables–often referred to as "ghost variables", and their associated modes as "ghost modes"–can be tuned in order to dampen out their associated ghost modes, ensuring these modes do not dominate or cause numerical instabilities at the lattice scale.

The MRT collision operator is given by:

$$\boldsymbol{\Omega} = -\mathbf{M}^{-1}\mathbf{S}\mathbf{M}(\mathbf{f} - \mathbf{f}^{eq}), \tag{1.9}$$

where $\mathbf{M}$ is a transformation matrix that maps the particle distribution vector, $\mathbf{f}$, and quasiequilibrium distribution vector, $\mathbf{f}^{eq}$, from the particle distribution space into moment space. The result of mapping the vectors $\mathbf{f}$ and $\mathbf{f}^{eq}$ into moment space will be denoted by $\mathbf{m}$

and $\mathbf{m}^{eq}$, respectively. The relationships between $\mathbf{m}$, $\mathbf{M}$ and $\mathbf{f}$ can be written as follows:

$$
\mathbf{m} = \begin{bmatrix} \rho \\ e \\ \epsilon \\ j_x \\ q_x \\ j_y \\ q_y \\ p_{xx} \\ p_{xy} \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ -4 & -1 & -1 & -1 & -1 & 2 & 2 & 2 & 2 \\ 4 & -2 & -2 & -2 & -2 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & -1 & 0 & 1 & -1 & -1 & 1 \\ 0 & -2 & 0 & 2 & 0 & 1 & -1 & -1 & 1 \\ 0 & 0 & 1 & 0 & -1 & 1 & 1 & -1 & -1 \\ 0 & 0 & -2 & 0 & 2 & 1 & 1 & -1 & -1 \\ 0 & 1 & -1 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & 1 & -1 \end{bmatrix} \begin{bmatrix} f_0 \\ f_1 \\ f_2 \\ f_3 \\ f_4 \\ f_5 \\ f_6 \\ f_7 \\ f_8 \end{bmatrix} = \mathbf{Mf}. \qquad (1.10)
$$

where ϵ is related to the square of the energy e; q_x and q_y correspond to the energy fluxes in the x and y directions; and p_{xx} and p_{xy} correspond to the diagonal and off-diagonal component of the viscous stress tensor [28]. The relaxation matrix, $\mathbf{S}$, is a diagonal matrix where each of the elements on the diagonal, $s_i \in [0, 2], i = 0, 1, ..., 8$, correspond to the relaxation rate of its associated hydrodynamic mode. In the case when $s_0 = s_1 = ... = s_8 = \omega$ (ω is the collision frequency in the BGK sense), the MRT collision operator is equivalent to the BGK collision operator. The relaxation parameters s_0, s_3, and s_5 are all set to zero as mass and momentum should be conserved. The relaxation parameters s_1 and $s_7 = s_8$ are related to the bulk and shear viscosities, respectively. The relationship for the shear viscosity is given by,

$$
\nu = c_s^2 \Delta t (\frac{1}{s_7} - \frac{1}{2}), \qquad (1.11)
$$

which is equivalent with the relationship to the collision frequency, ω, in the BGK sense (when $\Delta t = 1$). The remaining relaxation parameters, s_2, s_4, and s_6, are tuned in order to dampen out and separate the ghost modes from the modes affecting hydrodynamic transport. It is common practice, and Lallemand and Luo [28] recommends, that these three relaxation parameters be set to slightly larger than one.

The MRT collision operator has a greater numerical stability than its BGK counterpart [17, 19, 28], and because of the challenges associated with simulating non-Newtonian

17

flow, the MRT collision operator has become popular for simulating non-Newtonian flows [13, 15, 20, 24, 43, 46]. The main drawback of the MRT collision operator is its computational expense. Why MRT is more computationally expense is clear when one considers that (1.9) requires multiple matrix multiplications and (1.8) requires none. It has been reported that MRT is approximately 15% slower than BGK [19], but this was in the context of Newtonian flow. As will be shown later, for certain LBM implementations and non-Newtonian flows the increase in computational expense can be much greater.

1.3.6 Stability Enhancement through Artificial Dissipation: Entropic Filtering

To reduce nonequilibrium fluctuations in LBM, one can introduce artificial dissipation. The idea of artificial dissipation is to increase numerical stability, while sacrificing some physical accuracy. A model that has more physical justification but produces unstable and nonsensical results is much less useful than a model with some minor artificial features yet produces more stable results. A practical goal then would be to use only the necessary amount of artificial dissipation in order to ensure a stable solution. From this goal two questions naturally arise: "under what criteria does one decide that artificial dissipation necessary?" and "how much artificial dissipation does one introduce when it is necessary?".

To answer the first question, consider the discussion in Section 1.3.5.1 on problems with overrelaxation and nonphysical oscillations. Nonphysical oscillations due to overrelaxation would be damped out more quickly if particle distributions "far" from quasiequilibrium were brought closer to quasiequilibrium. A particle distribution vector, $\mathbf{f}$, "far" from equilibrium would therefor be a good candidate for artificial dissipation. There are many ways one can measure the distance between $\mathbf{f}$ and $\mathbf{f}^{eq}$; for example, a reasonable choice would be $||\mathbf{f} - \mathbf{f}^{eq}||_p$ for some p norm. A metric that has been developed and used successfully for determining when artificial dissipation should be introduced at a lattice site is the so called relative nonequilibrium entropy [9–11, 23, 33]. The relative nonequilibrium entropy, ΔS, is given by:

$$\Delta S = \sum_i f_i \ln(\frac{f_i}{f_i^{eq}}).$$
(1.12)

A more computationally efficient approximation of ΔS can be achieved by instead using the

second order Taylor expansion of (1.12),

$$\Delta S \approx \sum_i \frac{(f_i - f_i^{eq})^2}{2f_i^{eq}}. \tag{1.13}$$

Note that limiting nonequilibrium entropy in LBM is analogous to what flux limiters do in finite difference, finite volume, and finite element methods [11].

An option for defining criteria for when to introduce artificial dissipation that has been used successfully is to define a threshold, θ, such that dissipation is added when $\Delta S > \theta$ [9–11, 23]. A potential drawback of defining a threshold *a priori* is that in order to ensure the model still retains some physical integrity, only a small number of sites can have artificial dissipation added. If the threshold is too low, too many sites may have dissipation added. If the threshold is too high, a stable solution may not be achieved. The threshold can be determined on a case-by-case basis through trial-and-error or by a preliminary analysis. In the current work, the criteria that is used for determining whether dissipation should be added is,

$$\Delta S > \overline{\Delta S} + n_\sigma \cdot \sigma_{\Delta S}, \tag{1.14}$$

where $\overline{\Delta S}$ and $\sigma_{\Delta S}$ are the mean and standard of ΔS, respectively–both are calculated using values over the domain for the current timestep–and n_σ is the number of standard deviations greater than $\overline{\Delta S}$ that ΔS must be before dissipation is added. The number of standard deviations, n_σ, is chosen *a priori*. The criteria described in 1.14 has the advantage that one does not need to determine *a priori* what constitutes "far" from quasiequilibrium, but instead can think in terms of what would be the maximum percentage of sites one would want artificial dissipation to be added to.

Just as there are many ways to measure a lattice site's distance from quasiequilibrium, or from that measure define criteria for artificial dissipation, there are also many different ways of deciding how much dissipation to add. One method of adding dissipation is the so-called Ehrenfests' regularization [9], and involves setting a lattice site that is chosen for artificial dissipation to its quasiequilibrium state. Although this achieves the desired result, namely damping out large nonequilibrium fluctuations, it does so in an intense manner.

An alternative, the median filter, has been used successfully in conjunction with both the BGK and MRT collision schemes for simulating one-dimensional shock tubes and lid-driven cavity flows [11, 23, 33]. Median filtering is an effective noise reduction technique, often used in image processing, for "speckle noise" or "salt and pepper noise" [11]. In other words, median filters are good at reducing high frequency noise while having a minimal effect on lower frequency noise. In LBM this is a desirable way to introduce dissipation because it has the potential to reduce high frequency nonequilibrium fluctuations that might lead to numerical instability while retaining the lower frequency dynamics. To use the median filter one performs the collision step and then checks over the domain for lattice sites with ΔS that meet the criteria for artificial dissipation; sites that meet the criteria are updated as follows:

$$\mathbf{f} = \mathbf{f}^{eq} + \delta(\mathbf{f} - \mathbf{f}^{eq}),$$ (1.15)

where $\delta = \sqrt{\Delta S_{med}/\Delta S}$ is the scaling coefficient, and ΔS_{med} is the median value of ΔS for the nearest neighbors of the lattice site.

1.3.7 Strain-rate Tensor

In order to use the Picard-type algorithm outlined in Section 1.2.2, one must calculate the strain-rate, $\dot{\gamma}$. The strain-rate is given by the second invariant of the strain-rate tensor, $\mathbf{D}$, i.e.

$$\dot{\gamma} = \sqrt{\mathbf{D} : \mathbf{D}}.$$ (1.16)

When using the BGK collision scheme, the strain-rate tensor is determined by,

$$D_{\alpha\beta} = -\frac{\omega}{2\rho c_s^2} \sum_i \xi_{i\alpha}\xi_{i\beta}(f_i - f_i^{eq}),$$ (1.17)

and for the MRT collision scheme, the strain-rate tensor is determined by

$$D_{\alpha\beta} = -\frac{1}{2\rho c_s^2 \Delta t} \sum_i \xi_{i\alpha}\xi_{i\beta} \sum_j (\mathbf{M}^{-1}\mathbf{S}\mathbf{M})_{ij}(f_i - f_i^{eq}).$$ (1.18)

Computing the strain-rate tensor by either (1.17) or (1.18) is second order accurate in space [26, 27].

Upon comparison of (1.17) and (1.18), it is clear that calculating the strain-rate tensor for the MRT collision operator is more computationally expensive than it is for the BGK collision operator. The increase in computational expense is exacerbated by the fact that approximating the apparent viscosity, μ_{app}, by the algorithm outlined in Section 1.2.2 requires iteration and therefor multiple calculations of the strain-rate tensor.

1.3.8 Non-Newtonian Constitutive Equations

1.3.8.1 Bingham Plastic The Bingham plastic constitutive model is popular for yield stress flows. A Bingham plastic does not flow, i.e. the strain-rate is zero, when the shear stress is below the yield stress, and behaves in an almost Newtonian manner when the shear stress is above the yield stress. The Bingham plastic relationship is described mathematically as:

$$
\begin{cases}
\tau = \tau_y + \mu_p \dot{\gamma}, & |\tau| \geq \tau_y \\
\dot{\gamma} = 0, & |\tau| < \tau_y
\end{cases}
\tag{1.19}
$$

where τ is the shear stress, τ_y is the yield stress, and μ_p is the plastic viscosity [5].

Due to the discontinuous nature of (1.19), the Bingham plastic model is difficult to work with numerically. Thus, a smooth approximation to (1.19) formulated by Papanastasiou [34] is often used as:

$$
\tau = \tau_y(1 - e^{-m|\dot{\gamma}|}) + \mu_p \dot{\gamma},
\tag{1.20}
$$

where m is the stress growth exponent. The larger the value of m, the closer the approximation is to the Bingham plastic model.

Alternatively, the constitutive relationship can be interpreted through the apparent viscosity. The apparent viscosity is the value of the viscosity assuming a Newtonian relationship between stress and strain rate, $\mu_{app} = \frac{\tau}{\dot{\gamma}}$, and is used to determine the collision frequency for the BGK collision operator or s_7 and s_8 of the relaxation matrix for the MRT collision operator. Rearranging (1.20) results in the following expression for the apparent viscosity:

$$
\mu_{app} = \frac{\tau_y}{\dot{\gamma}}(1 - e^{-m|\dot{\gamma}|}) + \mu_p.
\tag{1.21}
$$

1.3.8.2   Power-law   The power-law constitutive relationship is useful for modeling fluids that experience shear-thinning or shear-thickening. The power-law relationship is given by,

$$\tau = k\dot{\gamma}^n, \tag{1.22}$$

where k is the flow consistency index and n is the flow behavior index. When $n = 1$, the (1.22) results in a Newtonian constitutive relationship with dynamic viscosity, $\mu = k$. A flow consistency index of $n < 1$ results in shear-thinning behavior, where as $n > 1$ results in shear-thickening. As with the Bingham plastic relationship, (1.22) can be modified to determine an apparent viscosity,

$$\mu_{app} = k\dot{\gamma}^{n-1}. \tag{1.23}$$

1.4   NUMERICAL STUDY

Some introduction to the numerical study. It has a purpose. Its purpose is to try to assess the accuracy, stability, and efficiency of the LBM models as applied to non-Newtonian flows. For all simulations presented in this section, the apparent viscosity is approximated using its corresponding constitutive relationship and the Picard-type algorithm outlined in Section 1.2.2 with the maximum number of iterations set at 15 and the convergence criteria,

$$\frac{\left|\mu_{app}^{k+1} - \mu_{app}^{k}\right|}{\mu_{app}^{k}} < 1.0 \times 10^{-6}. \tag{1.24}$$

All BGK-BRT collision schemes use $\omega \in [0.05, 1.995]$ as the bounds. All simulations with artificial dissipation use the median filter with $n_s = 2.7$. All numerical values given in this section are in lattice units unless otherwise stated.

1.4.1   Poiseuille Flow

Poiseuille flow is a useful benchmark because analytical solutions exist for both Bingham plastic and power-law fluids. Poiseuille flow is driven by a constant pressure gradient, $\frac{\partial p}{\partial x}$, through a two-dimensional channel. A schematic is shown in Figure 1.4. No-slip boundary

conditions are enforced at the top and bottom boundaries and periodic boundary conditions are enforced at the left and right boundaries. The total height of the channel will be denoted by H. The center of the channel is $y = 0$ and $y \in [-h, h]$ where $h = \frac{H}{2}$.

