fluids

MDPI

Review

Review of the Color Gradient Lattice Boltzmann Method for Simulating Multi-Phase Flow in Porous Media: Viscosity, Gradient Calculation, and Fluid Acceleration

Fizza Zahid [1,*] and Jeffrey A. Cunningham [2]

[1] Institute of Environmental Engineering and Research, University of Engineering and Technology, Lahore 54890, Pakistan
[2] Department of Civil and Environmental Engineering, University of South Florida, Tampa, FL 33620, USA; cunning@usf.edu
* Correspondence: fzahid1813@gmail.com

Abstract: The lattice Boltzmann method (LBM) is widely applied to model the pore-scale two-phase flow of immiscible fluids through porous media, and one common variant of the LBM is the color gradient method (CGM). However, in the literature, many competing algorithms have been proposed for accomplishing different steps in the CGM. Therefore, this paper is the first in a series that aims to critically review and evaluate different algorithms and methodologies that have been proposed for use in the CGM. Specifically, in this paper, we (1) provide a brief introduction to the LBM and CGM that enables and facilitates consideration of more sophisticated topics subsequently; (2) compare three methods for modeling the behavior of fluids of moderately different viscosities; (3) compare two methods for calculating the color gradient; and (4) compare two methods for modeling external forces or accelerations acting upon the fluids of interest. These topics are selected for the first paper in the series because proper selection of these algorithms is necessary and sufficient to perform two common "benchmark" simulations, namely bubble tests and layered Poiseuille flow. Future papers in the series will build upon these topics, considering more challenging conditions or phenomena. By systematically reviewing key aspects, features, capabilities, and limitations of the CGM, this series of papers will extend our collective ability to apply the method to a variety of important fluid flow problems in geosciences and engineering.

Keywords: lattice Boltzmann; color gradient; multi-phase flow; porous media; viscosity; interfacial tension; bubble test; Poiseuille flow

check for updates

Academic Editors: Akira Nakayama, Giuliano De Stefano and D. Andrew S. Rees

Received: 14 February 2025
Revised: 27 April 2025
Accepted: 8 May 2025
Published: 13 May 2025

Citation: Zahid, F.; Cunningham, J.A. Review of the Color Gradient Lattice Boltzmann Method for Simulating Multi-Phase Flow in Porous Media: Viscosity, Gradient Calculation, and Fluid Acceleration . *Fluids* **2025**, *10*, 128. https://doi.org/10.3390/fluids10050128

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

The flow of immiscible (multi-phase) fluids in porous media is important in many problems in geosciences and environmental engineering. Examples include water percolation through the vadose zone, the storage of supercritical carbon dioxide in deep saline aquifers, and the clean-up of groundwater contaminated by non-aqueous-phase liquids (e.g., petroleum and chlorinated solvents). To fully understand such systems, it is necessary to characterize fluid behavior at the pore scale, because macroscopic (bulk) fluid behavior depends on processes occurring at the pore scale [1]. Pore-scale factors that might be important include grain and pore morphology, inter-pore connectivity, interfacial tension, and fluid–fluid–solid interactions (i.e., wettability). However, quantifying the importance of such factors is challenging: the subsurface environment cannot be observed directly, and it is difficult to create physical (laboratory) models that are both realistic and controllable.

Because of the challenges of running controlled physical experiments with realistic porous media, pore-scale multi-phase fluid flow is often studied via numerical models that aim to accurately capture the most important physics of real systems. There are several different methods available that model pore-scale processes and systems, including pore network models, the lattice Boltzmann method (LBM), Monte Carlo methods, particle dynamics methods, and grid-based computational methods that track interfaces [2]. Many modern pore-scale models are able to sufficiently resolve multi-phase fluid interactions and adequately depict the behavior that is seen in laboratory experiments [3].

Of the methods developed heretofore, the LBM has proven to be a robust and reliable tool to simulate fluid flow (including multi-phase flow) in porous media [4–9]. A defining feature of the LBM is that it accounts for pore-scale fluid dynamics that satisfy the incompressible Navier–Stokes equations, without explicitly solving the partial differential equation [10]. Among the advantages of the LBM are that it is well suited to application in complex geometries such as porous media [11] and that computations are local and explicit, making the method amenable to parallel implementations [12–19].

Different types of lattice Boltzmann methods have been developed for describing multi-phase fluid flow. These include the pseudopotential model (sometimes called the "Shan-Chen" method) [20], free energy method [21], mean field theory [22], stabilized diffuse-interface model [23], and color gradient method (sometimes called the "Rothman-Keller" method) [24,25]. Each of these variants has its own strengths, weaknesses, and applications [14,16]. Over the years, the color gradient method (CGM) has matured into a tool that has characterized various flows in porous media [26].

The CGM was developed and implemented for immiscible fluids in porous media by Gunstensen et al. [25], based on the lattice gas model [24]. A feature of the CGM is its ability to keep the immiscible fluids sufficiently distinct and separate with the help of an interface thickness parameter. This minimizes the mixing of the fluids at the interface and reduces distortion [27]. This method also conserves momentum at the local and global scale [28], which makes it possible to implement different types of boundary conditions (free exit) that cannot be implemented with other LBM variants. However, the method's main strength is the ability to independently tune fluid and flow parameters (e.g., interfacial tension, contact angle, fluid densities, and interface thickness) with individual color gradient constants or parameters [29]. The CGM has been utilized to simulate various porous media fluid flow applications such as carbon dioxide transport in subsurface systems [30], water transport in proton exchange membrane fuel cells [31], temperature-driven fluid flow in heated microchannels [32], drainage and imbibition in oil-water systems [33], spontaneous imbibition in micromodels [34], the behavior of droplets on surfaces [35], the adsorption of fluids onto the surfaces of porous media [36,37], the flow of particle suspensions [38], the separation of gas mixtures [9], and the behavior of droplets during inkjet printing [39], to name just a few. Due to the CGM's popularity, some previous authors have compared its capabilities to those of other leading multi-phase LBM methods, establishing the CGM's capacity to capture the physics of multi-phase fluid flow phenomena [3,40–42].

The execution and application of the CGM have been described and reviewed previously by other authors, often as part of broader reviews of multi-phase LBM methods (e.g., [16,26,43]). However, certain important practical aspects of implementing the CGM (e.g., how to model the flow of fluids with widely differing densities or viscosities and how to implement "open" boundary conditions) have not yet been systematically reviewed in the previous literature. Also, improvements to the CGM continue to appear in the literature frequently, and some recent advancements to the CGM (e.g., how to best account for wettability and contact angles and how to extend the method from two spatial dimensions to three [35,42,44–46]) have not been considered by previous reviews of the

method. Finally, the previous review papers generally do not guide the reader in terms of algorithm development and the practical implementation of the CGM. Therefore, a need exists to systematically review the capabilities and limitations of the CGM, starting with fundamental aspects for readers who are not yet experts in the method, then progressing to more sophisticated and "modern" topics, all while emphasizing the practical aspects of implementing the algorithms reviewed.

Therefore, the overall objective of this series of papers is to critically review the capabilities and implementation of the CGM as a method of modeling pore-scale multiphase flow in porous media. We aim to provide a review that is helpful both to new users of the LBM and to more experienced modelers with a higher level of expertise. For the current paper, the first in the series, the specific objectives are as follows:

1. Provide a brief introduction to the LBM and CGM that subsequently enables and facilitates consideration of more sophisticated topics;
2. Critically review different methods for modeling the flow of fluids of moderately different viscosities;
3. Critically compare two methods for calculating the color gradient;
4. Critically compare two methods for modeling external forces or accelerations acting upon the fluids of interest;
5. Summarize some of the challenges in applying the CGM with respect to the topics listed immediately above.

The rationale for focusing on these particular aspects of the CGM in the first paper of the series is that understanding these algorithms will allow the reader to run two of the most common "benchmark" tests for lattice Boltzmann modeling, namely bubble tests (droplet tests) and layered Poiseuille flow; examples of these tests will be provided at the end of this paper. Future papers in the series will build upon the topics covered here, considering phenomena or conditions such as fluids of widely differing viscosity and/or density, how to introduce the "wettability" of solid surface and specify a desired contact angle, how to simulate fluid flow in a domain with an open boundary condition (e.g., fluid flushing out of a porous medium), and how to extend the CGM from two spatial dimensions to three. By systematically reviewing these key aspects and features, this series of papers will extend our collective ability to apply the CGM to a variety of important fluid flow problems in geosciences and engineering.

2. Review of LBM Basics

The lattice Boltzmann method (LBM) for modeling fluid flow is based on a marriage of two sub-fields of physics: lattice gas cellular automata and Boltzmann's kinetic theory of gases (i.e., statistical mechanics).

A lattice gas cellular automaton is a system in which fictitious "gas" particles reside on a grid (lattice) and move between nodes of the grid. Typically, each particle has the same mass and moves with the same speed as each other. When two particles arrive at the same node at the same time, the particles collide, and the particles' velocities (momenta) may be altered by the collision. Frisch et al. [47] demonstrated the remarkable result that if the lattice is formulated properly, and if the collisions are modeled in a manner that conserves particle mass and momentum, then the lattice gas cellular automaton is able to simulate the flow of realistic fluids. That is, the aggregate or macroscopic behavior of the particles on the lattice can be shown to satisfy the Navier–Stokes equations. This discovery by Frisch et al. [47] enabled lattice gas cellular automata to be developed as tools for computational fluid dynamics.

An important advancement to the pioneering work of Frisch et al. [47] was modifying the lattice gas model to account for the spatio-temporal evolution of *distributions* of

particles, or *probabilities* of particle velocities, rather than the movements of individual particles [48–50]. The evolution of the particle distributions can be described with the following equation, which, as pointed out by Wolfram [51], is similar to a Boltzmann transport equation:

$$f_i(\mathbf{x} + \mathbf{e}_i\,\delta t, t + \delta t) = f_i(\mathbf{x}, t) + \Omega_i(f(\mathbf{x}, t)) \quad (1)$$

where $f_i(\mathbf{x}, t)$ is the probability of a particle at location $\mathbf{x}$ having the velocity $\mathbf{e}_i$ at time t, δt is a small incremental time interval, and Ω_i is the collision operator that operates on f. Because particle movement is constrained to the lattice, there is a finite set of velocity vectors $\mathbf{e}_i$. Solving for the distribution function f allows for the simulation of macroscopic fluid behavior without the "noise" that is encountered when working with individual particles.

In practice, the time variable t is generally taken as a discrete integer-valued variable corresponding to the number of time steps taken; movement from one node on the lattice to an adjacent node requires one time step, and δt is always taken as one time step. Therefore, many papers in the literature re-write Equation (1) more compactly as the following:

$$f_i(\mathbf{x} + \mathbf{e}_i, t + 1) = f_i(\mathbf{x}, t) + \Omega_i(f(\mathbf{x}, t)) \quad (2)$$

In this form, the units on velocity $\mathbf{e}_i$ are lattice units per time step (l.u./t.s.); the reader must understand that it is implied that $\mathbf{e}_i$ is multiplied by one time step, so the addition $\mathbf{x} + \mathbf{e}_i$ is acceptable.

Figure 1 shows two possible lattice configurations that have been applied for modeling flow in two dimensions, along with the corresponding set of velocity vectors $\mathbf{e}_i$. These two configurations are commonly referred to as D2Q7 and D2Q9. The "D2" notation refers to the fact that the lattices are two-dimensional, and "Q7" or "Q9" refers to the number of neighbors for each node in the lattice (including the node itself as one of its neighbors, which allows for the possibility that a particle can remain stationary rather than moving to an adjoining node).

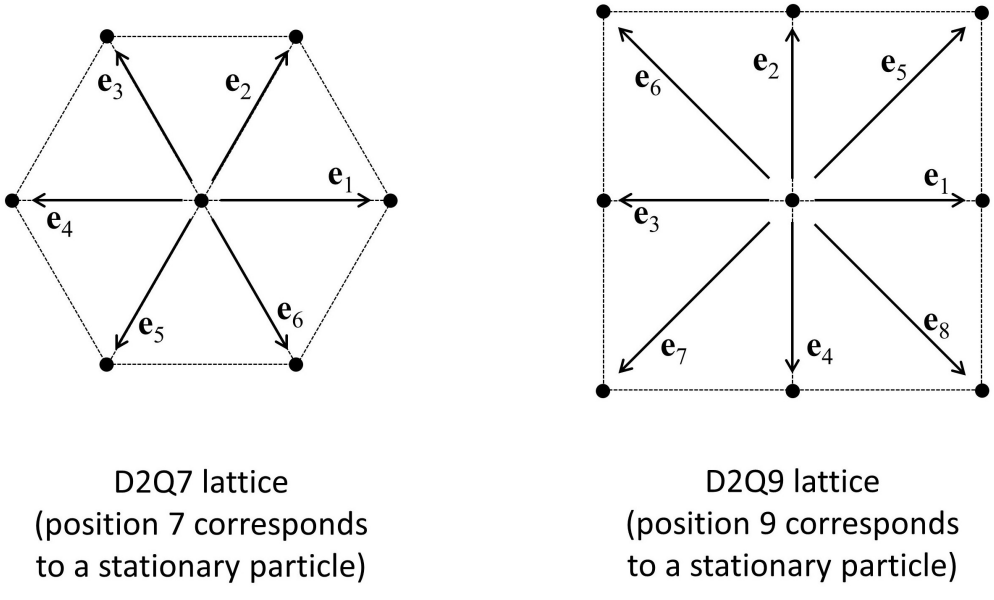


Figure 1. Two possible lattice configurations for simulating two-dimensional fluid flow.