Unless otherwise stated, all of the Poiseuille flow simulations were computed on a 32×64 lattice for 25000 timesteps (although the length, L, seems small, the effect is negligible as the east-west boundary conditions are periodic). In reference to computational time, each of the simulations in this section were run on a single core of an Intel I7-860 Quad-Core 2.80GHz processor. For the MRT relaxation matrix the free parameters were set to $s_1 = s_2 = s_4 = s_6 = 1.1$.

1.4.1.1 Bingham Plastic Fluids The Bingham plastic simulations were carried out with a pressure gradient of $\frac{\partial p}{\partial x} = 1.0 \times 10^{-5}$, and a plastic viscosity of $\mu_p = 0.2$. The yield stress was varied between the four different values $\tau_y = [4.0, 8.0, 12.0, 16.0] \times 10^{-5}$, and four different LBM schemes were used: (1) BGK with $m = 10^5$, (2) BGK with $m = 10^8$, (3) BGK with $m = 10^8$ and the median filter, and (4) MRT with $m = 10^8$. (Recall m is the stress growth exponent for the Papanastasiou approximation.) The relative L_2 and relative L_∞ errors with respect to the analytical solution were computed for each simulation. The analytical solution for Bingham plastic flow is given by:

$$u_x(y) = \begin{cases} \frac{1}{2\mu_p} \left(-\frac{\partial p}{\partial x}\right) [h^2 - y_\tau^2] - \frac{\tau_y}{\mu_p} (h - y_\tau), & 0 \leq |y| \leq y_\tau, \\ \frac{1}{2\mu_p} \left(-\frac{\partial p}{\partial x}\right) [h^2 - y^2] - \frac{\tau_y}{\mu_p} (h - |y|), & y_\tau < |y| \leq h, \end{cases} \tag{1.25}$$

where $y_\tau = -\tau_y / \frac{\partial p}{\partial x}$ is the point at which the fluid yields.

The results are presented in Table 1.1. The Reynold's number was computed by $Re = \frac{\rho U H}{\mu_p}$, where U is the maximum velocity given by the analytical solution and $H = 2h$ is the total height of the channel. The Bingham number was computed by $Bn = \frac{\tau_y H}{\mu_p U}$.

As has been reported previously, for the BGK collision operator–regardless of whether or not median entropic filtering is used–using a stress growth exponent of $m = 10^5$ is more accurate with respect to the analytical solution than using a stress growth exponent of $m = 10^8$ [15]. A larger stress growth exponent results in a more accurate Papastasiou approximation of the true Bingham plastic constitutive model, however, it leads to more

23
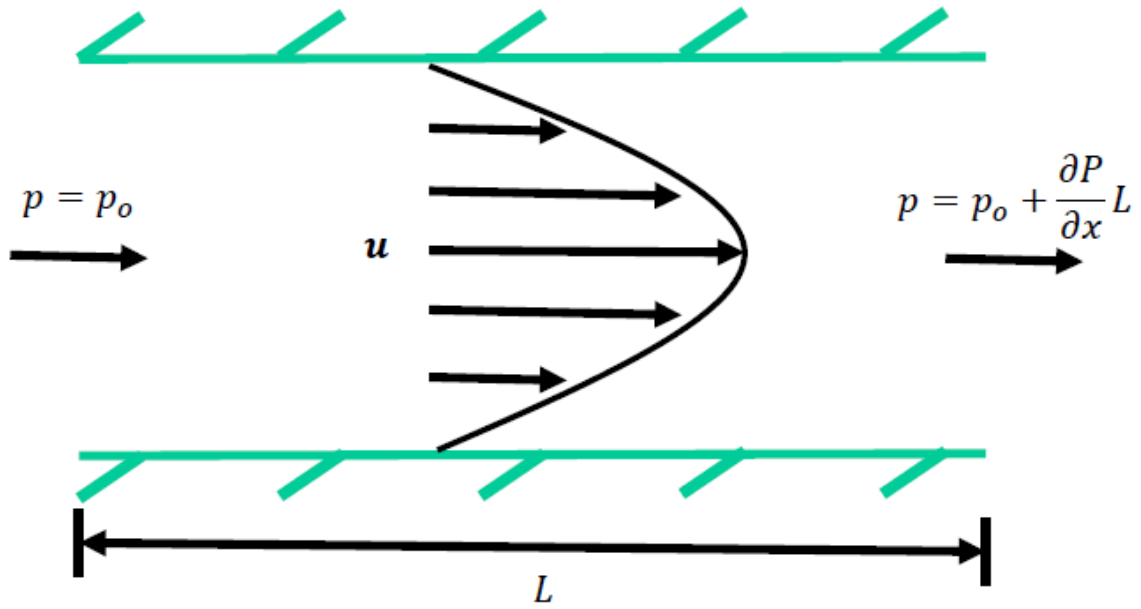
Figure 1.4: Schematic of Poiseuille flow; no-slip boundary conditions are enforced at the top and bottom boundaries, and periodic boundary conditions are enforced at the left and right boundaries.

Table 1.1: Bingham plastic Poiseuille flow

Collision Operator	Median Filter	m	τ_y $(\times 10^{-5})$	Re	Bn	L_2	L_∞	Time (sec)
BGK	No	10^5	4.0	6.05	0.68	0.0062	0.0153	1857
			8.0	4.42	1.85	0.0204	0.0411	2329
			12.0	3.04	4.04	0.0503	0.0891	3345
			16.0	1.92	8.52	0.1161	0.1879	2029
BGK	No	10^8	4.0	6.05	0.68	0.0109	0.0282	2831
			8.0	4.42	1.85	0.0330	0.0670	3509
			12.0	3.04	4.04	0.0788	0.1570	4838
			16.0	1.92	8.52	0.1991	0.3539	3790
BGK	Yes	10^8	4.0	6.05	0.68	0.0100	0.0273	2903
			8.0	4.42	1.85	0.0361	0.0823	3567
			12.0	3.04	4.04	0.2439	0.3832	4800
			16.0	1.92	8.52	0.7533	1.0718	3507
MRT	No	10^8	4.0	6.05	0.68	0.0013	0.0013	5914
			8.0	4.42	1.85	0.0018	0.0018	7908
			12.0	3.04	4.04	0.0026	0.0026	7160
			16.0	1.92	8.52	0.0041	0.0041	5559

noise that degrades the numerical solution for the BGK collision operator. Upon inspection of Table 1.1, it does not appear that entropic median filtering helps dissipate the noise that occurs as a result of using $m = 10^8$ for Bingham plastic Poiseuille flow. In fact, the median filter resulted in a less accurate solution in nearly all cases and rendered the solution altogether useless for the higher yield stress flows (relative errors of 25-75%).

The BGK collision operator using a stress growth exponent of $m = 10^5$ was in general the fastest. This can be attributed to the fact that a smoother approximation of the Bingham plastic constitutive model would lead to a solution for μ_{app} converging with less iterations. The BGK collision operator using a stress growth exponent of $m = 10^5$ also performed acceptably in terms of accuracy for lower yield stress flows, however, for $\tau_y = 12 \times 10^{-5}$ and 16×10^{-5} the relative L_∞ errors were 8.9% and 19%, respectively–larger than what would be considered acceptable for most engineering applications.

The LBM approximations are plotted with the analytical solution for the BGK collision operator using $m = 10^5$ and $m = 10^8$ in Figure 1.5 and Figure 1.6, respectively. Figure 1.5 shows that the error for the BGK model with $m = 10^5$ is not due to noise, but instead the inaccuracy of the Papastasiou approximation with a lower stress growth exponent. Figure 1.6 shows that the error for the BGK model with $m = 10^8$ is due to numerical noise. In order to better understand why median filtering did not eliminate the noise in high yield stress flows, but instead exacerbate the problem, it is necessary to investigate what is happening at the particle distribution scale.

Figure 1.7 compares particle distributions to their respective quasiequilbiriums. It can be seen that in the BGK, $m = 10^8$ solution, the noise due to nonlinear constitutive relation pollutes the quasiequilibrium values. Because the noise makes its way into the quasiequilibrium values, entropic median filtering does not help dampen the noise but instead contracts the particle distributions closer to the quasiequilibrium noise–explaining why the median filtered results were less accurate for Bingham plastic Poiseuille flow. In order to ensure that this phenomena was a side effect of all entropic filtering and not just entropic median filtering with $n_s = 2.7$, entropic filtering was tested with median filtering and Ehrenfests' regularization for $n_s = 2.7$ and for ΔS thresholds of $\delta = 10^{-4}$ and $\delta = 10^{-5}$. All filtering schemes suffered from approximately the same level of noise.
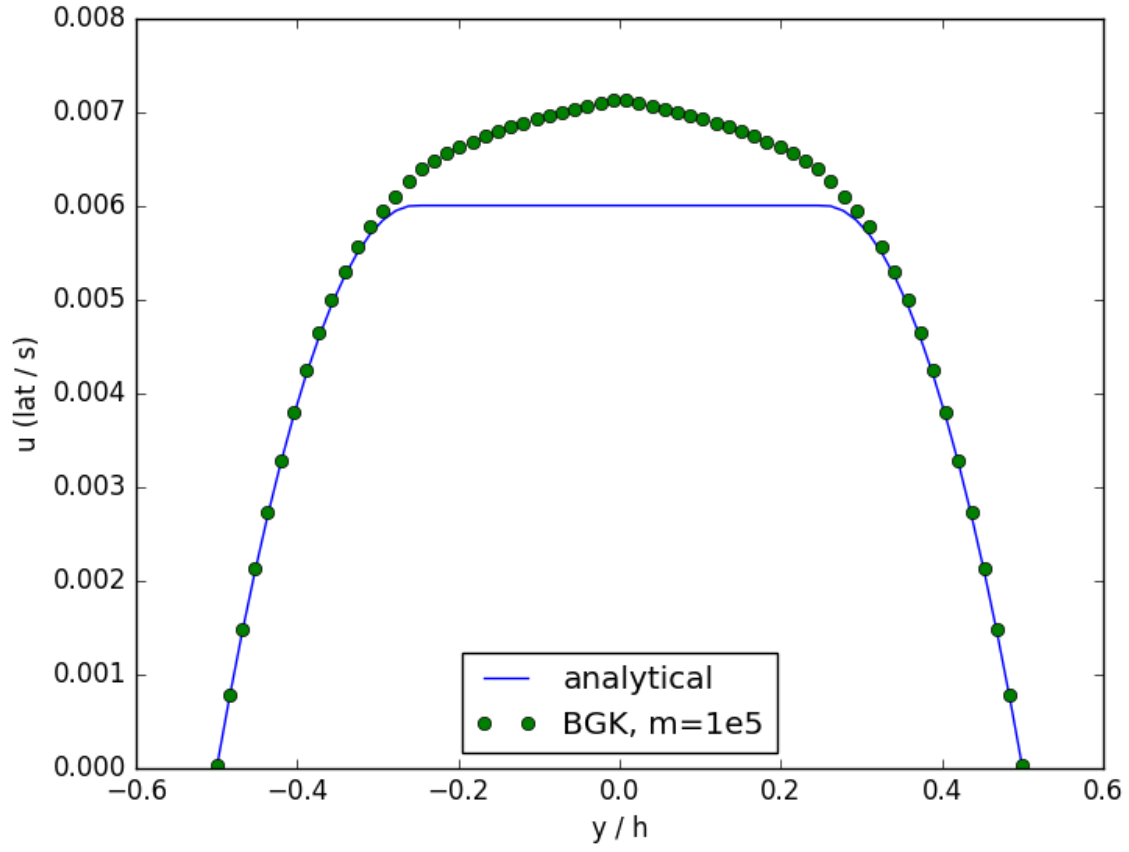
Figure 1.5: LBM approximation using BGK and $m = 10^5$ compared to the analytical solution for $\tau_y = 16 \times 10^{-5}$.
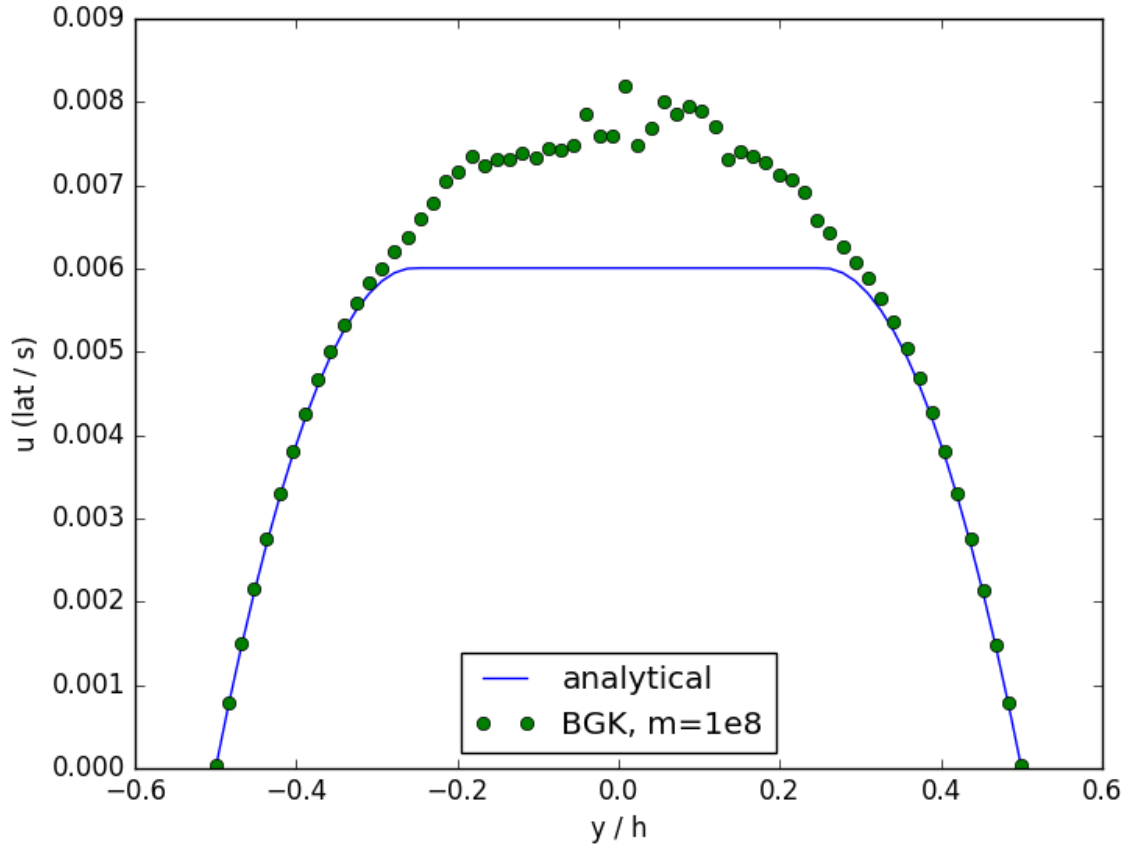
Figure 1.6: LBM approximation using BGK and $m = 10^8$ compared to the analytical solution for $\tau_y = 16 \times 10^{-5}$.
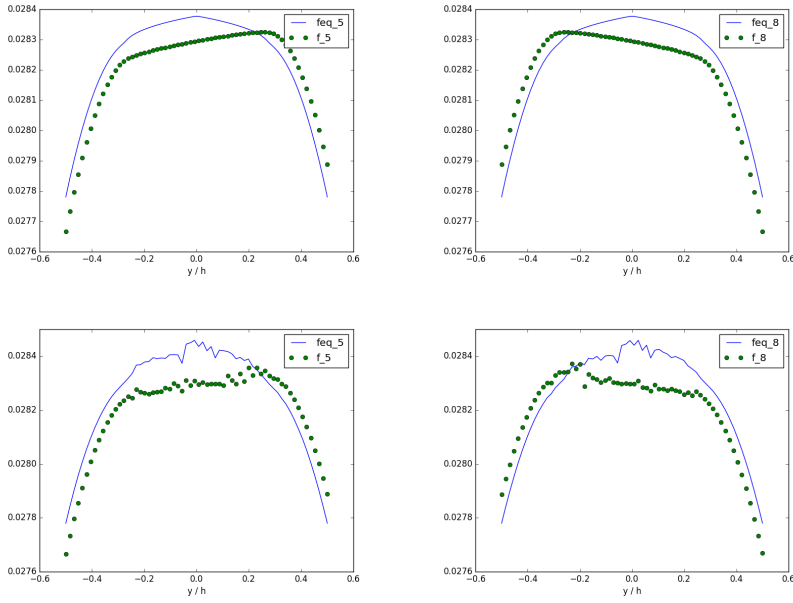
Figure 1.7: Particle distributions in the 5 direction (left) and 8 direction (right) compared to their respective quasiequilibriums. The top two plots are for the BGK with $m = 10^5$, the bottom two are for the BGK with $m = 10^8$.
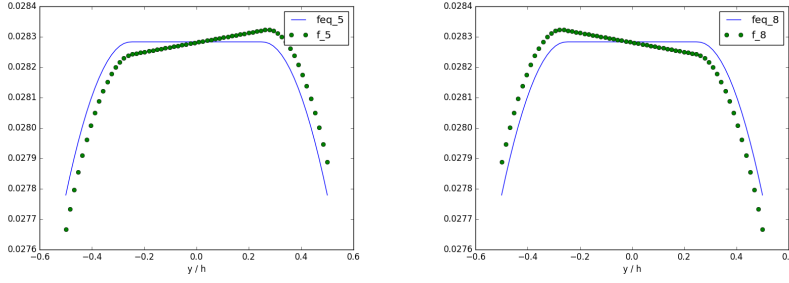
Figure 1.8: Particle distributions in the 5 direction (left) and 8 direction (right) compared to their respective quasiequilibriums for the MRT with $m = 10^8$.

On average, the LBM model using the MRT collision operator was 2.9 times slower than the fastest model for the given material parameters. However, the increased computational time can be justified for the MRT collision operator because it did not suffer from the same level of noise as the BGK collision operator when using the more accurate approximation of the Bingham plastic model, i.e. when $m = 10^8$. The MRT collision operator was therefor the most accurate solution for all cases. Figure 1.8 compares particle distributions to quasiequilibrium for the MRT model; in contrast to the BGK with $m = 10^8$, there is no noise in the quasiequilibrium distributions.

A natural question would be to ask "if the strength of the MRT collision operator is damping out the ghost modes, then is the noise for the BGK with $m = 10^8$ (with and without median filtering) indeed due to the ghost modes?". Figure 1.9 shows a measure of the nonequilibrium ϵ moment with respect to time for each of the collision schemes and was calculated using:

$$\frac{||\boldsymbol{\epsilon} - \boldsymbol{\epsilon}^{eq}||_2}{||\boldsymbol{\epsilon}^{eq}||_2} \tag{1.26}$$

where $\epsilon_j = 4f_0(\mathbf{x}_j) - 2f_1(\mathbf{x}_j) - 2f_2(\mathbf{x}_j) - 2f_3(\mathbf{x}_j) - 2f_4(\mathbf{x}_j) + f_5(\mathbf{x}_j) + f_6(\mathbf{x}_j) + f_7(\mathbf{x}_j) + f_8(\mathbf{x}_j)$ is related to the square of the energy, and $\mathbf{x}_j = (16, j)$ for $j = 1, 2, ..., 64$, i.e. $\mathbf{x}_j$ is taken at 16 nodes in from the left in the x-direction and for the full height of the channel in the y-direction. The values measured for the BGK with $m = 10^8$, with and without median filtering, were significantly higher than for either the MRT or BGK with $m = 10^5$ collision

schemes–which suggests that the numerical noise the BGK with $m = 10^8$ suffered from was in part due to the ϵ moment.

Figure 1.10 shows a measure of the nonequilibrium q_x moment, or energy flux in the x-direction, with respect to time for each of the collision schemes and was calculated using:

$$\frac{||\mathbf{q_x} - \mathbf{q_x}^{eq}||_2}{||\mathbf{q_x}^{eq}||_2} \tag{1.27}$$

where $q_{x_j} = -2f_1(\mathbf{x}_j) + 2f_3(\mathbf{x}_j) + f_5(\mathbf{x}_j) - f_6(\mathbf{x}_j) - f_7(\mathbf{x}_j) + f_8(\mathbf{x}_j)$ and $\mathbf{x}_j$ were taken across the height of the channel in the same manner as with the ϵ moment. The values measured for the BGK with $m = 10^8$ and median filtered were consistently greater than any other collision scheme. The q_x moment was likely the cause of the significant error and noise in the BGK with $m = 10^8$ and median filtering solution. As expected, the q_x^{neq} relative L_2 for the BGK with $m = 10^8$ was slightly greater than the BGK with $m = 10^5$, and the q_x^{neq} relative L_2 was negligible for the MRT collision scheme. These results are consistent with the solution noise and error.

It is likely that the nonequilibrium ϵ and q_x moment cause oscillations at the lattice level that have a cumulative effect in time. Figure 1.11 and Figure 1.12 show the cumulative measures of the nonequilibrium moments in time. Figure 1.12 suggests that the q_x^{neq} dominates and is the primary source of oscillations. If an LBM collision scheme is to be developed for simulating high yield stress flows that is more accurate than the BGK with $m = 10^5$ and more computationally efficient than the MRT, it should focus on a means of dampening the nonequilibrium energy flux moments, namely q_x and q_y.

1.4.1.2 Power-law Fluids The power-law simulations were carried out with a pressure gradient of $\frac{\partial p}{\partial x} = 1.0 \times 10^{-5}$, and a flow consistency index of $k = 0.2$. The flow behavior index was varied between the four different values $n = [0.5, 0.75, 1.25, 1.50]$, and four different LBM schemes were used: (1) BGK, (2) BGK with median filtering, (3) MRT, and (4) BGK-BRT, with ω bounded within $[0.05, 1.995]$. The relative L_2 and relative L_∞ errors with respect to the analytical solution were computed for each simulation. The analytical solution for power-law fluid flow is given by:

$$u_x(y) = \frac{n}{n+1}\left(-\frac{1}{k}\frac{\partial p}{\partial x}\right)^{1/n}\left[h^{\frac{n+1}{n}} - |y|^{\frac{n+1}{n}}\right]. \tag{1.28}$$
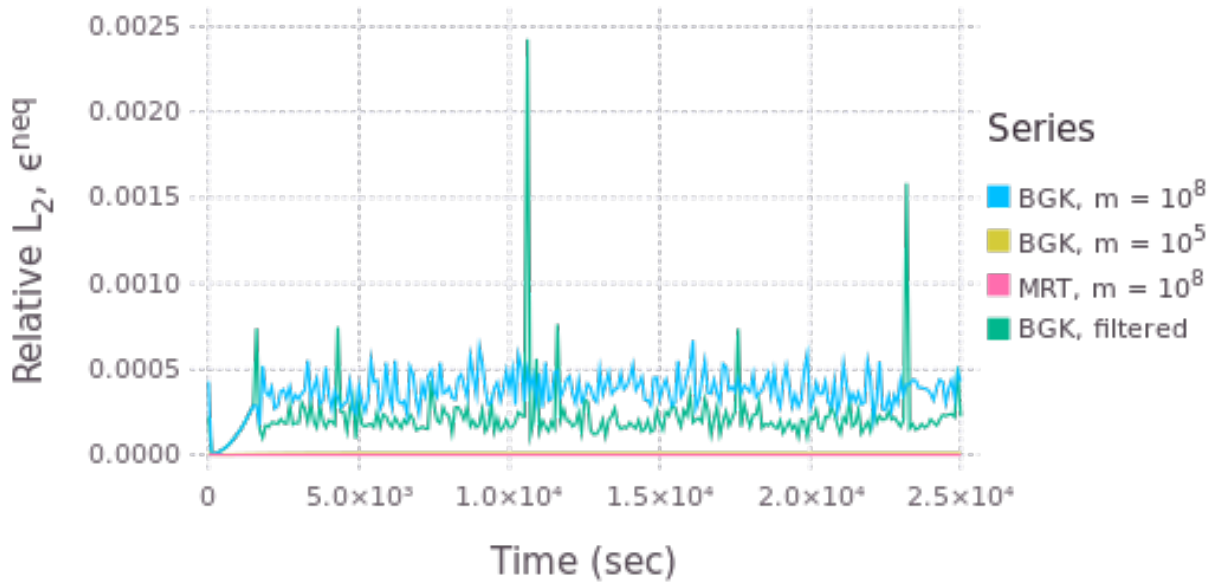
31