In classical statistical mechanics, $f(\mathbf{x}, \mathbf{v}, t)$ is the probability density function for fluid particles at location $\mathbf{x}$ having velocity $\mathbf{v}$ at time t. The Boltzmann equation for the movement of the fluid particles (in the absence of an external force field) is

$$\frac{\partial f}{\partial t} + \mathbf{v} \cdot \nabla f = \Omega \tag{3}$$

where Ω represents the change in particle momentum due to collisions. It can be seen that Equation (2) is a form of Equation (3) in which the time derivative has been discretized and in which the particle velocities are constrained to the lattice. Therefore, Equation (2) is sometimes termed the "lattice Boltzmann equation".

A major advancement in the application of Equation (2) was developing a convenient way of estimating the collision operator. In classical statistical mechanics, Bhatnagar et al. [52] developed a method for estimating the collision operator in Equation (3); specifically, the collision operator is estimated as "relaxation" of the function f toward an equilibrium distribution f^{eq}. The method of Bhatnagar et al. [52] was applied to the lattice Boltzmann equation by Qian et al. [53]. Qian et al. [53] presented the following equations, which constitute the very popular Bhatnagar–Gross–Krook (or "BGK") formulation of the lattice Boltzmann method:

$$f_i(\mathbf{x} + \mathbf{e}_i\,\delta t, t + \delta t) = f_i(\mathbf{x}, t) + \frac{1}{\tau}\left[f_i^{\text{eq}}(\mathbf{x}, t) - f_i(\mathbf{x}, t)\right] \tag{4}$$

$$f_i^{\text{eq}}(\mathbf{x}, t) = w_i\,\rho\left[1 + 3(\mathbf{e}_i \cdot \mathbf{u}) + \frac{9}{2}(\mathbf{e}_i \cdot \mathbf{u})^2 - \frac{3}{2}(\mathbf{u} \cdot \mathbf{u})\right] \tag{5}$$

where τ is a relaxation time characterizing the particle collision, ρ is the fluid density (in the lattice domain), $\mathbf{u}$ is the macroscopic velocity vector of the fluid at location $\mathbf{x}$ and time t, and w_i is a weighting parameter. Equation (5) is the formulation of f^{eq} that enables the macroscopic fluid behavior to adhere to the Navier–Stokes equations [53]. For this formulation, the viscosity of the fluid (in the lattice domain, units l.u.^{2}/t.s.) is given by

$$\nu = \frac{1}{6}(2\tau - 1). \tag{6}$$

The weighting parameters, w_i, depend on which lattice geometry is employed. For a D2Q9 lattice (shown in Figure 1), w is given by the following [53–55]:

$$w = \left[\frac{1}{9}\ \frac{1}{9}\ \frac{1}{9}\ \frac{1}{9}\ \frac{1}{36}\ \frac{1}{36}\ \frac{1}{36}\ \frac{1}{36}\ \frac{4}{9}\right]. \tag{7}$$

For the flow of a fluid in two dimensions, the solution of Equations (4)–(7) enables us to track the particle distribution function f over time and space. However, Equation (5) depends on the macroscopic fluid properties ρ and $\mathbf{u}$. These can be determined by calculating the moments of f, according to the following:

$$\rho(\mathbf{x}, t) = \sum_i f_i(\mathbf{x}, t) \tag{8}$$

$$\mathbf{u}(\mathbf{x}, t) = \frac{\sum_i f_i(\mathbf{x}, t)\ \mathbf{e}_i}{\sum_i f_i(\mathbf{x}, t)} \tag{9}$$

Equations (4)–(9) constitute the foundation for simulating two-dimensional fluid flow with the lattice Boltzmann method. In practice, the method is implemented through an alternating series of "streaming" steps and "collision" steps that together satisfy Equation (4). More information about these steps is provided below in Section 3, but some details of implementation are omitted here in the interest of space. The reader is referred to previous work by Sukop and Thorne [6], Guo and Shu [43], Huang et al. [26], Krüger et al. [56], or Mohamad [57] for additional details of implementation.

It is worth noting that the LBM can also be used to simulate transport as well as fluid flow—for instance, mass transport via diffusion [58] and/or heat transfer via conduction [59]. Using the LBM to model transport rather than flow requires modification to the selection of the equilibrium distribution function f^{eq} and the relaxation time τ (e.g., [58,59]). However, in the present paper, we are concerned only with the multi-phase flow of immiscible fluids in porous media, and we do not consider diffusive mass transport or conductive heat transfer.

3. CGM Framework

The original CGM [25] was developed as a combination of the immiscible cellular gas automaton [24] and the LBM [50]. The method considers two immiscible fluids, a "red" fluid and a "blue" fluid, and thus considers two particle distribution functions f^k, where the superscript k indicates the red or the blue fluid. Particles of both fluids collide and stream in the lattice domain following the regular LBM operation described above (Section 2), but with an added "perturbation" step that enables interfacial tension to be included in the model, and with a "recoloring" step to maintain the fluids as immiscible.

Over the past three decades, the overall model algorithm has remained essentially unchanged, but improvements to the equations have resulted in better accuracy and better representations of true multi-phase fluid physics. Some early developments include increased capability to model fluids of unequal viscosity and/or density [60], improvement in the recoloring step [27], and modification of the perturbation equation [55]. These advances have enabled the model to be used for a wider range of fluid applications and have paved the way for developing three-dimensional models [42,61,62]. This section provides the framework of a two-dimensional CGM that includes the aforementioned improvements of Grunau et al. [60], Latva-Kokko and Rothman [27], and Reis and Phillips [55]. More recent updates pertinent to the contrast in fluid viscosity, calculation of the color gradient, and application of external forces are considered subsequently in Sections 4–6. Also, the extension of the two-dimensional CGM to three dimensions will be discussed in a future paper in this series.

The essence of the CGM can be expressed as the following analog to Equation (2) [62]:

$$f_i^k(\mathbf{x} + \mathbf{e}_i, t + 1) = f_i^k(\mathbf{x}, t) + \Omega_i^k(\mathbf{x}, t) \tag{10}$$

where the collision operator $\Omega_i^k(\mathbf{x}, t)$ now represents the aggregate of three steps: the single-phase collision $(\Omega_i^k)^1$, the perturbation operation $(\Omega_i^k)^2$, and a recoloring process. In practice, the order of the operations for implementing the CGM is as follows:

1. Calculate macroscopic variables ρ_{tot}, ρ^k, and $\mathbf{u}$;
2. Calculate equilibrium distributions for red and blue fluids, $f^{\text{eq},k}$;
3. Apply the single-phase collision operator, $(\Omega_i^k)^1$, for both fluids;
4. Calculate the "color gradient" between red and blue fluids, $\mathbf{C}(\mathbf{x}, t)$;
5. Apply the perturbation operator, $(\Omega_i^k)^2$, to account for interfacial tension;
6. Perform a "re-coloring" step to maintain the immiscibility of fluids;
7. Stream the particle distributions f^k.

These steps are discussed in more detail in the sub-sections following.

We note that there are variants in the CGM that do not exactly follow the sequence of steps listed above. For instance, one such variant [63–66] enables the whole algorithm to operate with any color distribution function. There can be advantages to such variants, including lower memory demands. However, in this review, we focus only on the more traditional CGM that relies on red and blue fluid populations, because of that method's ubiquity and relative ease of implementation.

3.1. Calculation of Macroscopic Variables

The first step in the algorithm is the calculation of the macroscopic variables: the total density of the combined (composite) fluid ρ_{tot}, the individual fluid densities ρ^k for the red and blue fluids, and the velocity $\mathbf{u}$ of the combined fluid. Analogous to Equations (8) and (9), these macroscopic variables can be calculated from the moments of the red and blue distribution functions:

$$\rho^k(\mathbf{x}, t) = \sum_i f_i^k(\mathbf{x}, t) \tag{11}$$

$$\rho_{\text{tot}}(\mathbf{x}, t) = \sum_k \rho^k(\mathbf{x}, t) \tag{12}$$

$$\mathbf{u}(\mathbf{x}, t) = \frac{\sum_k \sum_i \mathbf{e}_i f_i^k(\mathbf{x}, t)}{\sum_k \sum_i f_i^k(\mathbf{x}, t)} \tag{13}$$

3.2. Equilibrium Distribution Functions

The second step involves computing the equilibrium distribution functions $f^{\text{eq},k}$ for the red and blue fluids, which are used subsequently (step 3) to calculate the single-phase collision operator [67,68]:

$$f_i^{\text{eq},k}(\mathbf{x}, t) = \rho^k \left(C_i^k + w_i \left[3(\mathbf{e}_i \cdot \mathbf{u}) + \frac{9}{2}(\mathbf{e}_i \cdot \mathbf{u})^2 - \frac{3}{2}(\mathbf{u} \cdot \mathbf{u}) \right] \right) \tag{14}$$

For a D2Q9 lattice, w_i in Equation (14) is given by Equation (7) and C_i^k is given by the following [55,67]:

$$C_i^k = \begin{cases} \frac{1-\alpha^k}{5} & \text{if } i = 1, 2, 3, 4 \\ \frac{1-\alpha^k}{20} & \text{if } i = 5, 6, 7, 8 \\ \alpha^k & \text{if } i = 9 \end{cases} \tag{15}$$

In Equation (15), α^k is a parameter that introduces a density contrast between the fluids while maintaining stable hydrodynamic pressures at the interface (cf. [55,60]). The selected values of α^r and α^b determine the density ratio between the two fluids, the speed of sound within each fluid (c_s^k), and the pressure within the system, as summarized in Table 1. In Table 1, the parameter ζ, which relates c_s^k to α^k, is lattice-dependent; values are provided by Leclaire et al. [69]. For a D2Q9 lattice, $\zeta = \frac{3}{5}$.

To simulate a stable density contrast, α^k must be selected within the range $0 < \alpha^k < 1$ for both fluids. In particular, for a D2Q9 lattice, if $\alpha^k = 4/9$, Equation (14) reduces to Equation (5) for fluid k, and the speed of sound within fluid k is $c_s^k = 1/\sqrt{3}$. For other values of α^k, the speed of sound in fluid k will be different, as given in Table 1. To model the flow of two fluids of equal density, α^k for both fluids may usually be set equal to $4/9$. An introduction of different values of α^k for the two fluids is able to achieve density ratios up to about 18.5 [55]; systems with greater density ratios require a different approach, which will be reviewed in a subsequent paper in this series.

Note that to achieve a desired density ratio γ, there is not one unique pair of α^r and α^b that must be selected; for instance, selecting $\alpha^b = 0.02$ and $\alpha^r = 0.51$ will yield a density ratio $\gamma = 2$, but selecting $\alpha^b = 0.4$ and $\alpha^r = 0.7$ will also yield $\gamma = 2$. In limited trials during which we tested different combinations of α^r and α^b, we found that setting both values of α^k close to $4/9$ led to slightly better results than allowing one value of α^k to be far from that value. The selection of α^r and α^b according to the equations shown in Table 1 ensures that the pressure in the two fluids is continuous at the fluid–fluid interface [55,67].

It has been demonstrated [66] that when using the aforementioned $f^{\mathrm{eq},k}$ relations for fluids of unequal density, the tangential momentum of the two fluids is conserved along the fluid–fluid interface, but the tangential velocity along the fluid–fluid interface then becomes discontinuous. It is not yet clear if there is a practical implication of conserving momentum rather than velocity during simulations, nor if a different method for modeling unequal densities would be able to ensure the continuity of tangential velocities [66].

Table 1. User-specified CGM constants and their relations to fluid properties.