Figure 1.9: Evoluation of relative L_2, ϵ^{neq} with time. The norm of the ϵ^{neq} across the height of the channel is an indicator of an increase in oscillations at the lattice level due to the ϵ moment.
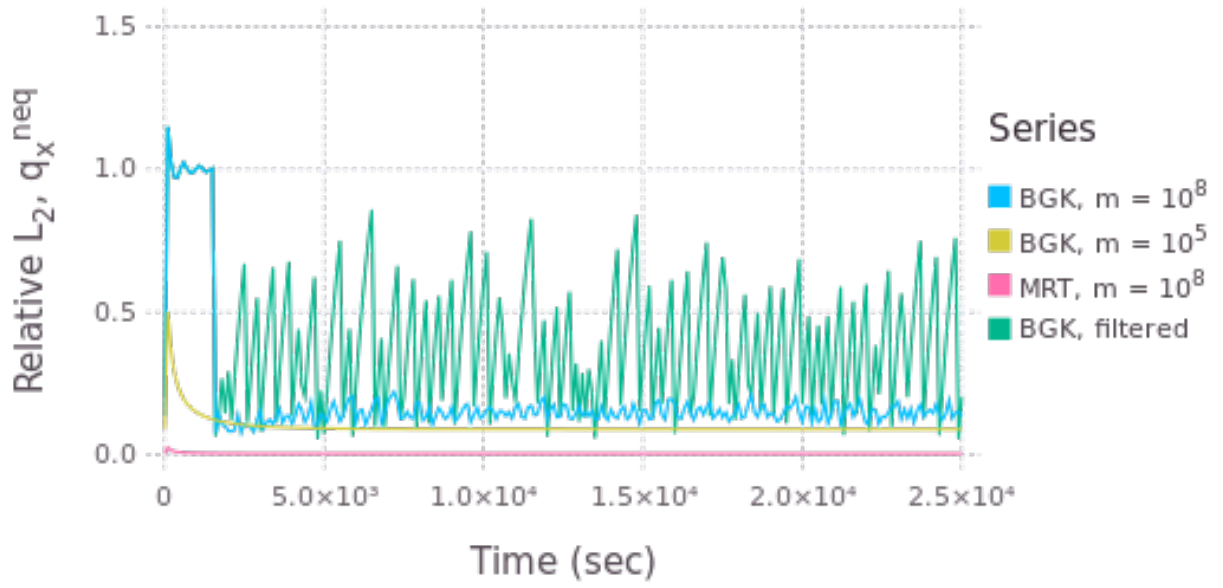
Figure 1.10: Evoluation of relative L_2, q_x^{neq} with time. The norm of the q_x^{neq} across the height of the channel is an indicator of an increase in oscillations at the lattice level due to the q_x moment.
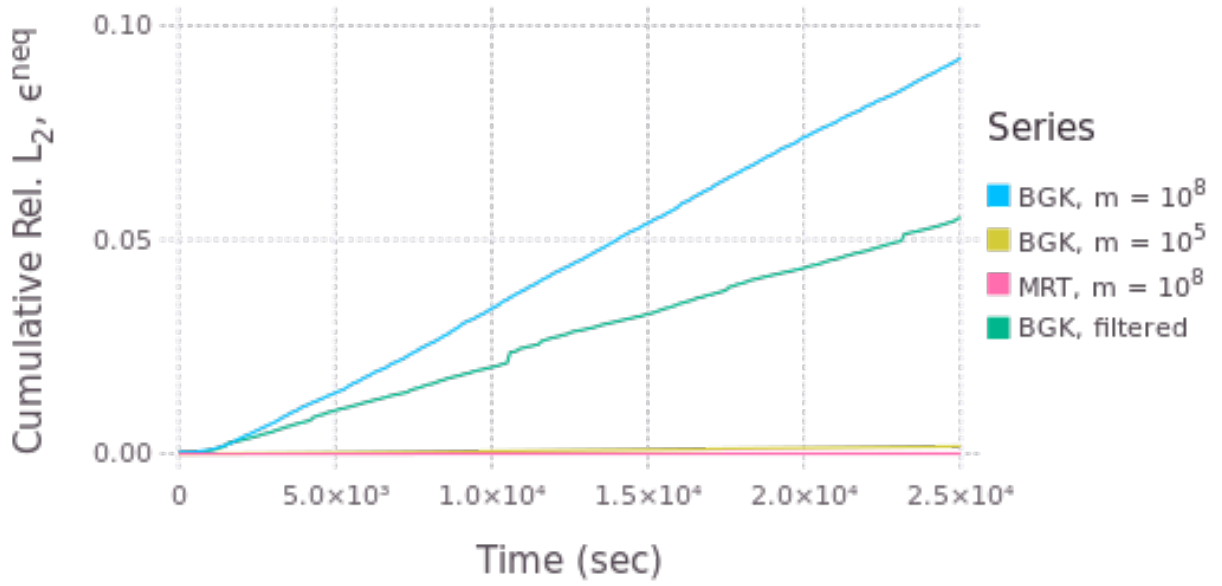
Figure 1.11: Cumulative relative L_2, ϵ^{neq} with time. Oscillations can have a positive effect on each other, building up. The cumulative relative L_2, ϵ^{neq} is a measure of how much oscillations due to the ϵ moment may have been building in time.
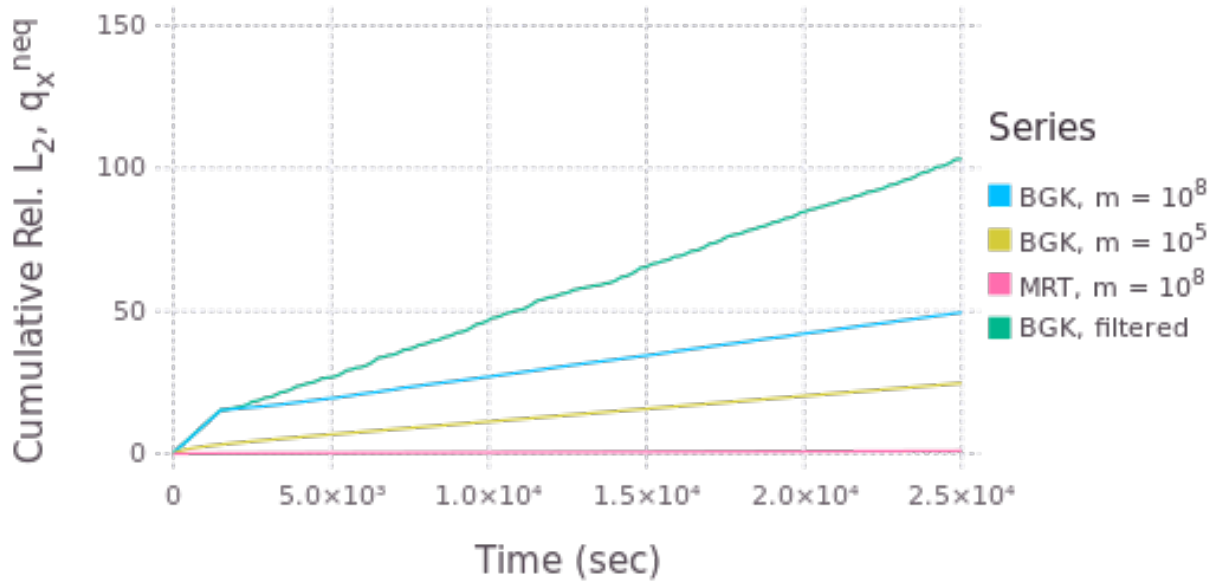
Figure 1.12: Cumulative relative L_2, q_x^{neq} with time. Oscillations can have a positive effect on each other, building up. The cumulative relative L_2, q_x^{neq} is a measure of how much oscillations due to the q_x moment may have been building in time.

The results are presented in Table 1.2. In reference to computational time, each of the simulations in this section were run on a single core of an Intel I7-860 Quad-Core 2.80GHz processor. The Reynold's number was computed by $Re = \frac{\rho U^{2-n} H^n}{k}$, where U is the maximum velocity given by the analytical solution and $H = 2h$ is the total height of the channel.

There are some interesting things to note from the power-law Poiseuille flow results. Although for $n = 0.5, 0.75, 1.25$ there is still the same efficiency for accuracy trade off when it comes to deciding between the BGK and MRT collision operators, for $n = 1.5$ the BGK operator with median filtering was the fastest, most accurate, and only solution that remained stable. In fact, besides for the case of $n = 0.5$, i.e. rather extreme shear-thinning, the BGK with median filtering was both fast and accurate. The BGK-BRT had a moderate improvement of accuracy and efficiency–with respect to the regular BGK collision scheme–for flows where the constitutive relationship was only somewhat nonlinear, i.e. for $n = 0.75, 1.25$. Note that none of the collision schemes were sufficiently accurate for the case of $n = 0.5$, and to improve this accuracy it would be necessary for a modeler to use a different simulation Mach number, a finer grid, and/or a smaller pressure gradient.

1.4.2 Lid-driven Cavity Flow

Lid-driven cavity flow is another good benchmark for two-dimensional fluid flow because it is sufficiently simple in terms of boundary conditions and implementation, while also complex in the flows and vortices it produces. The lid-driven benchmark was chosen for this numerical study because there are many results available in literature in which to compare with and because the vorticity of the flow coupled with the nonlinear constitutive equations should result in a challenge in terms of stability. Lid-driven cavity flow is generally characterized by a square cavity where a fluid velocity is prescribed tangential to the upper boundary and the remaining boundaries have a no-slip boundary condition. Lid-driven cavity flow is depicted in Figure 1.13.

The lid-driven cavity simulations presented in the section were all simulated on a coarse, 100×100 lattice; a coarse lattice was chosen in order to highlight concerns with stability and accuracy. The simulations were run for either 50000 timesteps, or until convergence was

Table 1.2: Power-law Poiseuille flow

Collision Operator	Median Filter	n	Re	L_2	L_∞	Time (sec)
BGK	No	0.50	0.0007	28.06	30.37	5232
		0.75	0.9125	0.0328	0.0569	2399
		1.25	423.2	0.0051	0.0055	2311
		1.50	2213	1.0	1.0	3622
BGK	Yes	0.50	0.0007	26.72	28.78	5347
		0.75	0.9125	0.0328	0.0569	2385
		1.25	423.2	0.0051	0.0055	2275
		1.50	2213	0.0570	0.0600	2357
MRT	No	0.50	0.0007	0.1758	0.1401	4906
		0.75	0.9125	0.0058	0.0045	4880
		1.25	423.2	0.0051	0.0055	4863
		1.50	2213	1.0	1.0	15992
BGK-BRT	No	0.50	0.0007	11.73	12.56	4450
		0.75	0.9125	0.0320	0.0475	1805
		1.25	423.2	0.0051	0.0055	1905
		1.50	2213	1.0	1.0	8194
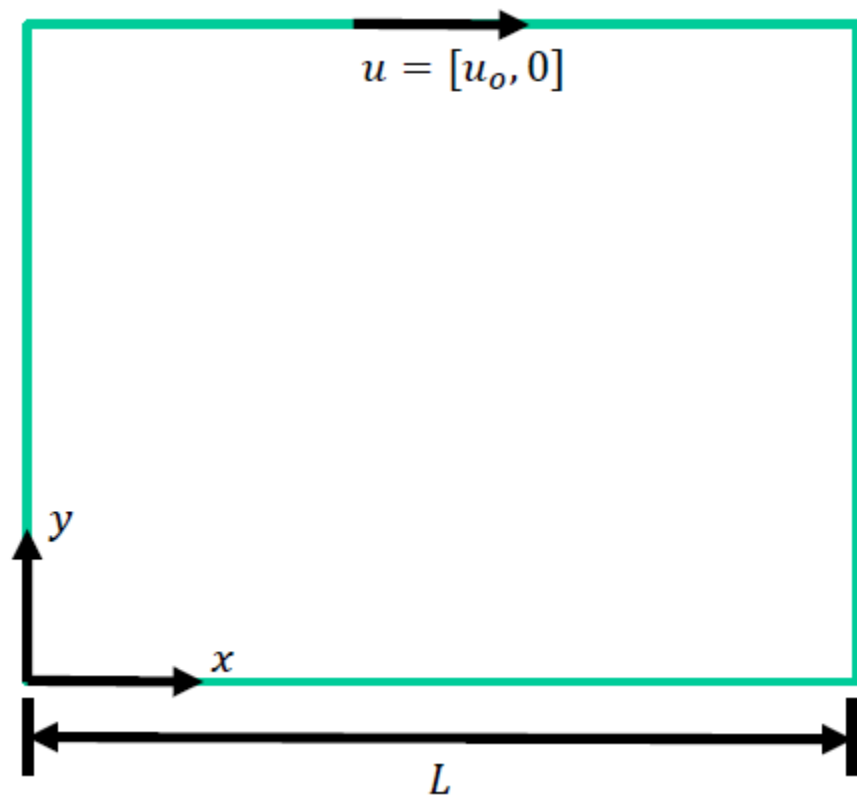
Figure 1.13: Schematic of lid-driven cavity flow; a velocity is prescribed tangential to the top boundary and no-slip is enforced at the remaining boundaries.

met. Convergence was defined by,

$$\sum_{m=1}^{100} \sum_{i,j} \frac{|u_{i,j}^k - u_{i,j}^{k-m}|}{|u_{i,j}^{k-m}|} < 1.0 \times 10^{-7}, \tag{1.29}$$

where i is the node index in the x-direction, j is the node index in the y-direction, and k is the current timestep. In reference to computational time, each of the simulations in this section were run on a single core of either an Intel I7-860 Quad-Core 2.80GHz processor, or a SOME OTHER PROCESSOR I NEED TO LOOK UP. The lid velocity was prescribed at $u_o = 0.1$. For the MRT relaxation matrix, the free parameters were set to $s_1 = 1.1$, $s_2 = 1.0$, and $s_4 = s_6 = 1.2$. All coordinate values presented in this section (used to describe the location of the center of vortices) are given normalized with respect to the length of the cavity side, i.e. as $(x/L, y/L)$.

1.4.2.1 Bingham Plastic Fluids For the Bingham plastic numerical tests the Reynold's number was varied, $Re = 100, 1000, 5000, 1000$, and the Bingham number was varied, $Bn = 1, 10, 100$ (Bn and Re were calculated the same as in Section 1.4.1.1). The collision schemes that were tested were: (1) BGK with $m = 10^5$, (2) BGK with $m = 10^8$, (3) BGK-BRT with $m = 10^8$, (4) BGK with $m = 10^8$ and median filtering, (5) MRT with $m = 10^8$ and MRT with $m = 10^8$ and median filtering. Tables 1.3–1.5 compare the center location of the main vortex to literature values taken from Syrakos et al. [41]. The main vortex location is determined by calculating the stream function using Simpson's rule and finding where it attains a maximum.

Just as before with the Bingham plastic Poiseuille flow, if a modeler is using the BGK collision operator and entropic filtering is not available, a stress growth exponent of $m = 10^5$ yields faster, more accurate results (assuming the literature values are a good measure of accuracy) than using a larger stress growth exponent. A smaller stress growth exponent is also more effective at producing accurate results in the BGK collision scheme than placing bounds on the relaxation time (BGK-BRT). However, although the BGK with $m = 10^5$ was the fastest model in all cases, it was also–like the other two BGK models without median filtering–unstable for $Re \geq 5000$.

Table 1.3: Bingham plastic, lid-driven cavity flow; $Bn = 1$.