Symbol	Parameter	Fluid Relation
τ^k	Relaxation time of the kth fluid	Kinematic viscosity $\nu^k = \frac{1}{3}(\tau^k - 0.5)$
α^k	Density parameter of the kth fluid	Density ratio of the two fluids $\gamma = \frac{\rho^r}{\rho^b} = \frac{1-\alpha^b}{1-\alpha^r}$; speed of sound $c_s^k = \sqrt{\zeta(1-\alpha^k)}$; pressure $P^k = \rho^k (c_s^k)^2 = \rho^k \zeta (1-\alpha^k)$
A^k	Interfacial tension constant of the kth fluid	Interfacial tension σ; see discussion in Section 3.5 regarding the relationship between A^k and σ
β	Interface thickness parameter	See discussion in Section 3.6; $\beta \leq 1$

3.3. Collision Operator

In the next step, the collision operator $(\Omega_i^k)^1$ is computed and applied. There are different ways to compute the collision operator $(\Omega_i^k)^1$ depending on the viscosity ratio of the two fluids. The simplest collision expression is the BGK formulation, which considers that collisions induce the "relaxation" of f^k toward the equilibrium population $f^{\mathrm{eq},k}$, as discussed previously in Section 2:

$$(\Omega_i^k)^1 = \frac{-1}{\tau}\left[f_i^k(\mathbf{x},t) - f_i^{\mathrm{eq},k}(\mathbf{x},t)\right] \tag{16}$$

The challenge with applying Equation (16) is that it depends on a relaxation time τ that is related to the fluid viscosity (as shown in Equation (6)), but the fluid viscosity might not be the same for the red and blue fluids. If the red and blue fluids have a viscosity ratio that is very close to one (say, within a factor of two), then they will have similar values of τ, and Equation (16) can be applied without difficulty. If the fluids have a viscosity contrast that is slightly less than ~0.5 or slightly greater than ~2, an "effective" viscosity can be computed by interpolation, as discussed subsequently in Section 4. For even greater viscosity contrasts, more sophisticated approaches are required; for instance, the BGK approach can be replaced by a multiple-relaxation-time (MRT) approach, which will be discussed in a subsequent paper in this series.

3.4. Computing the Color Gradient

Next, the "color gradient" (sometimes also called "color flux" or "color field"), $\mathbf{C}(\mathbf{x},t)$, is calculated. The color gradient $\mathbf{C}(\mathbf{x},t)$ is the vector perpendicular to the interface between the red and blue fluids, as shown in Figure 2.

In the literature, different approaches have been presented for computing the color gradient. The original approach, as proposed and presented by Rothman and Keller [24], Gunstensen et al. [25], and Grunau et al. [60], is the following:

$$\mathbf{C}(\mathbf{x},t) = \sum_i \mathbf{e}_i \left[\rho^r(\mathbf{x}+\mathbf{e}_i,t) - \rho^b(\mathbf{x}+\mathbf{e}_i,t)\right] \tag{17}$$

Two notable modifications to this "original" approach are the "isotropic" color gradient of Leclaire et al. [70] and the "modified" color gradient of Xu et al. [71]. Subsequently in this paper (Section 5), we will compare some results of simulations using the original color gradient to the results of simulations using the isotropic color gradient of Leclaire et al. [70].

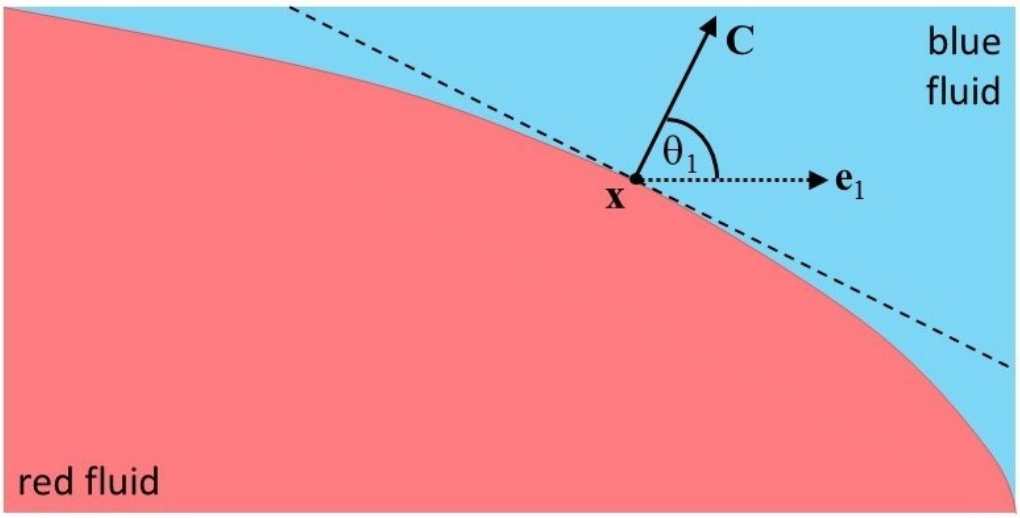


Figure 2. The color gradient **C**; figure modified from Reis and Phillips [55].

3.5. Perturbation

Gunstensen et al. [25] explain that the color gradient method proceeds by performing a collision step (as summarized above), then by "add[ing] a surface-tension-inducing perturbation, "then by recoloring the fluids and streaming the particles. In other words, the perturbation step of the CGM is the method of incorporating interfacial tension between the immiscible fluids; the color gradient **C** is utilized in this perturbation operation. In the paper of Gunstensen et al. [25], the perturbation step is represented by their Equation (17). Subsequently, perturbation strategies were proposed by Grunau et al. [60], Lishchuk et al. [72], Latva-Kokko and Rothman [27], and Reis and Phillips [55]. Ideally, the perturbation process should recover the correct interfacial force, reduce artificial velocities at the fluid–fluid interface (spurious velocities), retain the correct interface shape, and minimize interface thickness [62,72]. The approaches of Lishchuk et al. [72] and of Reis and Phillips [55] both fulfill these criteria. Furthermore, both methods have been applied successfully in both two-dimensional and three-dimensional models [42,62,72]. The approach of Lishchuk et al. [72] has been shown to model contact angles more accurately under certain conditions [45,73,74]. However, the approach of Reis and Phillips [55]—which is essentially an improvement upon the approaches of Grunau et al. [60] and Latva-Kokko and Rothman [27]—has been more extensively used because of its ability to include density ratios greater than one and because it is computationally less expensive. Here, we summarize the method of Reis and Phillips [55].

In this method, the perturbation step is accomplished through the application of the perturbation operator $(\Omega_i^k)^2$ that is applied to each f_i^k:

$$f_i^k(\mathbf{x}, t) = f_i^k(\mathbf{x}, t) + (\Omega_i^k)^2 \tag{18}$$

$$(\Omega_i^k)^2 = \frac{A^k}{2}|\mathbf{C}|\left[w_i \frac{(\mathbf{C}\cdot\mathbf{e}_i)^2}{\mathbf{C}\cdot\mathbf{C}} - B_i\right] \tag{19}$$

where the parameter A^k is a user-specified parameter (see Table 1) that controls the interfacial tension value. Equation (19) enables the conservation of mass and momentum, as explained by Reis and Phillips [55] and by Liu et al. [62]. In Equation (19), B is a vector of

values that depends on the lattice employed; the proper selection of B enables the recovery of the correct interfacial force term in the Navier–Stokes equations [55,62]. Specifically, for a D2Q9 lattice:

$$B = \left[\frac{2}{27} \; \frac{2}{27} \; \frac{2}{27} \; \frac{2}{27} \; \frac{5}{108} \; \frac{5}{108} \; \frac{5}{108} \; \frac{5}{108} \; \frac{-4}{27} \right]. \tag{20}$$

This method of introducing interfacial tension between the fluids has been shown to recover correct macroscopic equations for two-phase flows; detailed mathematical derivations are given by Reis and Phillips [55] and Liu et al. [62].

The interfacial tension between the two fluids, σ, is related to the A^k parameters [27,60]. Either the user can specify the A^k parameters, and the interfacial tension σ will result from the choice of the A^k, or the user can specify the desired interfacial tension σ, and values of A^k can be calculated accordingly. The problem is that the relationship between A^k and interfacial tension has not been completely resolved; different papers give different derivations or approximations of the relationship. Also, the relation between A^k and σ is known to depend on the fluid viscosities and the associated collision relaxation times τ; if the fluids have unequal viscosities, then they also have unequal values of τ, which complicates the relation between A^k and σ.

To simplify the situation somewhat, we found that, if the two fluids have equal densities or if the density ratio is acceptably low ($0.5 \leq \gamma \leq 2.0$), a single value of A can often be used for both fluids, i.e., $A^r = A^b = A$. In this case, A can be computed from a desired value of σ by following Equations (36) and (37) of Leclaire et al. [70]. That is, if the color gradient $\mathbf{C}$ is computed according to Equation (17), then A and σ are related by

$$A = \frac{3}{4} \frac{\sigma}{(\rho^r + \rho^b)\tau} \tag{21}$$

in which τ is an interpolated or effective relaxation time, which will be explained subsequently in Section 4. If, instead of using Equation (17) to calculate $\mathbf{C}$, we instead use the isotropic color gradient of Leclaire et al. [70] (which will be discussed in more detail subsequently), then A must instead be computed by the following:

$$A = \frac{9}{2} \frac{\sigma}{(\rho^r + \rho^b)\tau} \tag{22}$$

in which, again, τ is an interpolated or effective relaxation time. Notably, because the interpolated relaxation time τ varies both spatially and temporally (as will be seen subsequently in this paper), the parameter A also varies spatially and temporally, so it must be computed at every lattice location at every time step.

If Equations (21) and (22) do not successfully produce the desired interfacial behavior, then alternatively, the interfacial tension σ for selected values of A^k can be verified or estimated by conducting a bubble test (static droplet test) and measuring the simulated pressures and droplet radius [62,68,75]. The perturbation term and its relationship to interfacial tension are also considered in more detail in Appendix A of Ginzburg and Steiner [63].

3.6. Recoloring

After the perturbation step, the fluids are still mixed and require separation. This is achieved by a "recoloring" step. The original recoloring procedure [25] suffered from numerical issues such as "lattice pinning" and fluctuations in velocity (also called "spurious currents" or "anomalous currents") near the interface [27,76]. Subsequent recoloring algorithms proposed in the literature (e.g., [27,55,61,67]) were designed largely to address

these numerical issues. The algorithm proposed by Latva-Kokko and Rothman [27]—which is an improvement to that of D'Ortona et al. [77]—has been found to perform well for a variety of tests and is presented here:

$$f_i^r(\mathbf{x},t) = \frac{\rho^r}{\rho_{\text{tot}}} f_i^{\text{tot}} + \beta \frac{\rho^r \rho^b}{(\rho_{\text{tot}})^2} \cos(\Theta_i) f_i^{\text{eq},0} \tag{23}$$

$$f_i^b(\mathbf{x},t) = \frac{\rho^b}{\rho_{\text{tot}}} f_i^{\text{tot}} - \beta \frac{\rho^r \rho^b}{(\rho_{\text{tot}})^2} \cos(\Theta_i) f_i^{\text{eq},0} \tag{24}$$

where β is a user-specified parameter that controls the thickness of the interface (Table 1), and ρ_{tot} is the total fluid density as defined in Equation (12).

In Equations (23) and (24), $f_i^{\text{tot}}(\mathbf{x},t) = f_i^r(\mathbf{x},t) + f_i^b(\mathbf{x},t)$ and $f_i^{\text{eq},0}(\mathbf{x},t)$ are the equilibrium distribution function for the combined fluid under stationary conditions, i.e., evaluated at $\mathbf{u} = 0$. For the calculation of $f_i^{\text{eq},0}(\mathbf{x},t)$, we recommend using Equations (7), (14) and (15) given above, which differ slightly from Equations (1) and (2) of Latva-Kokko and Rothman [27]. Also, Θ_i represents the angle between the color gradient vector $\mathbf{C}$ and the lattice velocity vector $\mathbf{e}_i$, as shown in Figure 2 and given by the following equation:

$$\cos(\Theta) = \frac{\mathbf{e}_i \cdot \mathbf{C}}{|\mathbf{e}_i||\mathbf{C}|} \tag{25}$$

The parameter β in Equations (23) and (24) is specified by the user and is set between 0 and 1. A β value close to one gives minimum interface thickness but increases spurious velocities and reduces the accuracy of the interfacial tension value [62,68,78]. Conversely, if β is set too low, the spurious velocities at the interface reduce, but the interface thickness increases. Usually, a value of β between 0.70 and 0.99 is selected to maintain a thin interface without introducing significant spurious velocities (e.g., [79,80]), though Halliday et al. [78] argue that β should not exceed $\sqrt{2}/2 \approx 0.71$. Also, Leclaire et al. [81] offer a method for using a variable, rather than constant, value of β to improve performance. Despite these subtleties, we generally found that a constant value of $\beta \approx 0.8$ works well as a "default" value, unless it is determined that a more sophisticated approach is required.

In the literature, alternatives to the recoloring procedure have also been proposed. A few of these have been reviewed briefly by Ginzburg [66]. For instance, Kehrwald [65] proposes an "anisotropic diffusion" algorithm as an alternative to recoloring, and demonstrates that the algorithm is able to reduce the spurious currents (anomalous currents) mentioned above. However, most implementations of the CGM still rely on the recoloring algorithm described above, perhaps because it is relatively simpler to implement.