Bn	Re	Collision Operator	Median Filter	m	Vortex Center (literature)	Vortex Center (LBM)	Time (sec)
1	100	BGK	No	10^5	(0.63, 0.79)	(0.63, 0.79)	17685
		BGK	No	10^8		(0.63, 0.79)	19398
		BGK-BRT	No	10^8		(0.63, 0.79)	22656
		BGK	Yes	10^8		(0.63, 0.79)	21535
		MRT	No	10^8		(0.63, 0.79)	79040
		MRT	Yes	10^8		(0.63, 0.79)	85295
1	1000	BGK	No	10^5	(0.54, 0.57)	(0.54, 0.57)	16109
		BGK	No	10^8		(0.54, 0.57)	17035
		BGK-BRT	No	10^8		(0.54, 0.57)	16879
		BGK	Yes	10^8		(0.54, 0.57)	20170
		MRT	No	10^8		(0.54, 0.57)	46048
		MRT	Yes	10^8		(0.54, 0.57)	55818
1	5000	BGK	No	10^5	(0.52, 0.53)	-	-
		BGK	No	10^8		-	-
		BGK-BRT	No	10^8		-	-
		BGK	Yes	10^8		(0.54, 0.53)	17248
		MRT	No	10^8		(0.51, 0.55)	50572
		MRT	Yes	10^8		(0.52, 0.53)	54225
1	10000	BGK	No	10^5	N/A	-	-
		BGK	No	10^8		-	-
		BGK-BRT	No	10^8		-	-
		BGK	Yes	10^8		(0.56, 0.60)	18186
		MRT	No	10^8		(0.48, 0.48)	43246
		MRT	Yes	10^8		(0.46, 0.54)	50864

40

Table 1.4: Bingham plastic, lid-driven cavity flow; $Bn = 10$.

Bn	Re	Collision Operator	Median Filter	m	Vortex Center (literature)	Vortex Center (LBM)	Time (sec)
10	100	BGK	No	10^5	(0.53, 0.87)	(0.54, 0.87)	25217
		BGK	No	10^8		(0.55, 0.87)	38285
		BGK-BRT	No	10^8		(0.55, 0.87)	29539
		BGK	Yes	10^8		(0.54, 0.87)	35817
		MRT	No	10^8		(0.53, 0.88)	143741
		MRT	Yes	10^8		(0.54, 0.88)	135059
10	1000	BGK	No	10^5	(0.80, 0.85)	(0.78, 0.83)	11525
		BGK	No	10^8		(0.78, 0.83)	26671
		BGK-BRT	No	10^8		(0.78, 0.83)	19248
		BGK	Yes	10^8		(0.79, 0.84)	34143
		MRT	No	10^8		(0.79, 0.84)	105136
		MRT	Yes	10^8		(0.79, 0.84)	111942
10	5000	BGK	No	10^5	(0.60, 0.55)	-	-
		BGK	No	10^8		-	-
		BGK-BRT	No	10^8		-	-
		BGK	Yes	10^8		(0.52, 0.55)	34638
		MRT	No	10^8		(0.55, 0.55)	111274
		MRT	Yes	10^8		(0.55, 0.53)	121685
10	10000	BGK	No	10^5	N/A	-	-
		BGK	No	10^8		-	-
		BGK-BRT	No	10^8		-	-
		BGK	Yes	10^8		(0.49, 0.54)	19351
		MRT	No	10^8		(0.53, 0.54)	69249
		MRT	Yes	10^8		(0.53, 0.53)	69054

Table 1.5: Bingham plastic, lid-driven cavity flow; $Bn = 100$.

Bn	Re	Collision Operator	Median Filter	m	Vortex Center (literature)	Vortex Center (LBM)	Time (sec)
100	100	BGK	No	10^5	(0.51, 0.95)	(0.51, 0.95)	13237
		BGK	No	10^8		(0.54, 0.96)	26782
		BGK-BRT	No	10^8		(0.53, 0.96)	33095
		BGK	Yes	10^8		(0.58, 0.96)	34430
		MRT	No	10^8		(0.49, 0.95)	161998
		MRT	Yes	10^8		(0.54, 0.96)	162085
100	1000	BGK	No	10^5	(0.53, 0.95)	(0.54, 0.95)	42233
		BGK	No	10^8		(0.64, 0.95)	47364
		BGK-BRT	No	10^8		(0.60, 0.92)	46289
		BGK	Yes	10^8		(0.79, 0.95)	48225
		MRT	No	10^8		(0.54, 0.95)	190119
		MRT	Yes	10^8		(0.55, 0.95)	188242
100	5000	BGK	No	10^5	(0.93, 0.97)	-	-
		BGK	No	10^8		-	-
		BGK-BRT	No	10^8		-	-
		BGK	Yes	10^8		(0.91, 0.95)	57354
		MRT	No	10^8		(0.92, 0.96)	163318
		MRT	Yes	10^8		(0.93, 0.95)	159523
100	10000	BGK	No	10^5	(0.92, 0.94)	-	-
		BGK	No	10^8		-	-
		BGK-BRT	No	10^8		-	-
		BGK	Yes	10^8		(0.84, 0.88)	32080
		MRT	No	10^8		(0.89, 0.91)	152707
		MRT	Yes	10^8		(0.89, 0.91)	145733

For flows with $Re \geq 5000$, the BGK with $m = 10^8$ and median filtering produced numerically stable results, but did not always seem to be consistent with literature values. The apparent inaccuracy with regards to BGK with $m = 10^8$ and median filtering is probably due to how the numerical stability is enhanced–namely through artificial, nonphysical dissipation. In contrast to the nonphysical natural in which median filtering enhances stability, the stability enhancement used by MRT does not directly affect the macroscopic hydrodynamics of interest, so it again provides a stable, and typically, the most accurate solution. The apparent superiority in terms of stability and accuracy of the MRT collision operator still comes at a price though. The MRT collision operator was, in general, 5-10 times slower than any of the BGK collision schemes. The extreme increase in computational expense is probably not due to the collision process itself, i.e. (1.9), but instead calculating the strain-rate (1.18) for each iteration of the solution for μ_{app}.

For the sake of completeness, the MRT collision operator was also tested with median filtering applied. There does not seem to be any reason to recommend the MRT collision operator with median filtering in practice because the MRT collision operator is stable all on its own and the artificial dissipation introduced by median filtering has the potential to adversely affect the accuracy of the results.

In summary,

- for low Reynold's number flows when computational efficiency is important, the BGK with $m = 10^5$ is recommended,

- for high Reynold's number flows when computational efficiency is important, the BGK with $m = 10^8$ and median filtering is recommended,

- and for high Reynold's number flows when accuracy is more important than computational efficiency, the MRT with $m = 10^8$ is recommended.

1.4.2.2 Power-law Fluids   For the power-law numerical tests the Reynold's number was varied, $Re = 100, 1000, 5000, 1000$, and the flow behavior index was varied, $n = 0.5, 1.5$ (Re was calculated the same as in Section 1.4.1.2). The collision schemes that were tested were: (1) BGK, (2) BGK-BRT, (3) BGK with median filtering, (5) MRT with $m = 10^8$ and MRT with $m = 10^8$ and median filtering. Table 1.6 and Table 1.7 compare the center location

of the main vortex to literature values taken from Li et al. [31]. The main vortex location is determined by calculating the stream function using Simpson's rule and finding where it attains a maximum.

For the shear-thinning results, $n = 0.5$, the BGK and BGK-BRT collision schemes were unstable when $Re \geq 1000$. Although there still appears to be somewhat of a trade off between computational efficiency and accuracy in regards to the BGK with median filtering and MRT collision schemes, it does not seem to be nearly as drastic as when simulating Bingham plastic lid-driven flow; e.g. neither collision scheme consistently agrees with the literature values; and the MRT collision operator is only 2-3 times slower than the BGK with median filtering.

For the shear-thickening results, $n = 1.5$, there are no issues of stability. What is interesting is that all of the collision schemes produce similar results for each of the cases (with the exception of the high Reynold's number case, $Re = 10000$), and the BGK-BRT is consistently the fastest solution. The reason the BGK-BRT may be the fastest is that because the collision frequency, ω, is bounded, and consequently μ_{app} is bounded, meaning the solution to the constitutive equation may be converging to a bound with less iterations than the other collision schemes.

In summary,

- for shear-thinning flows, either the BGK with median filtering or MRT collision schemes are recommended for stability reasons,

- for shear-thickening flows, there is much less concern for numerical instability.

1.5 CONCLUSION

A numerical investigation into the accuracy, stability, and efficiency of LBM collision models when applied to non-Newtonian flows was presented. The numerical investigation included testing the BGK and MRT collision operators, with and without entropic filtering, as applied to Bingham plastics and power-law fluid flows. Two different benchmark problems were

Table 1.6: Power-law, lid-driven cavity flow; $n = 0.5$.

n	Re	Collision Operator	Median Filter	Vortex Center (literature)	Vortex Center (LBM)	Time (sec)
0.5	100	BGK	No	(0.72, 0.78)	(0.71, 0.77)	21429
		BGK-BRT	No		(0.71, 0.77)	14547
		BGK	Yes		(0.72, 0.78)	33221
		MRT	No		(0.71, 0.77)	77287
		MRT	Yes		(0.71, 0.77)	110403
0.5	1000	BGK	No	(0.58, 0.55)	-	-
		BGK-BRT	No		-	-
		BGK	Yes		(0.53, 0.59)	24935
		MRT	No		(0.53, 0.54)	72102
		MRT	Yes		(0.53, 0.54)	78300
0.5	5000	BGK	No	(0.53, 0.52)	-	-
		BGK-BRT	No		-	-
		BGK	Yes		(0.63, 0.68)	23517
		MRT	No			
		MRT	Yes		(0.51, 0.58)	54225
0.5	10000	BGK	No	(0.53, 0.52)	-	-
		BGK-BRT	No		-	-
		BGK	Yes		(0.50, 0.55)	22603
		MRT	No			
		MRT	Yes		(0.50, 0.54)	63892

Table 1.7: Power-law, lid-driven cavity flow; $n = 1.5$.

n	Re	Collision Operator	Median Filter	Vortex Center (literature)	Vortex Center (LBM)	Time (sec)
1.5	100	BGK	No	(0.56, 0.73)	(0.57, 0.73)	10897
		BGK-BRT	No		(0.57, 0.73)	6239
		BGK	Yes		(0.56, 0.73)	11847
		MRT	No		(0.56, 0.73)	39017
		MRT	Yes		(0.56, 0.73)	96033
1.5	1000	BGK	No	(0.55, 0.64)	(0.54, 0.61)	19764
		BGK-BRT	No		(0.54, 0.61)	11949
		BGK	Yes		(0.54, 0.61)	24516
		MRT	No		(0.54, 0.61)	58714
		MRT	Yes		(0.54, 0.61)	69884
1.5	5000	BGK	No	(0.53, 0.61)	(0.52, 0.57)	20147
		BGK-BRT	No		(0.52, 0.57)	12116
		BGK	Yes		(0.52, 0.57)	23140
		MRT	No		(0.53, 0.57)	60584
		MRT	Yes		(0.52, 0.58)	58519
1.5	10000	BGK	No	(0.51, 0.55)	(0.53, 0.55)	21570
		BGK-BRT	No		(0.53, 0.55)	12841
		BGK	Yes		(0.53, 0.55)	21570
		MRT	No		(0.49, 0.56)	61453
		MRT	Yes		(0.49, 0.55)	67546

chosen for the flows: Poiseuille flow, and lid-driven square cavity flow. The results showed that:

- For high yield stress fluid, Poiseuille flows only the MRT collision operator did not suffer from numerical noise.
    - The noise appeared to be due to oscillations arising from high nonequilibrium values for the moment related to the square of the energy, ϵ, and the energy flux moment in the direction of flow, q_x.
    - If a collision scheme is to be developed for high yield stress, Poiseuille-type flows that is more accurate than the BGK with $m = 10^5$ and more computationally efficient than the MRT, then it should focus on dampening the nonequilibrium energy flux moment.

- Median filtering can be an effective technique for enhancing stability, especially for high Reynold's number flows; however, if the filter is not carefully tuned by properly adjusted the threshold, δ, or number of standard deviations, n_s, then the physical intregrity, and consquently accuracy, of the model can be adversely impacted.

- In general, the MRT collision operator is much more computationally expensive than its BGK counterpart and is some times even orders of magnitude slower.

- To summarize plastic, lid-driven cavity flows,
    - for low Reynold's number flows when computational efficiency is important, the BGK with $m = 10^5$ is recommended,
    - for high Reynold's number flows when computational efficiency is important, the BGK with $m = 10^8$ and median filtering is recommended,
    - and for high Reynold's number flows when accuracy is more important than computational efficiency, the MRT with $m = 10^8$ is recommended.

- To summarize power-law, lid-driven cavity flows,
    - for shear-thinning either the BGK with median filtering or MRT collision schemes are recommended for stability reasons,
    - for shear-thickening, lid-driven cavity flows, there is much less concern for numerical instability.

1.6   ACKNOWLEDGEMENTS

2.0   FREE SURFACE AND NON-NEWTONIAN SIMULATION OF WELLBORE CEMENTING USING THE LATTICE BOLTZMANN METHOD

2.1   ABSTRACT

Wellbore cementing is

2.2   INTRODUCTION

Cementing of oil and gas wellbores is important because if cement does not completely fill the annulus then pathways can form for fluids and gasses to flow through. The pathways need not be large–a pathway of SOMETHING is sufficient for enough loss of pressure at the wellhead for the well to lose SOME percent production. There are also environmental concerns as oils and gasses can migrate into ground water or the atmosphere causing environmental damage.

Of critical importance to drilling operations of all kinds is that barriers are placed to prevent gasses and fluids from migrating from one geological zone to another (i.e., zonal isolation is achieved). If zonal isolation is not achieved, gasses and fluids may migrate through the well to the surface and up into the atmosphere–potentially causing pollution, or at the very least, contributing to an increase in greenhouse gasses–or, worse yet, formation gasses and fluids can migrate into aquifers, potentially harming wild life and people. Furthermore, it can be argued that the most important, yet vulnerable, barrier in terms of achieving zonal isolation of a wellbore is the cement annulus that is created between the casing and the formation. There are various mechanisms by which this cement annulus can fail. According

to previous studies [6, 16, 30, 37, 42, 54], the most prevalent failure mechanisms of the cement annulus are:

- Stresses developing in the cement annulus as a result of temperature gradients, moisture gradients, overburden pressure, and chemical shrinkage of the cement matrix. These stresses can cause cracking in the cement, or debonding at either the casing–cement or cement–formation interfaces.

- Poor fill of the annulus. If cement slurry does not fill the annulus then if drilling mud is left behind it can dry, crack, and provide a weak path for fluids and gasses under pressure to push through, or if a dry hole is is cemented and not filled properly, fluids and gasses can travel through channels and voids left in the annular space.

- Gas channeling through the annulus during cement curing. Initially, after the annulus is poured, the cement column transmits its full hydrostatic pressure against the rock formation. As the cement annulus hydrates, it solidifies, becoming stiffer and more self-supporting. When the column begins to support itself, the hydrostatic pressure of the cement column–against the rock formation–begins to drop. If the hydrostatic pressure of the cement column drops below the pressure of a formation gas before the cement annulus has developed adequate strength to resist it, the formation gas can penetrate int the cement annulus and form channels, or degrade the cement annulus in other ways.

In order to better understand, and thus improve wellbore cementing for zonal isolation, several works have attempted to model certain aspects of the multiphase and multiphysics processes of wellbore cement placement. For example, Beirute and Flumerfelt [3] developed a mathematical model to "describe the miscible displacement of drilling muds by cement slurries under laminar flow conditions". Among their assumptions were that the process could be modeled in a quasi-steady manner, the volumetric flow rate was constant at each cross section of the flow, and the pressure fields were only dependent on the y-coordinate. The model was then used to explore the effect of various parameters on the displacement efficiency of the spacer, such as the density ratio between the mud and the spacer, the viscosity ratio between the mud and the spacer, the displacement rate, the displaced phase yield stress, and the displacing phase yield stress. It was concluded that displacement efficiency could

be improved by increasing the density of the displacing phase, increasing the viscosity of the displacing phase, (in general) decreasing the displacement rate, decreasing the yield stress of the displaced phase, and increasing the yield stress of the displacing phase. Bittleston et al. [6] developed a mathematical model for primary cementing of an oil well using a Hele-Shaw displacement model. The model was more general than the model developed by Beirute and Flumerfelt as it used less restrictive assumptions and was able to consider an eccentric annulus. Annulus eccentricity occurs often in wellbores and can sometimes be extreme as it is difficult to keep long strings of casing centered in an imperfectly drilled hole. The bulk fluid motion of the spacer and drilling mud during primary cementing were investigated. The intended use of the model was not to make general statements about what properties of the spacer and drilling mud were desirable, but to instead be used iteratively during the design process for the cement job. Results showed that the model met its intended purpose, and in particular, was able to simulate an unyielded channel of mud left on the narrow side of the annulus. Zulqarnain [54] used a computational fluids dynamics (CFD) solver with volume of fluid (VOF) method to simulate the cementing process of a 50 ft. section of wellbore. The model included the interfacial dynamics of, not just the spacer and drilling mud, but also the cement slurry–which was not considered in either Beirute and Flumerfelt or Bittleston et al.. The drilling mud and cement slurry properties were kept constant, while the spacer density, viscosity, and displacement rates were varied. Zulqarnain concluded that the high displacement efficiency occurred when the spacer was the same density as the drilling mud and had a smaller viscosity, similar to that of water. The displacement efficiency was poor for all scenarios in which the spacer and cement slurry densities were equal. Zhao et al. [52] developed a model simulating wellbore cementing using the Lattice Boltzmann Method. Zhao et al. used the model to investigate the displacement interface shape in a horizontal, eccentric annulus; and in order to simplify the model from a 3D to 2D flow, used the classical Hele-Shaw approach. It was found, in general, that a greater density ratio between the displacing phase, i.e. cement slurry, and displaced phase, i.e. drilling mud, and lower eccentricity, results in better displacement in a horizontal, eccentric annulus.

Overall, past studies have tended to idealize the rock formation as a straight wall. Although this is a desirable simplification to make for capturing bulk fluid phenomena, it does

not accurately describe what is happening locally at the surface of the rock formation. Flow near the surface of the formation, and in and around the imperfections of the formation geometry, could influence the bond of the cement annulus to the rock formation, and the fill of the cement slurry very near to the rock formation. If an adequate bond does not develop, debonding can occur as a result of thermal and mechanical stresses. Even if most of the annulus is filled by the cement slurry, a small gap at the cement–formation interface that stretches for a long enough span of the wellbore would still be susceptible to fluid and gas migration, i.e. loss of zonal isolation. One could even argue that the characteristic of the cement slurry flow is most important local to the cement–formation interface, and that considering imperfections in the rock formation is imperative to determining whether zonal isolation has been achieved.

The purpose of the current work is to develop a computational approach to investigate the flow of cement slurry in and around the–possibly–imperfect features of a rock formation in a vertical section of a wellbore, with the hope of better understanding what parameters of the primary cement job will lead to a good fill of the cement annulus and high bond strength at imperfect rock formation features (i.e., the purpose is an extension to the work present in Grasinger et al. [24]). The Lattice Boltzmann Method (LBM) was chosen as the most suitable modeling technique for the desired purpose because of its ability to model flow through and around complex geometries [44], such as those found at the surface of the rock formation, among other benefits. LBM also has other advantages as an approach for simulating wellbore cementing:

- LBM is well-suited for simulating non-Newtonian fluids because the strain-rate tensor is computed local to each node and is second-order accurate [26, 27]. Many fluids present in wellbore cementing such as the spacer, drilling mud, and most importantly, the cement slurry, are best modeled as non-Newtonian fluids.

- LBM is well-suited for simulating multiphase multicomponent flows and free-surface flows because it can handle complicated interface shapes between fluid phases and components for reasons similar to why the no-slip boundary condition works well with complex geometries.

- LBM is easily written in parallel. As hardware architectures shifts from one or a few cores

to, especially in the case of scientific computing, several CPUs, it becomes increasingly important that code for simulating complex and computationally expense phenomena be written in parallel.

The current work focuses on the problem of cementing a dry hole, i.e. an annulus that is not filled with drilling mud prior to cementing. In order to numerically investigate this problem, an LBM model for non-Newtonian and free-surface flow is developed. The details of the LBM model are outlined in Section 2.3. What follows in Section 2.4 is a preliminary study on how different characteristics of flow, e.g. cement slurry yield stress, cement slurry inlet velocity, etc., perform in regards to fill of the annulus for different wellbores with imperfect rock formation geometries. Section 2.5 are conclusions resulting from the numerical study and a discussion of future work.

2.3 NUMERICAL METHODS

2.3.1 Lattice Boltzmann Method

2.3.2 Yield Stress Flow

2.3.3 Free Surface Flow

A free-surface flow is a flow in which two-fluids exist but the dynamics of one fluid does not need to be explicitly modeled. Typically, the phase of the primary fluid, or the fluid of interest, is a liquid; and the secondary phase that does not need to be explicitly modeled is a gas, often times air at atmospheric pressure. The primary fluid is modeled using a standard CFD method, such as LBM, and the interaction of the secondary fluid on the primary fluid is modeled as a boundary condition at the interface between the two fluids. The interface is what is referred to as the free-surface. The advantage of modeling a multiphase multicomponent flow as a free-surface is that because the dynamics of the secondary fluid are not explicitly tracked and simulated the resulting algorithm is more computationally efficient in terms of both memory and CPU overhead. The algorithm for extending the

Lattice Boltzmann Method for free-surface flow used in this work follows the algorithm developed and presented in Körner et al. [25], Thürey et al. [45].

2.3.4 Capturing the Free-Surface

Consider at each node a cell of size $dx \times dx$ where dx is the distance between nodes and the cell is centered at the node. In order to track where the primary fluid is and where it is not, i.e. which cells contain the primary fluid, a variable is introduced, ϵ, which describes what fraction of the cell is covered by the primary fluid. Cells that have a fluid fraction of $\epsilon = 0$ contain no primary fluid, i.e. are empty; cells that have a fluid fraction of $\epsilon = 1$ are filled; and cells that have a fluid fraction $0 < \epsilon < 1$ are neither completely filled nor are they completely empty. The fluid fraction concept is similar to how fluids are tracked in Volume of Fluid (VOF) methods.

2.3.4.1   Mass Transfer   It is important to note that the particle distributions, $f_0, f_1, ..., f_8,$ ▮ cannot be used to directly determine the fluid fraction of a cell. Neither can the zeroth velocity moment of the particle distributions, $\rho = \sum_i f_i$. For the algorithm used in this work, neither can be considered a measure of the amount of primary fluid in a cell, i.e. $\epsilon(\mathbf{x}, t) \neq \epsilon(\mathbf{f}(\mathbf{x}, \mathbf{t}), \rho(\mathbf{x}, t))$. Instead, each f_i is related to the amount of primary fluid advection in each of the $\boldsymbol{\xi}_i$ discrete velocity directions. To track the fluid fraction and how it changes, another variable is introduced, $M(\mathbf{x}, t)$, which denotes the fluid mass in each cell. The fluid mass is related to the fluid fraction by $\epsilon = M/\rho$. The change in mass, $\Delta M(\mathbf{x}, t)$ is related to $\mathbf{f}(\mathbf{x}, t)$ and particle distribution functions in the neighborhood of $\mathbf{x}$. Given initial conditions on $M(\mathbf{x}, 0)$ and a fixed time step size Δt, the fluid mass is determined by $M(\mathbf{x}, t_n) = M(\mathbf{x}, 0) + \Delta t \sum_{k=1}^n \Delta M(\mathbf{x}, t_k)$.

2.3.4.2   Cell States   Each of the cells in the domain are assigned one of three states: fluid, interface, or gas. The set of fluid, interface, and gas cells are denoted by $\mathcal{F}$, $\mathcal{I}$, and $\mathcal{G}$, respectively. A fluid cell is a cell that is entirely covered by the primary fluid and a gas cell is a cell that is completely empty, or does not contain any primary fluid. An interface is

partially covered, and there is always a layer of interface cells between fluid and gas cells so that a fluid cell and a gas cell are never neighbors. The change in mass is calculated across fluid and interface cells for each discrete velocity direction as follows:

$$\Delta M_i(\mathbf{x}, t) = \begin{cases} f_{\bar{i}}(\mathbf{x} + \boldsymbol{\xi}_i \Delta t, t) - f_i(\mathbf{x}, t), & \mathbf{x} \in \mathcal{F} \text{ or } \mathbf{x} + \boldsymbol{\xi}_i \in \mathcal{F} \\ [(\epsilon(\mathbf{x}, t) + \epsilon(\mathbf{x} + \boldsymbol{\xi}_i \Delta t, t))/2] \left(f_{\bar{i}}(\mathbf{x} + \boldsymbol{\xi}_i, t) - f_i(\mathbf{x}, t) \right), & \mathbf{x} \in \mathcal{I} \text{ and } \mathbf{x} + \boldsymbol{\xi}_i \in \mathcal{I} \\ 0, & \mathbf{x} + \boldsymbol{\xi}_i \in \mathcal{G} \end{cases}$$

$$(2.1)$$

where $\boldsymbol{\xi}_{\bar{i}}$ represents the discrete velocity vector that is opposite in direction to $\boldsymbol{\xi}_i$, i.e. $\boldsymbol{\xi}_{\bar{i}} = -\boldsymbol{\xi}_i$. (2.1) states that when the border of the two cells is covered with fluid, the change in mass for a discrete velocity direction is equal to the difference of the particle distribution entering the cell from the opposite direction and particle distribution leaving the cell in that direction. When the border of the two cells are not covered with fluid, i.e. when both cells are interface cells, the difference is weighted by the average fluid fraction of the two cells because mass is only able to transfer across the fraction of the border that is covered in fluid. The total change in mass for a cell is simply $\Delta M(\mathbf{x}, t) = \sum_i \Delta M(\mathbf{x}, t)$. Finally, note that $\Delta M_i(\mathbf{x}, t) = -\Delta M_{\bar{i}}(\mathbf{x} + \boldsymbol{\xi}_i \Delta t, t)$ for all $\mathbf{x} \in \mathcal{X}$ where $\mathcal{X}$ is the domain. As a consequence, it is clear that $\sum_{\mathbf{x} \in \mathcal{X}} \Delta M(\mathbf{x}, t) = 0$, i.e. that mass is conserved for the mass transfer step.

Tracking the transport of mass throughout the domain is an important step in determining what parts of the domain are covered in the primary fluid and what are not. However, for the free-surface, i.e. layer of interface cells between the fluid cells and gas cells, to change shape and propagate, cells need to be able to transition state. Cell states change based on the rule set presented in Table 2.1 where δ_{trans} is a small, computational constant that keeps cells from transitioning back and forth between fluid and interface or interface and gas states in subsequent time steps. To summarize Table 2.1, gas cells cannot transition to fluid cells and vice versa, interface cells transition to fluid cells when their mass has exceeded ρ by a computational constant, and gas cells when their mass is less than zero minus a computational constant. In addition, fluid cells and gas cells transition to the interface state in order to maintain continuity of the interface, i.e. to ensure there is always a layer of interface cells

Table 2.1: Cell state transition rules.