3.7. Summary of Parameters Specified by Model User

To summarize, the parameters that must be specified by the model user are those listed in Table 1.

- We generally recommend $\beta = 0.8$ as a default value for the interface thickness parameter β.
- For simulating fluids of equal density, we recommend $\alpha^r = \alpha^b = \frac{4}{9}$. For density ratios other than one, setting both values of α^k reasonably close to $4/9$ is recommended over setting one of the values far from $4/9$.
- For most applications, we recommend $A^r = A^b = A$, which can be calculated to produce a desired interfacial tension σ as given by Equation (21) or (22). However, A must be calculated at each lattice node at each time step because of the temporal and spatial variability of the effective relaxation time τ.

- Finally, the model user must specify either the relaxation time τ^k or the kinematic viscosity ν^k for each fluid, using Equation (6) to calculate whichever of the two was not specified explicitly.

4. Accounting for Fluids of (Moderately) Unequal Viscosity

The ratio of viscosity between the two fluids can strongly influence the overall behavior of the system. A classic example is "viscous fingering" that results from a less viscous fluid displacing a more viscous fluid in a porous medium [82,83]. The LBM is sometimes used as a tool to investigate and understand such mechanisms (e.g., [68,84–86]).

Although the CGM provides a robust framework to model multi-phase flow when the viscosity ratio is reasonably close to 1 (perhaps in the range 0.5–2), modifications to the methods of Section 3 are required if the viscosity ratio is far from 1. For instance, Rannou [87] demonstrated that the "standard" CGM could properly simulate two-phase flow in a channel for viscosity ratios as high as 10, but that simulated velocity profiles deviated from the correct values at higher viscosity ratios. This is because two fluids with unequal viscosities also have unequal values of τ, the relaxation time in the BGK collision operator, according to Equation (6). Therefore, at the interface between the two fluids, τ is not clearly defined, which can lead to incorrect calculations in the vicinity of the interface [68,88].

To address this problem, an effective value of τ near the interface can be interpolated from the adjacent known τ^k values before the first collision operator $(\Omega_i^k)^1$ is calculated. Methods available for interpolating τ have been presented by Grunau et al. [60], Tölke et al. [61], and Liu et al. [88], and are reviewed here.

In the method of Grunau et al. [60], an "order parameter" P (also sometimes called the "phase field" in the literature) can be calculated at any location $\mathbf{x}$ to indicate which of the two fluids, red or blue, dominantly occupies that location:

$$P(\mathbf{x},t) = \frac{\rho^r(\mathbf{x},t) - \rho^b(\mathbf{x},t)}{\rho^r(\mathbf{x},t) + \rho^b(\mathbf{x},t)} \tag{26}$$

Values of P range from -1 (region is occupied only by the blue fluid) to $+1$ (region is occupied only by the red fluid). The value of τ at any location $\mathbf{x}$ near the interface is determined from P and a user-specified interface parameter, δ, which must be ≤ 1:

$$\tau(\mathbf{x},t) = \begin{cases} \tau^r & P > \delta \\ g^r(P) & \delta \geq P > 0 \\ g^b(P) & 0 \geq P > -\delta \\ \tau^b & P < -\delta \end{cases} \tag{27}$$

$$\begin{aligned} g^r(P) &= s_1 + s_2 P + s_3 P^2 \\ g^b(P) &= t_1 + t_2 P + t_3 P^2 \\ s_1 = t_1 &= 2\frac{\tau^r \tau^b}{\tau^r + \tau^b} \\ s_2 &= 2\frac{\tau^r - s_1}{\delta} \\ s_3 &= -\frac{s_2}{2\delta} \\ t_2 &= 2\frac{t_1 - \tau^b}{\delta} \\ t_3 &= -\frac{t_2}{2\delta} \end{aligned} \tag{28}$$

Tölke et al. [61] use the same order parameter P as Grunau et al. [60], but the calculation of τ based on P is different. Specifically, Tölke et al. [61] use a simple linear interpolation to compute τ in the vicinity of the fluid–fluid interface:

$$\tau(\mathbf{x},t) = \frac{1}{2}(1+P(\mathbf{x},t))\tau^r + \frac{1}{2}(1-P(\mathbf{x},t))\tau^b \tag{29}$$

Finally, the interpolation scheme of Liu et al. [88] is based on a method devised by Zu and He [89] for use in the free-energy method. This interpolation scheme again uses the same order parameter P, but now τ is evaluated as a weighted harmonic mean of τ^r and τ^b:

$$\frac{1}{\tau(\mathbf{x},t)-0.5} = \frac{1+P(\mathbf{x},t)}{2(\tau^r-0.5)} + \frac{1-P(\mathbf{x},t)}{2(\tau^b-0.5)} \tag{30}$$

Figure 3 demonstrates the effectiveness of these three popular interpolation schemes for the classic benchmark simulation of two-phase flow in a channel (layered Poiseuille flow). For Figure 3, the testing domain is a 10×200 lattice domain with periodic boundary conditions and with a force applied to the fluid in the middle of the channel (methods for applying a force to the fluid will be described subsequently in Section 6). Other details regarding the simulation parameters are provided by Liu et al. [88]. In this figure, we simulated the flow of two fluids with a viscosity ratio of 12.5, i.e., the fluid in the middle of the channel (from lattice units 50 to 150) has a higher viscosity than that of the fluid near the edges of the channel. The CGM is used according to Section 3, and the LBM results are compared to a known analytical solution provided in Chapter 2 of Huang et al. [26] (cf., [32,40,87]). In Figure 3, we generated the results in the top-left panel without viscosity interpolation, and we generated the results in the remaining panels using the viscosity interpolation methods of Grunau et al. [60], Tölke et al. [61], and Liu et al. [88], as indicated on the figure. The improvement afforded by each viscosity interpolation is obvious.

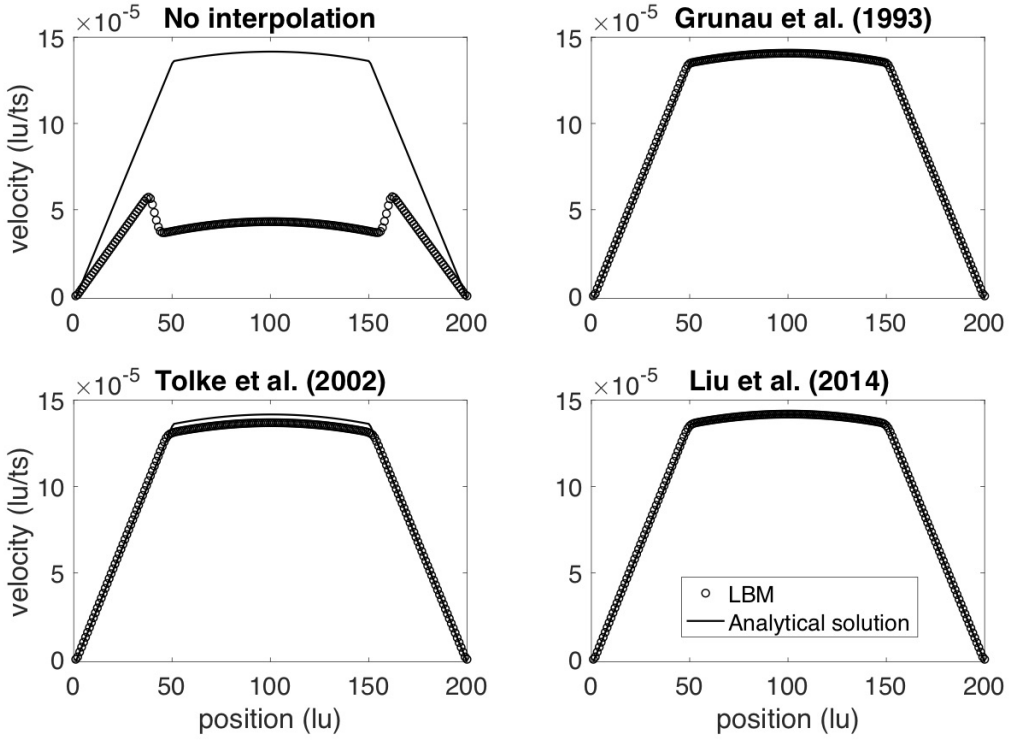


Figure 3. Predicted velocity profiles for two-phase flow in a channel, without and with the τ interpolation schemes [60,61,88]. The viscosity ratio of the two fluids is 12.5, i.e., the fluid in the center of the channel has a viscosity higher than that of the fluid at the edges of the channel.

All three interpolation schemes considerably reduce the error between the analytical and LBM velocities. Among these, the methods of Grunau et al. [60] and Liu et al. [88] perform similarly and accurately while the method of Tölke et al. [61] can not match the analytical solution completely. The methods of both Grunau et al. [60] and Liu et al. [88] have been successfully implemented and are popularly used [42,55,68,85,90]. The choice of which method to use is mostly up to the user's preference, but we note that the method of Liu et al. [88] requires fewer computations without any apparent degradation of performance. We therefore use the method of Liu et al. [88] in any simulations discussed subsequently in this paper (Sections 5 and 6).

In many applications, such as in simulations of drainage or imbibition, the fluid–fluid interface moves over time. Thus, the values of $P(\mathbf{x}, t)$ change over space and time; so too then do the values of the interpolated viscosity $\tau(\mathbf{x}, t)$. This means that any other variables that depend on τ—for instance, A in Equations (21) and (22)—also change over space and time. The values of P, τ, and A must therefore be recalculated at each lattice node at each time step.

As can be seen from Figure 3, the use of viscosity interpolation can, under some conditions, allow the CGM to perform well when the two fluids are of unequal viscosities. However, if the ratio of the fluid viscosities becomes extreme (e.g., the viscosity ratio is greater than $\sim$10), viscosity interpolation might not be sufficient to ensure that simulations are stable and accurate. Furthermore, the results observed in Figure 3 depend not only on the ratio of ν^r/ν_b but also on the specific values of ν^r and ν^b selected; choosing different values of ν^r and ν^b might alter the results even if the ratio ν^r/ν^b is held constant. BGK solutions are not necessarily fixed by non-dimensional numbers [66,91,92]. For either or both of these reasons, it might be necessary in some situations to employ more sophisticated approaches, such as replacing the BGK collision algorithm with a multiple relaxation time (MRT) algorithm. The application of the MRT to the color gradient method will be considered and reviewed in a future paper in this series.

When specifying values near an interface, as discussed immediately above, Ginzburg [66] distinguishes between an "implicit" interface approach (in which a blue fluid node is adjacent to a red fluid node, with an interface implied to exist halfway in between) and an "explicit" interface approach (in which a grid node is located at the interface, with blue fluid nodes on one side and red fluid nodes on the other side). For the case of an implicit interface, it is possible to select the CGM parameters in a way that the lattice Boltzmann method exactly reproduces the analytical solution for two-phase Poiseuille flow [66]. For the case of an explicit interface, Ginzburg [66] argues for using the harmonic mean value of viscosity, ν, rather than relaxation time, τ. This approach ensures the continuity of the tangential shear stress at the interface and the exact momentum solution inside each phase in two-phase Poiseuille flow, but needs to extrapolate it from the bulk to an explicit interface. Users may therefore wish to adopt the approach of Ginzburg [66] rather than interpolating values of τ. Nevertheless, for relatively simple problems, interpolating the value of τ near the interface is an expedient way of accounting for moderate differences in fluid viscosities, as demonstrated by Figure 3.

5. Comparison of Two Methods for Computing the Color Gradient

In Section 3.4, we presented the original method of calculating the color gradient as proposed by Rothman and Keller [24], Gunstensen et al. [25], and Grunau et al. [60]. This method of calculating the color gradient has been used successfully in many applications (e.g., [29,55]). However, Leclaire et al. [70] noted that standard color gradient models (as described heretofore in this paper) are plagued by problems such as spurious fluid velocities near fluid interfaces and the inability to model fluids of widely varying densitities,

and that these problems are due partly to the fact that the calculation of the color gradient (Equation (17)) contains discretization errors that are directionally biased (anisotropic). Leclaire et al. [70] therefore presented alternative ways of calculating the color gradient that remove the anisotropy of the discretization errors, which improves the accuracy and robustness of the model. We here refer to these as "isotropic" color gradient calculations, in contrast to the "original" color gradient calculation presented in Equation (17).

For a D2Q9 lattice, as shown in Figure 1, the following modification to Equation (17) can be used [70]:

$$\mathbf{C}(\mathbf{x},t) = \sum_i \mathbf{e}_i \, \xi_i \left[\rho^r(\mathbf{x}+\mathbf{e}_i,t) - \rho^b(\mathbf{x}+\mathbf{e}_i,t) \right] \tag{31}$$

in which the weights ξ_i are given by the following:

$$\xi = \left[\frac{1}{3} \; \frac{1}{3} \; \frac{1}{3} \; \frac{1}{3} \; \frac{1}{12} \; \frac{1}{12} \; \frac{1}{12} \; \frac{1}{12} \; 0 \right]. \tag{32}$$

For other lattice schemes (e.g., D3Q15 and D3Q27), the reader is referred to Leclaire et al. [70].