Current State	Condition	Transition
$(\mathbf{x}, t) \in \mathcal{I}$	$M(\mathbf{x}, t) < -\delta_{trans}$	$(\mathbf{x}, t + \Delta t) \to \mathcal{G}, \; M(\mathbf{x}, t + \Delta t) = 0$
	$M(\mathbf{x}, t) > \delta_{trans} + \rho(\mathbf{x}, t)$	$(\mathbf{x}, t + \Delta t) \to \mathcal{F},$
		$M(\mathbf{x}, t + \Delta t) = \rho(\mathbf{x}, t + \Delta t)$
$(\mathbf{x}, t) \in \mathcal{F}$	$(\mathbf{x} + \boldsymbol{\xi}_i \Delta t, t + \Delta t) \in \mathcal{G}$ for any $i = 1, 2, ..., 8$	$(\mathbf{x}, t + \Delta t) \to \mathcal{I}$
$(\mathbf{x}, t) \in \mathcal{G}$	$(\mathbf{x} + \boldsymbol{\xi}_i \Delta t, t + \Delta t) \in \mathcal{F}$ for any $i = 1, 2, ..., 8$	$(\mathbf{x}, t + \Delta t) \to \mathcal{I}$

in between fluid cells and gas cells. An example of a cell transitioning state is depicted in Figure 2.1.

2.3.4.3 Updating Cell States and Mass Redistribution The rules for updating cell states in Table 2.1 are fairly intuitive, however, there are some bookkeeping nuances that must be considered in order to ensure mass is properly conserved and interface continuity is achieved; e.g. note that if the rules outlined in Table 2.1 are followed naively that mass is not conserved when interface cells transition to fluid cells or gas cells. The algorithm for updating cell states follows that which was presented in Thürey et al. [45]. It is presented here for the sake of completeness and for outlining minor modifications.

1. Initialize a variable that represents the global excess mass, e.g. $M_{gex} = 0$.

2. For all cells $\mathbf{x} \in \mathcal{I}$, store cells that should be transitioned to $\mathcal{F}$ in one data structure, $\mathcal{I}_f$, and store cells that should be transitioned to $\mathcal{G}$ in another, $\mathcal{I}_g$.

3. Loop over cells that will be transitioned from interface to fluid cells, i.e. loop over $\mathcal{I}_f$, and prepare their neighborhoods for transition:

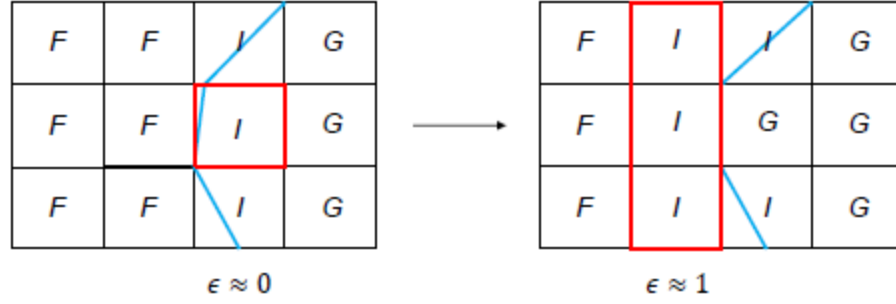$\epsilon \approx 0$ $\epsilon \approx 1$

Figure 2.1: Example of cell state transitions. The interface cell highlighted in red on the left is transitioned to a gas cell as it has emptied. On the right, the three cells highlighted in red were transitioned from fluid to interface cells in order to keep the interface continuous.

  a. Calculate the average density, ρ_{avg}, and velocity, $\mathbf{u}_{avg}$, of each neighborhood.

  b. For all cells in the neighborhood of the new fluid cell:

    i. If the cell is a gas cell, transition to interface state and set $f_i = f_i^{eq}(\rho_{avg}, \mathbf{u}_{avg})$, $\rho = \rho_{avg}$, and $\mathbf{u} = \mathbf{u}_{avg}$.

    ii. If the cell is an interface cell in $\mathcal{I}_g$, remove it from $\mathcal{I}_g$.

4. Loop over cells that will be transitioned from interface to gas cells, i.e. loop over $\mathcal{I}_g$, and prepare their neighborhoods for transition:

  a. For all cells in the neighborhood of the new gas cell:

    i. If the cell is a fluid cell, transition to interface state.

5. Loop over cells that will be transitioned from interface to fluid cells, i.e. loop over $\mathbf{x} \in \mathcal{I}_f$, and:

  a. Let $M_{ex} = M - \rho$.

  b. Calculate the unit norm outward from the interface, i.e. toward empty space, $\mathbf{n}$. Initialize $v_{sum} = 0$.

  c. For all cells in the neighborhood,

57

i. if an interface cell, store in a data structure, $\mathcal{I}_r$, and for the discrete velocity vector direction, $\boldsymbol{\xi}_i$, of the respective neighbor, let $v_i = \begin{cases} \mathbf{n} \cdot \boldsymbol{\xi}_i, & \mathbf{n} \cdot \boldsymbol{\xi}_i > 0 \\ 0, & \text{otherwise} \end{cases}$.

ii. $v_{sum} = v_{sum} + v_i$.

d. If $v_{sum} \geq 0$, then for each cell, $\mathbf{x}_j \in \mathcal{I}_r$, redistribute the excess mass according to $M(\mathbf{x}_j) = M(\mathbf{x}_j) + M_{ex}\frac{v_i}{v_{sum}}$.

e. If $v_{sum} = 0$ and $\mathcal{I}_r \neq \emptyset$, then for each cell, $\mathbf{x}_j \in \mathcal{I}_r$, redistributed the excess mass according to $M(\mathbf{x}_j) = M(\mathbf{x}_j) + M_{ex}/n_r$ where n_r is the number of cells in $\mathcal{I}_r$.

f. Otherwise, increment the global excess mass, $M_{gex} = M_{gex} + M_{ex}$.

g. Lastly, transition $\mathbf{x}$ from $\mathcal{I}$ to $\mathcal{F}$ and $M(\mathbf{x}) = \rho(\mathbf{x})$.

6. Loop over cells that will be transitioned from interface to gas cells, i.e. loop over $\mathbf{x} \in \mathcal{I}_g$, and:

   a. Let $M_{ex} = M$.

   b. Calculate the unit norm outward from the interface, i.e. toward empty space, $\mathbf{n}$. Initialize $v_{sum} = 0$.

   c. For all cells in the neighborhood,

   i. if an interface cell, store in a data structure, $\mathcal{I}_r$, and for the discrete velocity vector direction, $\boldsymbol{\xi}_i$, of the respective neighbor, let $v_i = \begin{cases} -\mathbf{n} \cdot \boldsymbol{\xi}_i, & \mathbf{n} \cdot \boldsymbol{\xi}_i < 0 \\ 0, & \text{otherwise} \end{cases}$.

   ii. $v_{sum} = v_{sum} + v_i$.

   d. If $v_{sum} \geq 0$ and $\mathcal{I}_r \neq \emptyset$, then for each cell, $\mathbf{x}_j \in \mathcal{I}_r$, redistribute the excess mass according to $M(\mathbf{x}_j) = M(\mathbf{x}_j) + M_{ex}\frac{v_i}{v_{sum}}$.

   e. If $v_{sum} = 0$ and $\mathcal{I} \neq \emptyset$, then for each cell, $\mathbf{x}_j \in \mathcal{I}_r$, redistributed the excess mass according to $M(\mathbf{x}_j) = M(\mathbf{x}_j) + M_{ex}/n_r$ where n_r is the number of cells in $\mathcal{I}_r$.

   f. Otherwise, increment the global excess mass, $M_{gex} = M_{gex} + M_{ex}$.

   g. Lastly, transition $\mathbf{x}$ from $\mathcal{I}$ to $\mathcal{G}$ and $M(\mathbf{x}) = 0$.

7. The global excess mass, i.e. the mass that could not be redistributed by other means, is redistributed according to:

   a. for all $\mathbf{x} \in \mathcal{I}$, $M(\mathbf{x}) = M(\mathbf{x}) + M_{gex}/n_i$ where n_i is equal to the number of cells in the interface set, $\mathcal{I}$.

The algorithm, as presented, ensures both mass conservation and continuity of the free-surface. Moreover, when mass redistribution inevitably occurs, it is first attempted to be redistributed and weighted in the direction in which the interface is propagating, e.g. as in Step 5.d or 6.d. If there are no interface cells in the direction of interface propagation, it is instead redistributed to neighboring interface cells, e.g. Step 5.e or 6.e. On the off-chance that there are no neighboring cells which are interface cells, the excess mass is added onto the global excess mass, M_{gex}, which is eventually redistributed among all of the interface cells, e.g. Step 7. The hierarchy of possible redistribution steps, though seemingly complex, ensures that mass redistribution is done in the most physically meaningful way as is possible. Lastly, note that the interface normal, $\mathbf{n}$, is calculated using a finite difference approximation,

$$
\mathbf{n}_{ij} = \frac{1}{2} \left\{ \begin{array}{c} \epsilon_{i-1,j} - \epsilon_{i+1,j} \\ \epsilon_{i,j-1} - \epsilon_{i,j+1} \end{array} \right\} \tag{2.2}
$$

where i is the index of the node in the x-direction and j is the index of the node in the y-direction.

2.3.5 Boundary Conditions at the Free-Surface

Two important considerations not yet discussed are how boundary conditions are implemented at the interface in order to capture the interaction between the primary phase and the secondary phase, and a still more subtle consideration, how particle distributions that are missing from the streaming step are reconstructed. The degrees of freedom, i.e. f_i, are not tracked for gas cells. After the streaming step, interface cells will be missing particle distributions, f_i, that would have streamed from gas cells–an issue that is depicted in Figure 2.2.

The missing particle distributions need reconstructed. Also momentum needs to be conserved at the interface. When the particle distributions are reconstructed, they will be reconstructed in such a way that momentum is conserved between the primary fluid and the secondary fluid. In order to perform the reconstruction, there are a few assumptions that need to be made:
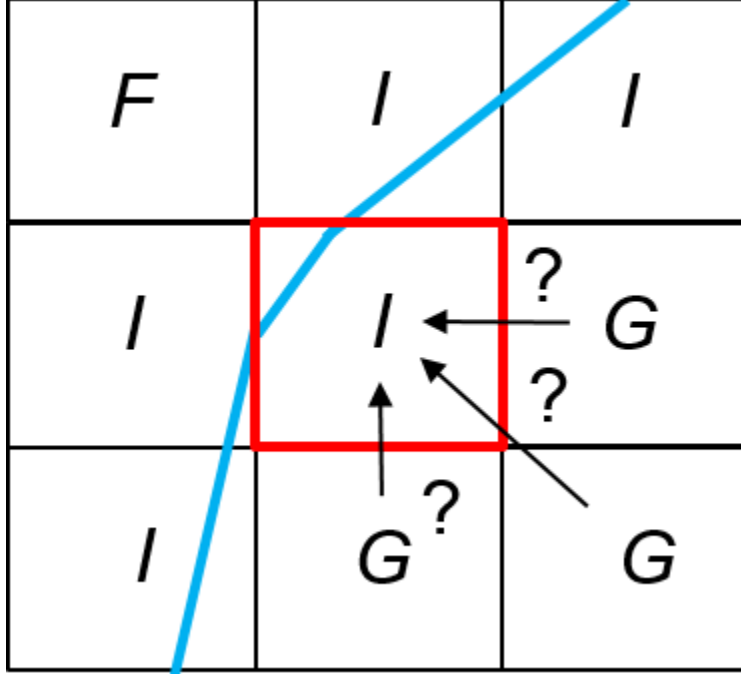
Figure 2.2: Example of a scenario in which particle distributions are missing post-streaming.

1. The velocity of the secondary fluid is equal to the velocity of the primary fluid at the free-surface, i.e. there is no-slip between the two-fluids.
2. The secondary fluid is at equilibrium and at constant pressure, p_G.
3. There is a force balance for opposite lattice directions.

Because particle distributions are a measure of momentum transport, the pressure of the secondary fluid can be converted to the particle distribution scale and the missing particle distribution can be solved for algebraically. Consider Figure 2.3. Balancing the opposite lattice directions results in:

$$f_i + f_{\bar{i}} = f_{\bar{i}}(\rho_G, \mathbf{u}) + f_i(\rho_G, \mathbf{u}), \tag{2.3}$$

$$f_{\bar{i}} = f_{\bar{i}}(\rho_G, \mathbf{u}) + f_i(\rho_G, \mathbf{u}) - f_i. \tag{2.4}$$

Missing distribution functions can be reconstructed according to (2.4). In order for conservation of momentum to be satisfied along the free-surface, distribution functions coming
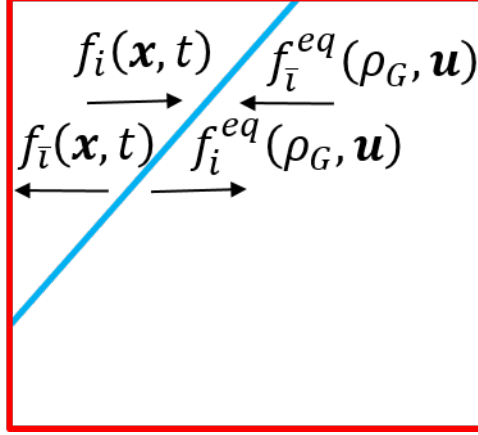
Figure 2.3: Particle distributions at free-surface for a pair of opposing lattice directions after gas pressure, ρ_G, is in the particle distribution form.

from the direction of the interface normal, $\mathbf{n}$, must also be reconstructed. Thus distribution functions are reconstructed if coming from a gas cell, or if $\mathbf{n} \cdot \boldsymbol{\xi}_{\bar{i}} > 0$.

2.3.6 Resulting Algorithm

The details of the free-surface algorithm have not been presented in chronological order, but instead in a logical order. For reasons of clarity, the steps of the algorithm are presented in order:

1. Mass transfer (Section 2.3.4.2)

2. Stream

3. Reconstruct particle distributions (Section 2.3.5)

4. Collide

5. Enforce boundary conditions

6. Calculate macroscopic variables

7. Update cell states (Section 2.3.4.3)

2.4   WELLBORE CEMENTING

The free-surface LBM model was implemented and used to simulate primary cementing in a dry annulus. The focus of the current study is the performance of different cement slurry mixes and characteristic flows for different imperfect wellbore geometries. Because the focus in regards to the wellbore is on geometrical imperfections, the entire wellbore annulus is not simulated, but instead a small section around a geometrical imperfection is considered. An example is shown in Figure 2.4.
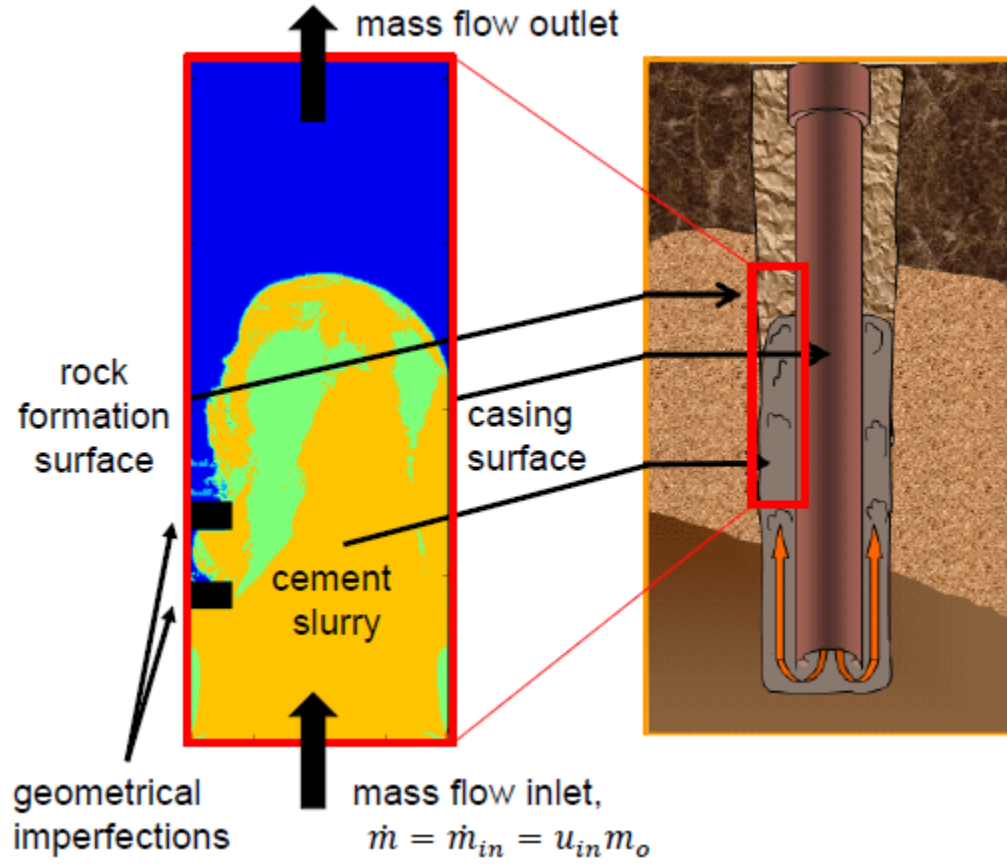
2.5   CONCLUSION

Figure 2.4: Example simulation used in wellbore cementing study. The black rectangles on the left boundary represent notches protruding from the rock formation surface. The contours are of the primary fluid (cement slurry) mass.

3.0   CONCLUSION

4.0 BIBLIOGRAPHY

[1] M. Ashrafizaadeh and H. Bakhshaei. A comparison of non-newtonian models for lattice boltzmann blood flow simulations. *Computers & Mathematics with Applications*, 58(5): 1045–1054, 2009.

[2] S. Bawazeer. Stability and accuracy of lattice boltzmann method. 2013.

[3] R. M. Beirute and R. W. Flumerfelt. Mechanics of the displacement process of drilling muds by cement slurries using an accurate rheological model. Society of Petroleum Engineers, 1977.

[4] P. Bhatnagar, E. Gross, and M. Krook. A model for collisional processes in gases i: small amplitude processes in charged and neutral one-component system. *Phys. Rev.*, 94, 1954.

[5] E. C. Bingham. *Fluidity and plasticity*, volume 2. McGraw-Hill Book Compny, Incorporated, 1922.

[6] S. Bittleston, J. Ferguson, and I. Frigaard. Mud removal and cement placement during primary cementing of an oil well–laminar non-newtonian displacements in an eccentric annular hele-shaw cell. *Journal of Engineering Mathematics*, 2:229–253, 2002.

[7] G. Bohme. Non-newtonian fluid mechanics. series in applied mathematics and mechanics, 1987.

[8] J. Boyd, J. Buick, and S. Green. A second-order accurate lattice boltzmann non-

newtonian flow model. *Journal of physics A: Mathematical and General*, 39:14241–14247, 2006.

[9] R. Brownlee, A. N. Gorban, and J. Levesley. Stabilization of the lattice boltzmann method using the ehrenfests' coarse-graining idea. *Physical Review E*, 74(3):037703, 2006.

[10] R. Brownlee, A. N. Gorban, and J. Levesley. Stability and stabilization of the lattice boltzmann method. *Physical Review E*, 75(3):036711, 2007.

[11] R. Brownlee, A. N. Gorban, and J. Levesley. Nonequilibrium entropy limiters in lattice boltzmann methods. *Physica A: Statistical Mechanics and its Applications*, 387(2):385–406, 2008.

[12] C. Cercignani. *Mathematical methods in kinetic theory.* Springer, 1990.

[13] Z. Chai, B. Shi, Z. Guo, and F. Rong. Multiple-relaxation-time lattice boltzmann model for generalized newtonian fluid flows. *Journal of Non-Newtonian Fluid Mechanics*, 166 (5):332–342, 2011.

[14] S. Chen and G. D. Doolen. Lattice boltzmann method for fluid flows. *Annual review of fluid mechanics*, 30:329–364, 1998.

[15] S. Chen, Q. Sun, F. Jin, and J. Liu. Simulations of bingham plastic flows with the multiple-relaxation-time lattice boltzmann model. *Science China Physics, Mechanics and Astronomy*, 57:532–540, 2014.

[16] P. Cheung and R. M. Beirute. Gas flow in cements. *Journal of petroleum technology*, 06:1041–1048, 1985.

[17] P. J. Dellar. Incompressible limits of lattice boltzmann equations using multiple relaxation times. *Journal of Computational Physics*, 190(2):351–370, 2003.

[18] D. d'Humieres. Generalized lattice-boltzmann equations. *Rarefied gas dynamics- Theory and simulations*, pages 450–458, 1994.

[19] D. d'Humières. Multiple–relaxation–time lattice boltzmann models in three dimensions. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 360(1792):437–451, 2002.

[20] K. Fallah, M. Khayat, M. H. Borghei, A. Ghaderi, and E. Fattahi. Multiple-relaxation-time lattice boltzmann simulation of non-newtonian flows past a rotating circular cylinder. *Journal of Non-Newtonian Fluid Mechanics*, 177:1–14, 2012.

[21] S. Gabbanelli, G. Drazer, and J. Koplik. Lattice boltzmann method for non-newtonian (power-law) fluids. *Physical Review E*, 72(4):046312, 2005.

[22] I. Ginzburg. A free-surface lattice boltzmann method for modelling the filling of expanding cavities by bingham fluids. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 360(1792):453–466, 2002.

[23] A. N. Gorban and D. J. Packwood. Enhancement of the stability of lattice boltzmann methods by dissipation control. *Physica A: Statistical Mechanics and its Applications*, 414:285–299, 2014.

[24] M. Grasinger, Z. Li, A. Vuotto, J. Brigham, A. Iannacchione, J. Vandenbossche, et al. Simulation of cement slurry flow to assess the potential for voids and channels in wellbore cementing processes. In *SPE Eastern Regional Meeting*. Society of Petroleum Engineers, 2015.

[25] C. Körner, M. Thies, T. Hofmann, N. Thürey, and U. Rüde. Lattice boltzmann model for free surface flow for modeling foaming. *Journal of Statistical Physics*, 121:179–196, 2005.

[26] T. Krüger, F. Varnik, and D. Raabe. Shear stress in lattice boltzmann simulations. *Physical Review E*, 79:046704, 2009.

[27] T. Krüger, F. Varnik, and D. Raabe. Second-order convergence of the deviatoric stress tensor in the standard bhatnagar-gross-krook lattice boltzmann method. *Physical Review E*, 82(2):025701, 2010.

[28] P. Lallemand and L.-S. Luo. Theory of the lattice boltzmann method: Dispersion, dissipation, isotropy, galilean invariance, and stability. *Physical Review E*, 61:6546–6542, 2000.

[29] J. Latt. *Hydrodynamic limit of lattice Boltzmann equations*. PhD thesis, University of Geneva, 2007.

[30] D. C. Levine, E. W. Thomas, H. Bezner, and G. C. Tolle. Annular gas flow after cementing: a look at practical solutions. Society of Petroleum Engineers, 1979.

[31] Q. Li, N. Hong, B. Shi, and Z. Chai. Simulation of power-law fluid flows in two-dimensional square cavity using multi-relaxation-time lattice boltzmann method. *Communications in Computational Physics*, 15(01):265–284, 2014.

[32] X. Niu, C. Shu, Y. Chew, and T. Wang. Investigation of stability and hydrodynamics of different lattice boltzmann models. *Journal of statistical physics*, 117(3-4):665–680, 2004.

[33] D. Packwood. Entropy balance and dispersive oscillations in lattice boltzmann models. *Physical Review E*, 80(6):067701, 2009.

[34] T. C. Papanastasiou. Flows of materials with yield. *Journal of Rheology (1978-present)*, 31:385–404, 1987.

[35] J. D. Sterling and S. Chen. Stability analysis of lattice boltzmann methods. *arXiv preprint comp-gas/9306001*, 1993.

[36] J. D. Sterling and S. Chen. Stability analysis of lattice boltzmann methods. *Journal of Computational Physics*, 123(1):196–206, 1996.

[37] R. Stewart and S. FC. Gas invasion and migration in cemented annuli: Causes and cures. *SPE drilling engineering*, 01:77–82, 1988.

[38] S. Succi. *The Lattice Boltzmann Equation for Fluid Dynamics and Beyond*. Oxford University Press, New York, 2001.

[39] O. Švec, J. Skoček, H. Stang, J. F. Olesen, and P. N. Poulsen. Flow simulation of fiber reinforced self compacting concrete using lattice boltzmann method. 2011.

[40] O. Švec, J. Skoček, H. Stang, M. R. Geiker, and N. Roussel. Free surface flow of a suspension of rigid particles in a non-newtonian fluid: A lattice boltzmann approach. *Journal of Non-Newtonian Fluid Mechanics*, 179:32–42, 2012.

[41] A. Syrakos, G. C. Georgiou, and A. N. Alexandrou. Performance of the finite volume method in solving regularised bingham flows: Inertia effects in the lid-driven cavity flow. *Journal of Non-Newtonian Fluid Mechanics*, 208:88–107, 2014.

[42] S. Talabani, G. Chukwu, and D. Hatzignatiou. A unique experimental study reveals how to prevent gas migration in a cemented annulus. Society of Petroleum Engineers, 1993.

[43] G. Tang, S. Wang, P. Ye, and W. Tao. Bingham fluid simulation with the incompressible lattice boltzmann model. *Journal of Non-Newtonian Fluid Mechanics*, 166:145–151, 2011.

[44] D. T. Thorne and C. Michael. *Lattice Boltzmann modeling: An introduction for geoscientists and engineers.* Springer, 2006.

[45] N. Thürey, C. Körner, and U. Rüde. Interactive free surface fluids with the lattice boltzmann method. *Technical Report05-4. University of Erlangen-Nuremberg, Germany*, 2005.

[46] A. Vikhansky. Lattice-boltzmann method for yield-stress liquids. *Journal of Non-Newtonian Fluid Mechanics*, 155:95–100, 2008.

[47] C.-H. Wang and J.-R. Ho. Lattice boltzmann modeling of bingham plastics. *Physica A: Statistical Mechanics and its Applications*, 387:4740–4748, 2008.

[48] C.-H. Wang and J.-R. Ho. A lattice boltzmann approach for the non-newtonian effect in the blood flow. *Computers & Mathematics with Applications*, 62(1):75–86, 2011.

[49] L. Wang, J. Mi, X. Meng, and Z. Guo. A localized mass-conserving lattice boltzmann approach for non-newtonian fluid flows. *Communications in Computational Physics*, 17 (04):908–924, 2015.

[50] R. A. Worthing, J. Mozer, and G. Seeley. Stability of lattice boltzmann methods in hydrodynamic regimes. *Physical Review E*, 56(2):2243, 1997.

[51] M. Yoshino, Y. Hotta, T. Hirozane, and M. Endo. A numerical method for incompressible non-newtonian fluid flows based on the lattice boltzmann method. *Journal of non-newtonian fluid mechanics*, 147:69–78, 2007.

[52] Y. Zhao, Z. Wang, Q. Zeng, J. Li, and X. Guo. Lattice boltzmann simulation for steady displacement interface in cementing horizontal wells with eccentric annuli. *Journal of Petroleum Science and Engineering*, 145:213–221, 2016.

[53] Q. Zou and X. He. On pressure and velocity boundary conditions for the lattice boltzmann bgk model. *Physics of Fluids (1994-present)*, 9:1591–1598, 1997.

[54] M. Zulqarnain. Simulations of the primary cement placement in annular geometries during well completion using computational fluid dynamics (cfd). Technical report, Baton Rouge, 2012.