In Figure 4, we compare the original gradient approach to the isotropic gradient approach by applying both approaches to the problem of a "bubble test" or "droplet test". For this test, a red fluid droplet of radius R is emplaced in a field of blue fluid, the system is allowed to equilibrate, and then the pressure drop is measured across the interface of the droplet. In two dimensions, the theoretical pressure drop across the interface is $\Delta P = \sigma / R$, where σ is the interfacial tension specified by the user. The measured value of ΔP from the CGM simulation can be compared to the theoretical value of ΔP as a way to validate the CGM method. In Figure 4, the top row of panels shows the shape of the bubble after equilibration under different test conditions, and the bottom row of panels shows the agreement between measured and theoretical values of ΔP for those test conditions. For all simulations, the overall domain size was 128 × 128 l.u.; the specified interfacial tension was $\sigma = 0.0010$ (in LBM units); the interface thickness parameter $\beta = 0.85$; the viscosity $\nu^r = \nu^b = 0.16667$ l.u.^{2}/t.s.; the simulations were allowed to equilibrate for 80,000 time steps before ΔP was measured; and the radius of the red droplet was varied between 16 and 32 l.u. (the top row of panels shows the results for the cases in which $R = 32$ l.u.).

The left-most column in Figure 4 shows the results when $\rho^r = 2.0$, $\rho^b = 1.0$, and the original gradient is used. It can be seen that the droplet retains a mostly circular shape (panel (A)) and that the measured values of ΔP agree well with the theoretical relationship (panel (D)). A user might reasonably conclude that the original gradient calculation is acceptable for these test conditions. The center column of the figure shows the results when $\rho^r = \rho^b = 1.0$ and the original gradient is used. In this case, the measured values of ΔP are still close to the theoretical values (panel (E)), but the shape of the droplet is noticeably non-circular (anisotropic; panel (B)). The error in the gradient calculation is observable in the anisotropic shape of the droplet. Therefore, the right-most column of Figure 4 shows the results from the same conditions as the center column, but using the isotropic gradient calculation instead of the original gradient calculation. The measured values of ΔP are slightly improved (panel (F)), and the droplet shape attains the desired circular shape (panel (C)). A comparison of panel (B) to panel (C) shows that the isotropic gradient is clearly preferable to the original gradient under these test conditions.

Moreover, there is a distinct difference in the generated spurious velocities in the original color gradient and the isotropic color gradient methods. For the test case shown in Figure 4 when R = 32 l.u., the maximum spurious velocities are equal to 2.3×10^{-4} l.u./t.s. when using the original gradient calculation but are reduced to 7.1×10^{-6} l.u./t.s. when the isotropic color gradient is used. These results are in

agreement not only with the observations of Leclaire et al. [70] but also with those of Lafarge et al. [93] and Mora et al. [94]. The subject of spurious velocities in these simulations will be revisited in Section 7.

Therefore, in general, we recommend the isotropic gradient calculation (Equations (31) and (32)) over the original gradient calculation (Equation (17)). The simulation results are improved without any meaningful increase in the complexity or run time of the computer codes. However, when employing the isotropic color gradient, the user must be certain to use Equation (22) rather than Equation (21) when calculating the A parameter to yield a desired value of σ.

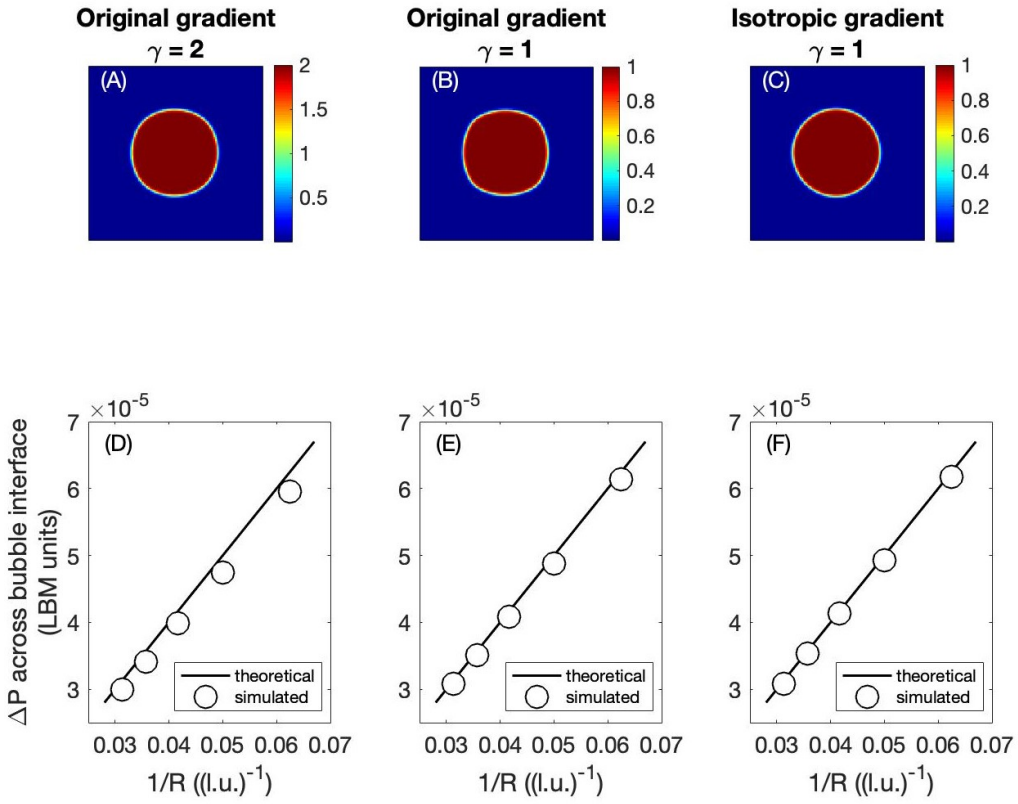


Figure 4. A comparison of the original and isotropic color gradient methods using a droplet test (bubble test) as a validation.

6. Application of a Force or Acceleration to the Fluids

A fluid flows in response to a pressure gradient or in response to a force acting on the fluid. Therefore, to simulate either single-phase or multi-phase flow, we must have methods for incorporating pressure gradients and/or external forces acting on the fluid(s). One way to do this is by specifying boundary conditions, e.g., specifying the fluid pressure or the fluid velocity on the boundaries of the domain [95]. Another approach is by including forces into the LBM simulations; methods that incorporate forces are often referred to as "forcing schemes" [56,96].

Many forcing schemes have been presented in the literature; a thorough review has been provided by Bawazeer et al. [96] for the flow of a single fluid. Here, we consider two of the most common forcing schemes and demonstrate how to apply them to a multi-phase flow simulation. In the first approach, originally proposed by Shan and Chen [20], external forces acting on the fluid are accounted for by including an acceleration term in the equilibrium distribution function f^{eq}. In the second approach, proposed by Luo [97,98],

a source term is added to include the force or acceleration as part of the collision step of the CGM.

In the method of Shan and Chen [20], the macroscopic velocity $\mathbf{u}$ in the computation of $f^{\mathrm{eq},k}$ is replaced by a new velocity $\mathbf{u}^{\mathrm{eq},k}$ that includes the effects of an acceleration $\mathbf{a}^k$ acting on fluid k. The acceleration $\mathbf{a}^k$ need not necessarily be the same on the red fluid and the blue fluid. Thus, Equation (14) becomes

$$f_i^{\mathrm{eq},k}(\mathbf{x},t) = \rho^k\left(C_i^k + w_i\left[3\left(\mathbf{e}_i\cdot\mathbf{u}^{\mathrm{eq},k}\right) + \frac{9}{2}\left(\mathbf{e}_i\cdot\mathbf{u}^{\mathrm{eq},k}\right)^2 - \frac{3}{2}\left(\mathbf{u}^{\mathrm{eq},k}\cdot\mathbf{u}^{\mathrm{eq},k}\right)\right]\right) \tag{33}$$

in which

$$\mathbf{u}^{\mathrm{eq},k}(\mathbf{x},t) = \mathbf{u}(\mathbf{x},t) + \mathbf{a}^k\,\tau \tag{34}$$

and τ represents the interpolated relaxation time at location $\mathbf{x}$ and time t. The remainder of the color gradient algorithm remains unchanged. Thus, in practice, only the calculation of $f^{\mathrm{eq},k}$ needs to be changed to implement this forcing scheme.

The method of Luo [97,98] accounts for forces in a different manner than the method of Shan and Chen [20]. Here, the macroscopic velocity $\mathbf{u}$ and the calculation of $f^{\mathrm{eq},k}$ remain unchanged, i.e., Equation (14) is used as presented. To account for the acceleration $\mathbf{a}^k$ acting on the fluids, a source term S_i^k is calculated as:

$$S_i^k(\mathbf{x},t) = w_i\frac{\rho^k(\mathbf{x},t)}{\left(c_s^k\right)^2}\left(\mathbf{e}_i\cdot\mathbf{a}^k\right) \tag{35}$$

in which w_i is given by Equation (7). The source term S_i^k must be calculated for both fluids at each location $\mathbf{x}$, each time t, and for each of the nine directions. As with the method of Shan and Chen [20], the method of Luo [97,98] allows for a different acceleration acting on the red and blue fluids. The source term is then incorporated into the collision step as follows:

$$f_i^k(\mathbf{x}+\mathbf{e}_i, t+1) = f_i^k(\mathbf{x},t) + (\Omega_i^k)^1 + S_i^k(\mathbf{x},t) \tag{36}$$

in which $(\Omega_i^k)^1$ is given by Equation (16) assuming that the BGK algorithm is used.

We compared the two forcing schemes described above via a layered Poiseuille flow validation test. In this test, red and blue fluids both flow between two parallel plates. A layer of red fluid is "sandwiched" in the middle of the channel between two layers of blue fluid. An acceleration a is applied in the $+x$ direction to the red fluid only, which induces flow. The situation is depicted graphically in Figure 5. This is a classic validation test for simulations of multi-phase flow because there is an analytical solution for the resulting velocity profile in the channel; see, for instance, Liu et al. [88].

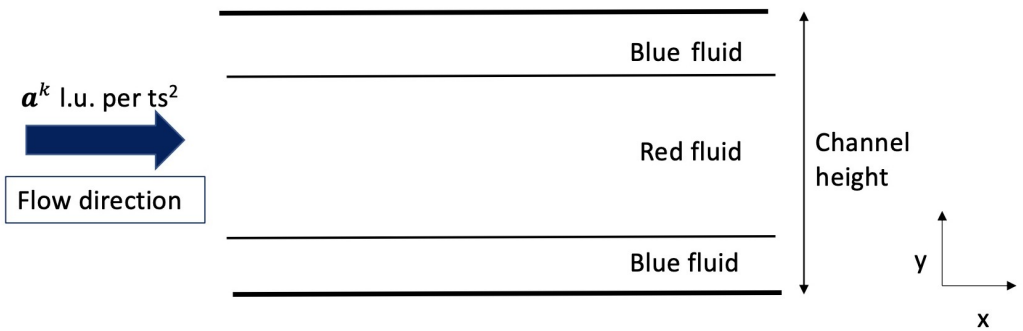


Figure 5. Two-phase Poiseuille flow simulation setup [88].

We performed the test using a domain of 10 nodes in the x-direction and 120 nodes in the y-direction. As shown in Figure 5, the top and bottom walls are solid boundaries at

which a no-slip boundary condition (half-way bounceback) is imposed. Periodic boundary conditions are applied on the left and right boundaries, which is why only 10 nodes are required in the x-direction. We performed the test for both the Shan and Chen [20] forcing scheme, which we now call the "feq-accel" method following Bawazeer et al. [96], and the Luo [97,98] forcing scheme, which we now call the "source term" method. For both methods, we tested viscosity ratios (ν^r/ν^b) of 0.25 and 4.0; thus, a total of four simulations were performed. For all tests, the interfacial tension $\sigma = 0.010$, the applied acceleration on the red fluid was $a^r = 1.5 \times 10^{-8}$ l.u./(t.s.)^2, the interface thickness parameter $\beta = 0.9$, the fluid densities $\rho^r = \rho^b = 1.0$, and simulations were run for at least 80,000 time steps to ensure that steady state was reached.

The results of the simulations are shown in Figure 6. The figure shows the velocity profiles of the fluids between the two parallel plates. At the top and bottom of the domain (i.e., at the two plates), the fluid velocity is 0 because of the no-slip boundary condition. The velocity reaches a maximum in the middle of the channel. There are two key findings as illustrated in Figure 6: first, there is no noticeable difference between the the left two panels [20] and the right two panels [97,98] of the figure, indicating that the two forcing schemes yield indistinguishable results for this test problem; second, both methods yield results that agree nearly perfectly with the analytical solution, indicating that either method is acceptable for the conditions tested.

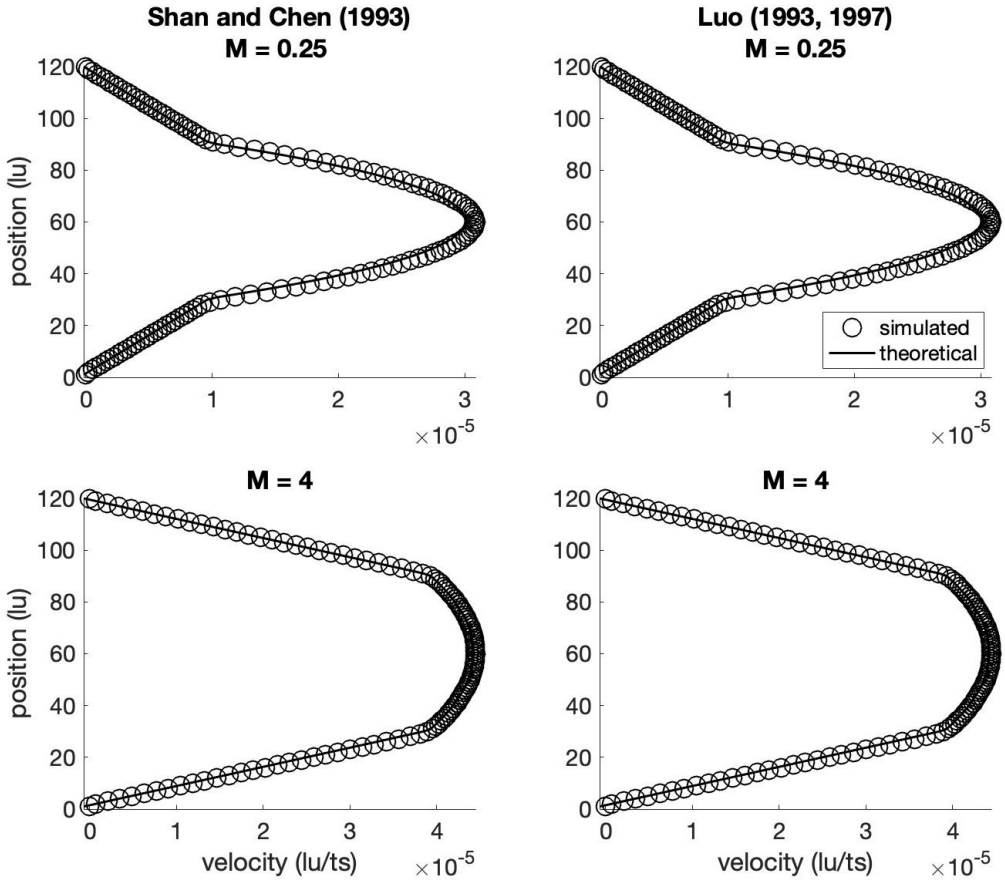


Figure 6. Comparison of layered Poiseuille flow results from the feq-accel forcing scheme of Shan and Chen [20] and the source term forcing scheme of Luo [97,98].

The equivalence of the two forcing schemes in Figure 6 is not surprising for the particular test problem considered. An examination of Equations (10), (14), (16) and (33)–(36) shows that the two methods are mathematically equivalent for cases in which the last two terms of the f^{eq} distribution functions can be ignored. This is the case for

grid-aligned Poiseuille flow; see Ginzburg et al. [91]. Also, the relationship and agreement between these two methods for the case of single-fluid flow have been elucidated by Huang et al. [99].

Based on the agreement between the two methods, it is not the case that one method is generally "better" than the other for multi-phase flow simulations that use the BGK collision operator. Nevertheless, we find it useful to present and compare the two proposed approaches here, for a few reasons. First, some previous papers have indicated that using gravitational acceleration in multi-phase simulations is not preferred (e.g., [19,75]), which points toward the use of an "external" forcing scheme. Second, Silva [100] demonstrated that if the applied force is spatially varying, including the acceleration term in the f^{eq} expression leads to a discrepancy between the CGM and the incompressible Navier–Stokes equations with an external body force under steady-state conditions; this also argues in favor of the source term forcing scheme. Finally, we found that when deploying more advanced CGM techniques, such as multiple relaxation times instead of the BGK collision scheme, one or the other of these two proposed forcing schemes can sometimes be easier to implement or code in practice. It is therefore useful to the reader to be familiar with both approaches in case a particular problem favors one over the other from a practical standpoint.

7. Challenges of Simulating Multi-Phase Flow in CGM

A multi-phase simulation in the CGM must be free from artificial fluid effects, which can arise near fluid–fluid and fluid–solid boundaries. For example, a good multi-phase simulation involves distinct phases separated by a thin interface, maintains fluid continuity between the phases, provides minimal fictitious or spurious velocities, and provides minimal fluid distortion. Therefore, a main challenge in the CGM is accurately depicting fluid flow while eliminating any artificial fluid effects. There is no numerical multi-phase flow model that is 100% accurate and free from error. However, by understanding the principles and theoretical foundations of the CGM, such errors can be minimized, and the method can provide a reasonable depiction of fluid flow processes.

One common error in the CGM is the presence of spurious velocities, i.e., the presence of fictitious fluid velocities where the fluids are supposed to be stationary. Such spurious velocities can arise near the interface of a bubble or droplet, as in Section 5. In that section, we discussed how the calculation of the color gradient can influence the droplet shape, as shown in Figure 4. The calculation of the color gradient can also affect the magnitude of spurious velocities. The algorithms discussed heretofore in this paper have made efforts to reduce or eliminate those fictitious velocities, but they usually do not disappear completely [68,70,94]. Specifically, for the case shown in Figure 4, the maximum magnitude of these spurious velocities is around 2.3×10^{-4} l.u./t.s. when using the "original" color gradient and around 7.1×10^{-6} l.u./t.s. when using the isotropic color gradient; see Figure 7. By using a higher-order gradient discretization, artificial velocities can be reduced, which also accompanies a reduction in the distortion in fluid shape (as seen in Figure 4).

In a CGM simulation, the major challenges are minimizing the presence of spurious velocities, ensuring pressure continuity between phases, minimizing the interface thickness, and minimizing distortion in fluid shape (e.g., droplet shape). This paper focused on the BGK-CGM formulation, and errors were considered in the context of common CGM code-validation tests that can be implemented with the BGK collision algorithm—which means that we considered relatively "easy" test conditions. In cases where simulations must account for fluid wettability, a higher viscosity ratio, and/or a higher density ratio, errors are likely to become more prominent. In future papers in this

series, we will review efforts that have been made to minimize errors in more sophisticated contexts [87,101,102].

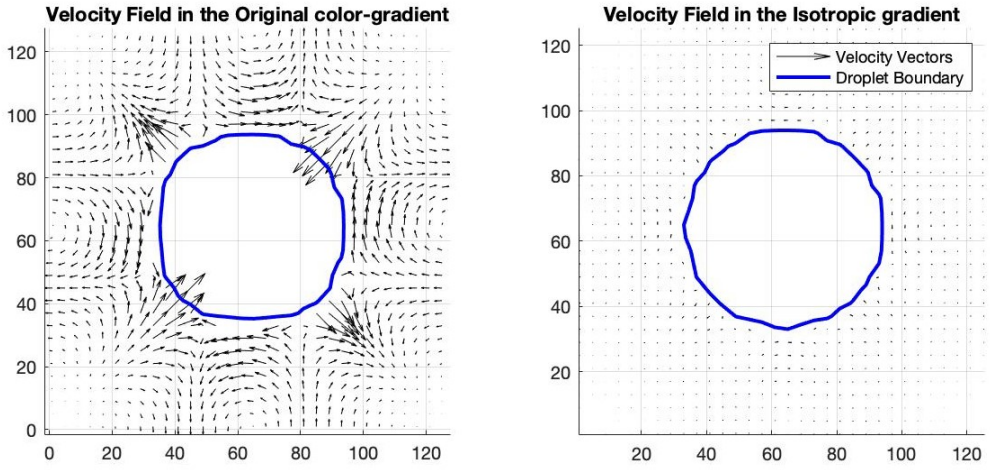


Figure 7. Spurious velocities generated by original and isotropic color gradient algorithms in a bubble test. Using the isotropic gradient reduces the magnitude of spurious velocities by a factor of about 30.

Another challenge in executing the LBM simulations—not limited to the color gradient method—is appropriately determining the relationship between "LBM variables" and "real-world variables". This is achieved by using appropriate unit conversions for length, time, and mass. For instance, what real-world length scale is represented by one lattice unit, and what real-world time scale is represented by one time step? The concepts of scaling the LBM units have been considered previously by Pan et al. [5], Krüger et al. [56], and Baakeem et al. [103], among others. A key point is that, in many instances, particularly if we want to model the flow of low-viscosity fluids, the LBM lattice unit and time step must correspond to very small values of real-world length and time—millimeters and milliseconds, or perhaps even smaller. Thus, a simulation that takes hours to run might only simulate, say, the fluid flow through a 10 cm $\times$ 10 cm square for a period of a few minutes. Therefore, modeling fluid flow through large domains (a scale of meters) over time scales of hours or days can be very computationally expensive. A way to address the challenge of computational expense is through the parallelization of the LBM algorithms, which has been considered elsewhere [12,15,18,19,104].

8. Summary and Conclusions

Over the past 30 years, the lattice Boltzmann method (LBM) has become increasingly popular for modeling multi-phase fluid flow in porous media, and one popular variant of the LBM is the color gradient method. However, along with the rising popularity of the color gradient method has come a proliferation of different proposed algorithms and strategies for implementing various aspects of the method. Selecting the "best" algorithms and subroutines is daunting, especially for relatively new users of the method. Therefore, we aim to critically review the capabilities and implementation of the CGM, particularly for modeling pore-scale multi-phase flow in porous media, through a series of papers, of which this is the first.

In this paper, we provided a brief introduction to the LBM and CGM that enables and facilitates consideration of the subsequent topics. Following that introduction, we reviewed three specific aspects of color gradient modeling: different methods for modeling the flow of fluids of moderately different viscosities, different methods for calculating the

color gradient, and different methods for modeling external forces or accelerations acting upon the flowing fluids. Specific findings from this paper include the following:

1. Spatial interpolation of the collision relaxation times, τ, enables accurate modeling of the flow of fluids of different viscosities as long as the viscosity contrast is moderate;
2. Of the methods considered for τ interpolation, that of Liu et al. [88] is accurate and easy to implement;
3. Using the "original" color gradient calculation of Rothman and Keller [24], Gunstensen et al. [25], and Grunau et al. [60] is sufficient in some cases, but the "isotropic" color gradient calculation of Leclaire et al. [70] improves upon the original method without any significant increase in model complexity;
4. External accelerations acting on the flowing fluids can be modeled using a variety of forcing schemes, but for simple problems, there is no significant difference between the "feq-accel" forcing scheme of Shan and Chen [20] and the "source term" forcing scheme of Luo [97,98].

Understanding these aspects of the color gradient method is necessary and sufficient to allow the user to perform two common "benchmark" tests of lattice Boltzmann modeling, namely bubble tests and layered Poiseuille flow.

Future papers in the series will build upon these topics, considering phenomena or conditions such as fluids of widely differing viscosity and/or density, how to introduce the "wettability" of solid surfaces and specify a desired contact angle, how to simulate fluid flow in a domain with an open boundary condition (e.g., fluid flushing out of a porous medium), and how to extend the CGM from two spatial dimensions to three. By systematically reviewing these key aspects and features, this series of papers will extend our collective ability to apply the CGM to a variety of important problems in geosciences and engineering.

Author Contributions: Conceptualization, F.Z. and J.A.C.; methodology, F.Z. and J.A.C.; software, F.Z. and J.A.C.; validation, F.Z. and J.A.C.; formal analysis, F.Z. and J.A.C.; investigation, F.Z.; resources, J.A.C.; data curation, F.Z. and J.A.C.; writing—original draft preparation, F.Z.; writing—review and editing, F.Z. and J.A.C.; visualization, F.Z. and J.A.C.; supervision, J.A.C.; project administration, F.Z. and J.A.C.; funding acquisition, F.Z. and J.A.C. All authors have read and agreed to the published version of the manuscript.

Funding: This paper is based on the funding and support from the Foreign Fulbright PhD Fellowship (awarded to F.Z.) and from the Commonwealth Scientific and Industrial Research Organization (CSIRO, Australia, support of J.A.C. during 2013–2014). Any opinions, findings, conclusions, or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the Fulbright Association or of CSIRO.

Data Availability Statement: Code development data related to the CGM will be available upon request from the authors.

Acknowledgments: Some of the computations for this work were performed on the Research Computing Resources of the University of South Florida (USF). The authors thank Ethan Yan for his assistance with code development and Amy Stuart for her assistance with deploying the codes on USF's Research Computing Resources. The authors also thank Sebastian Leclaire for sharing his Laplace bubble test MATLAB Version R2020b codes, which helped us develop our own Laplace bubble test. Lastly, the authors thank Al Valocchi for his words of encouragement at various stages during the coding process.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

BGK	Bhatnagar, Gross, Krook
CGM	Color gradient method
LBM	Lattice Boltzmann method
MRT	Multiple relaxation time

References

1. Blunt, M.J. *Multiphase Flow in Permeable Media: A Pore-Scale Perspective*; Cambridge University Press: Cambridge, UK, 2017.
2. Meakin, P.; Tartakovsky, A.M. Modeling and simulation of pore-scale multiphase fluid flow and reactive transport in fractured and porous media. *Rev. Geophys.* **2009**, *47*, RG3002. [CrossRef]
3. Zhao, B.; MacMinn, C.W.; Primkulov, B.K.; Chen, Y.; Valocchi, A.J.; Zhao, J.; Kang, Q.; Bruning, K.; McClure, J.E.; Miller, C.T.; et al. Comprehensive comparison of pore-scale models for multiphase flow in porous media. *Proc. Natl. Acad. Sci. USA* **2019**, *116*, 13799–13806. [CrossRef] [PubMed]
4. Guo, Z.; Zhao, T.S. Lattice Boltzmann model for incompressible flows through porous media. *Phys. Rev. E* **2002**, *66*, 036304. [CrossRef] [PubMed]
5. Pan, C.; Hilpert, M.; Miller, C.T. Lattice-Boltzmann simulation of two-phase flow in porous media. *Water Resour. Res.* **2004**, *40*, W01501. [CrossRef]
6. Sukop, M.C.; Thorne, D.T.J. *Lattice Boltzmann Modeling: An Introduction for Scientists and Engineers*; Springer: New York, NY, USA, 2006.
7. Porter, M.L.; Schaap, M.G.; Wildenschild, D. Lattice-Boltzmann simulations of the capillary pressure–saturation–interfacial area relationship for porous media. *Adv. Water Resour.* **2009**, *32*, 1632–1640. [CrossRef]
8. Boek, E.S.; Venturolli, M. Lattice-Boltzmann studies of fluid flow in porous media with realistic rock geometries. *Comput. Math. Appl.* **2010**, *59*, 2305–2314. [CrossRef]
9. Ho, M.; Tucny, J.M.; Ammar, S.; Leclaire, S.; Reggio, M.; Trepanier, J.Y. Lattice Boltzmann model for rarefied gaseous mixture flows in three-dimensional porous media including Knudsen diffusion. *Fluids* **2024**, *9*, 237. [CrossRef]
10. Chen, L.; Kang, Q.; Mu, Y.; He, Y.L.; Tao, W.Q. A critical review of the pseudopotential multiphase lattice Boltzmann model: methods and applications. *Int. J. Heat Mass Transf.* **2014**, *76*, 210–236. [CrossRef]
11. Atykhan, M.; Kabdenova, B.; Monaco, E.; Rojas-Solorzano, L. Modeling immiscible fluid displacement in a porous medium using lattice Boltzmann method. *Fluids* **2021**, *6*, 89. [CrossRef]
12. Kandhai, D.; Koponen, A.; Hoekstra, A.G.; Kataja, M.; Timonen, J.; Sloot, P.M. Lattice Boltzmann hydrodynamics on parallel systems. *Comput. Phys. Commun.* **1998**, *111*, 14–26. [CrossRef]
13. Wang, M.; Pan, N. Prediction of effective physical properties of complex multiphase materials. *Mater. Sci. Eng. R Rep.* **2008**, *63*, 1–30. [CrossRef]
14. Aidun, C.K.; Clausen, J.R. Lattice-Boltzmann method for complex flows. *Annu. Rev. Fluid Mech.* **2010**, *42*, 439–472. [CrossRef]
15. Tölke, J. Implementation of a lattice Boltzmann kernel using the Compute Unified Device Architecture developed by nVIDIA. *Comput. Vis. Sci.* **2010**, *13*, 29–39. [CrossRef]
16. Liu, H.; Kang, Q.; Leonardi, C.R.; Schmieschek, S.; Narváez, A.; Jones, B.D.; Williams, J.; Valocchi, A.J.; Harting, J. Multiphase lattice Boltzmann simulations for porous media applications. *Comput. Geosci.* **2016**, *20*, 777–805. [CrossRef]
17. McClure, J.E.; Li, Z.; Berrill, M.A.; Ramstad, T. The LBPM software package for simulating multiphase flow on digital images of porous rocks. *Comput. Geosci.* **2021**, *25*, 871–895. [CrossRef]
18. Yang, J.; Xu, Y.; Yang, L. Taichi LBM3D: A single-phase and multiphase lattice Boltzmann solver on cross-platform multicore CPU/GPUs. *Fluids* **2022**, *7*, 270. [CrossRef]
19. Yang, G.; Chen, S.; Wang, M. Implementation of a direct-addressing based lattice Boltzmann GPU solver for multiphase flow in porous media. *Comput. Phys. Commun.* **2023**, *65*, 108828. [CrossRef]
20. Shan, X.; Chen, H. Lattice Boltzmann model for simulating flows with multiple phases and components. *Phys. Rev. E* **1993**, *47*, 1815–1819. [CrossRef]
21. Swift, M.R.; Orlandini, E.; Osborn, W.R.; Yeomans, J.M. Lattice Boltzmann simulations of liquid-gas and binary fluid systems. *Phys. Rev. E* **1996**, *54*, 5041–5052. [CrossRef]
22. He, X.; Chen, S.; Zhang, R. A lattice Boltzmann scheme for incompressible multiphase flow and its application in simulation of Rayleigh-Taylor instability. *J. Comput. Phys.* **1999**, *152*, 642–663. [CrossRef]
23. Lee, T.; Liu, L. Lattice Boltzmann simulations of micron-scale drop impact on dry surfaces. *J. Comput. Phys.* **2010**, *229*, 8045–8062. [CrossRef]

24. Rothman, D.H.; Keller, J.M. Immiscible Cellular-Automaton Fluids. *J. Stat. Phys.* **1988**, *52*, 1119–1127. [CrossRef]
25. Gunstensen, A.K.; Rothman, D.H.; Zaleski, S.; Zanetti, G. Lattice Boltzmann model of immiscible fluids. *Phys. Rev. A* **1991**, *43*, 4320–4327. [CrossRef] [PubMed]
26. Huang, H.; Sukop, M.C.; Lu, X.Y. *Multiphase Lattice Boltzmann Methods: Theory and Application*; John Wiley & Sons: Chichester, UK, 2015.
27. Latva-Kokko, M.; Rothman, D. Diffusion properties of gradient-based lattice Boltzmann models of immiscible fluids. *Phys. Rev. E* **2005**, *71*, 056702. [CrossRef] [PubMed]
28. McClure, J.E.; Berill, M.A.; Gray, W.G.; Miller, C.T. Tracking interface and common curve dynamics for two-fluid flow in porous media. *J. Fluid Mech.* **2016**, *796*, 211–232. [CrossRef]
29. Ahrenholz, B.; Tölke, J.; Lehmann, P.; Peters, A.; Kaestner, A.; Krafczyk, M.; Durner, W. Prediction of capillary hysteresis in a porous material using lattice-Boltzmann methods and comparison to experimental data and a morphological pore network model. *Adv. Water Resour.* **2008**, *31*, 1151–1173. [CrossRef]
30. Chen, Y.; Li, Y.; Valocchi, A.J.; Christensen, K.T. Lattice Boltzmann simulations of liquid CO_2 displacing water in a 2D heterogenous micromodel at reservoir pressure conditions. *J. Contam. Hydrol.* **2018**, *212*, 14–27. [CrossRef]
31. Sarkezi-Selsky, P.; Schmies, H.; Kube, A.; Latz, A.; Jahnke, T. Lattice Boltzmann simulation of liquid water transport in gas diffusion layers of proton exhange membrane fuel cells: Parametric studies on capillary hysteresis. *J. Power Sources* **2022**, *535*, 231381. [CrossRef]
32. Fu, X.; Sun, J. Improved three-dimensional multiple-relaxation-time color-gradient lattice Boltzmann finite-difference model for thermocapillary flows. *Phys. Fluids* **2023**, *35*, 072105.
33. Zahid, F.; Cunningham, J.A. Effect of Grain-Size Distribution on Temporal Evolution of Interfacial Area during Two-phase Flow in Porous Media. *Transp. Porous Med.* **2022**, *144*, 283–300. [CrossRef]
34. Liu, H.; Sun, S.; Wu, R.; Wei, B.; Hou, J. Pore-scale modeling of spontaneous imbibition in porous media using the lattice Boltzmann method. *Water Resour. Res.* **2021**, *57*, 2020WR029219. [CrossRef]
35. Sheikholeslam-Noori, S.M.; Taeibi-Rahni, M.; Shams-Taleghani, S.A. Multiple-relaxation time color-gradient lattice Boltzmann model for simulating contact angle in two phase flows with high density ratios. *Eur. Phys. J. Plus* **2019**, *134*, 399. [CrossRef]
36. Zakirov, T.R.; Zhuchkova1, O.S.; Khramchenkov, M.G. Mathematical Model for Dynamic Adsorption with Immiscible Multiphase Flows in Three-dimensional Porous Media. *Lobachevskii J. Math.* **2024**, *45*, 888–898. [CrossRef]
37. Zakirov, T.R.; Khayuzkin, A.S.; Khramchenkov, M.G.; Galeev, A.A.; Kosterina, E.A. Effect of gravity number on soil contamination during dynamic adsorption of heavy metal ions: A lattice Boltzmann study. *Int. Commun. Heat Mass Transf.* **2024**, *158*, 107852. [CrossRef]
38. Li, Q.; Yang, G.; Huang, Y.; Lu, X.; Min, J.; Wang, M. Lattice Boltzmann method for particulate multiphase flow system. *Int. J. Mech. Sci.* **2024**, *273*, 109217. [CrossRef]
39. Datadien, K.P.N.; Staso, G.D.; Diddens, C.; Wijshoff, H.M.A.; Toschi, F. Comparison of Lattice Boltzmann, Finite Element and Volume of Fluid Multicomponent Methods for Microfluidic Flow Problems and the Jetting of Microdroplets. *Commun. Comput. Phys.* **2023**, *33*, 912–936. [CrossRef]
40. Huang, H.; Wang, L.; Lu, X.Y. Evaluation of three lattice Boltzmann models for multiphase flows in porous media. *Comput. Math. Appl.* **2011**, *61*, 3606–3617. [CrossRef]
41. Yang, J.; Boek, E.S. A comparison study of multi-component lattice Boltzmann models for flow in porous media applications. *Comput. Math. Appl.* **2013**, *65*, 882–890. [CrossRef]
42. Leclaire, S.; Parmigiani, A.; Chopard, B.; Latt, J. Three-dimensional lattice Boltzmann method benchmarks between color-gradient and pseudo-potential immiscible multi-component models. *Int. J. Mod. Phys. C* **2017**, *28*, 1750085. [CrossRef]
43. Guo, Z.; Shu, C. *Lattice Boltzmann Method and Its Applications in Engineering*; World Scientific Publishing Co.: Hackensack, NJ, USA, 2013.
44. Ba, Y.; Liu, H.; Li, Q.; Kang, Q.; Sun, J. Multiple-relaxation-time color-gradient lattice Boltzmann model for simulating two-phase flows with high density ratio. *Phys. Rev. E.* **2016**, *94*, 023310. [CrossRef]
45. Akai, T.; Bijeljic, B.; Blunt, M.J. Wetting boundary condition for the color-gradient lattice Boltzmann method: Validation with analytical and experimental data. *Adv. Water Resour.* **2018**, *116*, 56–66. [CrossRef]
46. Wen, Z.X.; Li, Q.; Yu, Y.; Luo, K.H. Improved three-dimensional color-gradient lattice Boltzmann model for immiscible two-phase flows. *Phys. Rev. E* **2019**, *100*, 023301. [CrossRef] [PubMed]
47. Frisch, U.; Hasslacher, B.; Pomeau, Y. Lattice-gas automata for the Navier-Stokes equation. *Phys. Rev. Lett.* **1986**, *56*, 1505–1508. [CrossRef] [PubMed]
48. McNamara, G.R.; Zanetti, G. Use of the Boltzmann equation to simulate lattice-gas automata. *Phys. Rev. Lett.* **1988**, *61*, 2332–2335. [CrossRef]
49. Higuera, F.J.; Succi, S.; Benzi, R. Lattice gas dynamics with enhanced collisions. *Europhys. Lett.* **1989**, *9*, 345–349. [CrossRef]
50. Higuera, F.J.; Jiménez, J. Boltzmann approach to lattice gas simulations. *Europhys. Lett.* **1989**, *9*, 663–668. [CrossRef]

51. Wolfram, S. Cellular automaton fluids, 1: Basic Theory. *J. Stat. Phys.* **1986**, *45*, 471–526. [CrossRef]
52. Bhatnagar, P.L.; Gross, E.P.; Krook, M. A model for collision processes in gases. I. Small amplitude processes in charged and neutral one-component systems. *Phys. Rev. E.* **1954**, *94*, 511–525. [CrossRef]
53. Qian, Y.H.; d'Humières, D.; Lallemand, P. Lattice BGK models for Navier-Stokes equation. *Europhys. Lett.* **1992**, *17*, 479–484. [CrossRef]
54. Lallemand, P.; Luo, L.S. Theory of the lattice Boltzmann method: Dispersion, dissipitation, isotropy, Galilean invariance, and stability. *Phys. Rev. E* **2000**, *61*, 6546. [CrossRef]
55. Reis, T.; Phillips, T.N. Lattice Boltzmann Model for Simulating Immiscible Two-Phase Flows. *J. Phys. A Math.* **2007**, *40*, 4033–4053. [CrossRef]
56. Krüger, T.; Kusumaatmaja, H.; Kuzmin, A.; Shardt, O.; Silva, G.; Viggen, E.M. *The Lattice Boltzmann Method: Principles and Practice*; Springer International Publishing: Cham, Switzerland, 2017.
57. Mohamad, A.A. *Lattice Boltzmann Methods: Fundamentals and Engineering Applications with Computer Codes*, 2nd ed.; Springer: London, UK, 2019.
58. Chai, Z.; Huang, C.; Shi, B.; Guo, Z. A comparative study on the lattice Boltzmann models for predicting effective diffusivity of porous media. *Int. J. Heat Mass Trans.* **2016**, *98*, 687–696. [CrossRef]
59. Zhu, W.; Kan, A.; Chen, Z.; Zhang, Q.; Zhang, J. A modified lattice Boltzmann method for predicting the effective thermal conductivity of open-cell foam materials. *Int. J. Heat Mass Trans.* **2022**, *133*, 105957. [CrossRef]
60. Grunau, D.; Chen, S.; Eggert, K. A lattice Boltzmann model for multiphase fluid flows. *Phys. Fluids A* **1993**, *5*, 2557–2562. [CrossRef]
61. Tölke, J.; Krafczyk, M.; Schulz, M.; Rank, E. Lattice Boltzmann simulations of binary fluid flow through porous media. *Philos. Trans. A Math. Phys. Eng. Sci.* **2002**, *360*, 535–545. [CrossRef]
62. Liu, H.; Valocchi, A.J.; Kang, Q. Three-dimensional lattice Boltzmann model for immiscible two-phase flow simulations. *Phys. Rev. E* **2012**, *85*, 046309. [CrossRef]
63. Ginzburg, I.; Steiner, K. A free-surface lattice Boltzmann method for modelling the filling of expanding cavities by Bingham fluids. *Philos. Trans. R. Soc. Lond. A* **2002**, *360*, 453–466. [CrossRef]
64. Ginzburg, I.; Steiner, K. Lattice Boltzmann model for free-surface flow and its application to filling process in casting. *J. Comput. Phys.* **2003**, *185*, 61–99. [CrossRef]
65. Kehrwald, D. Numerical Analysis of Immiscible Lattice BGK. Ph.D. Thesis, Fraunhofer Institut für Techno- und Wirtschaftsmathematik, Kaiserslautern, Germany, 2002.
66. Ginzburg, I. Lattice Boltzmann modeling with discontinuous collision components: Hydrodynamic and advection-diffusion equations. *J. Stat. Phys.* **2007**, *126*, 157–206. [CrossRef]
67. Leclaire, S.; Reggio, M.; Trépanier, J.Y. Numerical evaluation of two recoloring operators for an immiscible two-phase flow lattice Boltzmann model. *Appl. Math. Model.* **2012**, *36*, 2237–2252. [CrossRef]
68. Huang, H.; Huang, J.J.; Lu, X.Y. Study of immiscible displacements in porous media using color-gradient based multiphase lattice Boltzmann method. *Comput. Fluids* **2014**, *93*, 164–172. [CrossRef]
69. Leclaire, S.; Parmigiani, A.; Malaspinas, O.; Chopard, B.; Latt, J. Generalized three-dimensional lattice Boltzmann color-gradient method for immiscible two-phase pore-scale imbibition and drainage in porous media. *Phys. Rev. E* **2017**, *95*, 033306. [CrossRef] [PubMed]
70. Leclaire, S.; Reggio, M.; Trépanier, J.Y. Isotropic color gradient for simulating very high density ratios with a two-phase flow lattice Boltzmann model. *Comput. Fluids* **2011**, *48*, 98–112. [CrossRef]
71. Xu, Z.; Liu, H.; Valocchi, A.J. Lattice Boltzmann simulation of immiscible two-phase flow capillary valve effect in porous media. *Water Resour. Res.* **2017**, *53*, 3770–3790. [CrossRef]
72. Lishchuk, S.V.; Care, C.M.; Halliday, I. Lattice Boltzmann algorithm for surface tension with greatly reduced microcurrents. *Phys. Rev. E* **2003**, *67*, 036701. [CrossRef]
73. Xu, M.; Liu, H. Prediction of immiscible two-phase flow properties in a two-dimensional Berea sandstone using the pore-scale lattice Boltzmann simulation. *Eur. Phys. J. E* **2018**, *41*, 124. [CrossRef]
74. Chen, Y.; Valocchi, A.J.; Kang, Q.; Visvanathan, H.S. Inertial effects during the process of supercritical CO_2 displacing brine in a sandstone: Lattice Boltzmann simulations based on the continuum-surface-force and geometrical wetting models. *Water Resour. Res.* **2019**, *55*, 144–165. [CrossRef]
75. Huang, J.; Xiao, F.; Yin, X. Lattice Boltzmann simulation of pressure-driven two-phase flows in capillary tube and porous medium. *Comput. Fluids* **2017**, *155*, 134–145. [CrossRef]
76. Ginzbourg, I.; Adler, P.M. Surface tension models with different viscosities. *Transp. Porous Media* **1995**, *20*, 37–76. [CrossRef]
77. D'Ortona, U.; Salin, D.; Cieplak, M.; Rybka, R.B. Two-color nonlinear Boltzmann cellular automata: Surface tension and wetting. *Phys. Rev. E* **1995**, *51*, 3718–3728. [CrossRef]

78. Halliday, I.; Hollis, A.P.; Care, C.M. Lattice Boltzmann algorithm for continuum multicomponent flow. *Phys. Rev. E* **2007**, *76*, 026078. [CrossRef] [PubMed]
79. Lishchuk, S.V.; Halliday, I.; Care, C.M. Multicomponent lattice Boltzmann method for fluids with density contrast. *Phys. Rev. E* **2008**, *77*, 036702. [CrossRef] [PubMed]
80. Liu, H.; Wu, L.; Ba, Y.; Xi, G.; Zhang, Y. A lattice Boltzmann method for axisymmetric multicomponent flows with high viscosity ratio. *J. Comput. Phys.* **2016**, *27*, 052103. [CrossRef]
81. Leclaire, S.; Pellerin, S.; Reggio, M.; Trépanier, J.Y. An approach to control the spurious currents in a multiphase lattice Boltzmann method and to improve the implementation of initial condition. *Int. J. Numer. Methods Fluids* **2015**, *77*, 732–746. [CrossRef]
82. Chen, J.D.; Wilkinson, D. Pore-scale viscous fingering in porous media. *Phys. Rev. Lett.* **1985**, *55*, 1892–1895. [CrossRef]
83. Homsy, G. Viscous fingering in porous media. *Annu. Rev. Fluid Mech.* **1987**, *19*, 271–311. [CrossRef]
84. Dong, B.; Yan, Y.Y.; Li, W.; Song, Y. Lattice Boltzmann simulation of viscous fingering phenomenon of immiscible fluids displacement in a channel. *Comput Fluids* **2010**, *39*, 768–779. [CrossRef]
85. Liu, H.; Zhang, Y.; Valocchi, A.J. Lattice Boltzmann simulation of immiscible fluid displacement in porous media: Homogenous versus heterogeneous pore network. *Phys. Fluids* **2015**, *27*, 052103. [CrossRef]
86. Bakhshian, S.; Hosseini, S.A.; Shokri, N. Pore-scale characteristics of multiphase flow in heterogenous porous media using the lattice Boltzmann method. *Sci. Rep.* **2019**, *9*, 3377. [CrossRef]
87. Rannou, G. Lattice Boltzmann Method and Immiscible Two-Phase Flow. Ph.D. Thesis, Georgia Institute of Technology, Atlanta, GA, USA, 2008.
88. Liu, H.; Valocchi, A.J.; Werth, C.; Kang, Q.; Oostrom, M. Pore-scale simulation of liquid CO_2 displacement of water using a two-phase lattice Boltzmann model. *Adv. Water Resour.* **2014**, *73*, 144–158. [CrossRef]
89. Zu, Y.Q.; He, S. Phase-field based lattice Boltzmann model for incompressible binary fluid systems with density and viscosity contrasts. *Phys. Rev. E* **2013**, *87*, 043301. [CrossRef] [PubMed]
90. Huang, J. Pore-Scale Simulation of Multiphase Flow Using Lattice Boltzmann Method: Developments and Applications. Ph.D. Thesis, Colorado School of Mines, Golden, CO, USA, 2017.
91. Ginzburg, I.; Verhaege, F.; d'Humières, D. Two-Relaxation-Time Lattice Boltzmann Scheme: About Parametrization, Velocity, Pressure and Mixed Boundary Conditions. *Commun. Comput. Phys.* **2008**, *3*, 427–478.
92. d'Humières, D.; Ginzburg, I. Viscosity independent numerical errors for lattice Boltzmann models: From recurrence equations to 'magic' collision numbers. *Comput. Math. Appl.* **2009**, *58*, 823–840. [CrossRef]
93. Lafarge, T.; Boivin, P.; Odier, N.; Cuenot, B. Improved color-gradient method for lattice Boltzmann modeling of two-phase flows. *Phys. Fluids* **2021**, *33*, 082110. [CrossRef]
94. Mora, P.; Morra, G.; Yuen, D.A. Optimal surface-tension isotropy in the Rothman-Keller color-gradient lattice Boltzmann method for multiphase flow. *Phys. Rev. E* **2021**, *103*, 033302. [CrossRef]
95. Zou, Q.; He, X. On pressure and velocity boundary conditions for the lattice Boltzmann BGK model. *Phys. Fluids* **1997**, *9*, 1591–1598. [CrossRef]
96. Bawazeer, S.H.; Baakeem, S.S.; Mohamad, A.A. A critical review of forcing schemes in lattice Boltzmann method: 1993–2019. *Arch. Comput. Methods Eng.* **2021**, *28*, 4405–4423. [CrossRef]
97. Luo, L.S. Lattice-Gas Automata and Lattice Boltzmann Equations for Two-Dimensional Hydrodynamics. Ph.D. Thesis, Georgia Institute of Technology, Atlanta, GA, USA, 1993.
98. Luo, L.S. Analytic solutions of linearized lattice Boltzmann equation for simple flows. *J. Stat. Phys.* **1997**, *88*, 913–926. [CrossRef]
99. Huang, H.; Krafczyk, M.; Lu, X. Forcing term in single-phase and Shan-Chen-type multiphase lattice Boltzmann models. *Phys. Rev. E* **2011**, *84*, 046710. [CrossRef]
100. Silva, G. Discrete effects on the forcing term for the lattice Boltzmann modeling of steady hydrodynamics. *Comput. Fluids* **2020**, *203*, 104537. [CrossRef]
101. Huang, H.; Huang, J.J.; Lu, X.; Sukop, M.C. On simulations of high-density ratio flows using color-gradient multi-phase lattice Boltzmann models. *Int. J. Mod. Phys. C* **2013**, *24*, 1350021. [CrossRef]
102. Leclaire, S.; Pellerin, N.; Reggio, M.; Trépanier, J.Y. A multiphase lattice Boltzmann method for simulating immiscible liquid-liquid interface dynamics. *Appl. Math. Model.* **2016**, *40*, 6376–6394. [CrossRef]
103. Baakeem, S.S.; Bawazeer, S.H.; Mohamad, A.A. A novel approach of unit conversion in the lattice Boltzmann method. *Appl. Sci.* **2021**, *11*, 6386. [CrossRef]
104. Namvar, M.; Leclaire, S. Simple lattice Boltzmann model with low memory usage. *J. Comput. Sci.* **2022**, *62*, 101723. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.