Computers and Mathematics with Applications 58 (2009) 950–964

Contents lists available at ScienceDirect

Computers and Mathematics with Applications

journal homepage: www.elsevier.com/locate/camwa

A combined lattice BGK/level set method for immiscible two-phase flows☆

Jürgen Becker[a], Michael Junk[b], Dirk Kehrwald[a], Guido Thömmes[a,*], Zhaoxia Yang[b]

[a] Fraunhofer Institut für Techno- und Wirtschaftsmathematik, Fraunhofer Platz 1, 67663 Kaiserslautern, Germany

[b] Fachbereich Mathematik und Statistik, Universität Konstanz, Universitätsstraße 10, 78457 Konstanz, Germany

ARTICLE INFO

Keywords:
Lattice Boltzmann method
Level set method
Two-phase flow

ABSTRACT

We present a lattice Boltzmann method (LBM) for simulating immiscible multi-phase flows which is based on a coupling of LBM with the level set method. The computation of immiscible flows using the LBM with BGK collision operator is done separately in each of the fluid domains and coupled at the interface by an appropriate boundary condition. In this way we preserve sharp interfaces between different fluid phases. We apply a new interface condition that represents the fluid mechanical jump conditions at the interface in the kinetic LBM framework. The level set method is applied to compute the evolution of the interface between fluids. Numerical results demonstrate the applicability of the method even in the presence of large viscosity and density ratios.

1. Introduction

In many applications, the flow of two immiscible fluids separated by a sharp interface is encountered, e.g. bubbles in chemical reactors, viscous fingering in the extraction of oil from a reservoir, liquid metal jets in metal forming processes, or droplets in a gas atmosphere. Owing to the practical importance of these processes, many different methods have been applied to simulate and predict the behavior of such flows. In this work we apply a new approach based on the lattice Boltzmann method (LBM) to address the problem of immiscible two-phase flows.

The lattice Boltzmann method is a kinetic-based scheme that is inspired by the microscopic picture of particles, which constitutes the fluid flow, and can be viewed as a specific discretization of the Boltzmann equation. During the past two decades, LBM has been developed as an alternative numerical scheme for solving the incompressible Navier–Stokes equations. The main advantages attributed to LBM are the ease of implementation, since the non-linear Navier–Stokes equations are replaced by the semi-linear Boltzmann equation, the simplicity in simulating domains with complex geometry for example in porous media, and the ease of parallelization, because only local operations are performed. An introduction to the method and its applications can be found in [1–4,7].

Since the study of immiscible fluids is an important model problem for free surface flows, the lattice Boltzmann method has also been extended to the simulation of immiscible two-phase flows. Essentially three generic LBM approaches have been developed over the years. The most important are the color gradient method [5,6,36], and the free energy method [8,9]. Moreover, Shan and Chen proposed a method based on interaction potentials [10].

The color gradient method [5–7] maintains two sets of LBM populations, one for each phase, and models phase separation and interface tension using a recoloring step. In each node, the algorithm attempts to separate the two phases as much as

☆ This work has been supported by DFG grants *JU 440/1-3*, *STE 871/1* and *KL 1105/9*.

* Corresponding author.
E-mail address: guido.thoemmes@itwm.fraunhofer.de (G. Thömmes).

0898-1221/$ – see front matter © 2009 Published by Elsevier Ltd
doi:10.1016/j.camwa.2009.02.005

possible by redistributing the two sets of populations. Interfaces are implicitly defined by the fluid fraction iso-surface where the contents of the two fluids are equal. In general the method is applicable only for small density and viscosity differences and in particular the recoloring step causes grid-dependent artifacts at the interface.

The free energy method was first introduced by Swift et al. [8,11]. It also relies on a second set of populations which describes the fluid fraction and is determined by the free energy of the system. Also this method cannot simulate large density ratios. However, there have been successful attempts to modify the method such that flows with very high density ratios can be computed [9].

A different approach was presented by Shan and Chen [10,12], who introduced the concept of interaction potentials. The method in principle models miscible fluids, and immiscible flows can only approximately be described and in some cases the method also suffers from mass losses.

Recently, a hybrid method combining LBM and the front-tracking method has also been proposed [13]. It combines the lattice Boltzmann algorithm with the front-tracking method and incorporates interface tension via curvature-dependent volume forces along the surface.

The method we propose also adopts the idea of hybrid methods, which combines LBM with standard numerical methods for tracking or capturing the interface. In our approach, the motion of fluid interfaces in immiscible two-phase flows is resolved using the interface capturing level set method, which represents the surface by the zero level set of a smooth signed distance function. The method has several interesting features. In the level set framework, topology changes are handled in a natural way. This may be important when simulating the coalescence or break-up of droplets observed in experiments. Furthermore, the level set method can be programmed very efficiently using special implementation techniques.

The new method is applied to simulate immiscible two-phase flows separated by sharp interfaces. We test channel flows with two fluid layers and the classical Young–Laplace law for the pressure in a spherical bubble to validate the scheme. Furthermore, we simulate viscous fingering experiments, and the evolution of rising bubbles. It turns out that the algorithm is capable of dealing with large density such as 1:1000 in the bubble test case described in Section 5.3.

The paper is organized as follows. In Section 2 we introduce the mathematical model for the two-phase flow problem, and describe the lattice Boltzmann method for the flow and the level set method for the surface evolution. The third section is devoted to the LBM interface conditions that couple different fluid phases at the interface. In the Sections 4 and 5 we present results of various test cases and numerical experiments.

2. The lattice Boltzmann method and the level set method

In this section we describe the two basic parts of our hybrid LBM approach. We model immiscible two-phase flows using the incompressible Navier–Stokes equations in each fluid phase. After the presentation of our model we discuss the lattice Boltzmann method used to solve the fluid dynamical equations and describe of the level set method for capturing the interface evolution.

2.1. Two-phase flow model

We consider the incompressible Navier–Stokes equations in a domain partitioned by subdomains Ω_i for each fluid phase:

$$\nabla \cdot u^{(i)} = 0, \tag{1a}$$

$$\partial_t u^{(i)} + (u^{(i)} \cdot \nabla) u^{(i)} = -\frac{1}{\varrho^{(i)}} \nabla p^{(i)} + \nu^{(i)} \Delta u^{(i)} + a^{(i)}, \tag{1b}$$

with appropriate boundary and initial conditions, for example

$$u^{(i)}(x,t) = u_b^{(i)}(x,t), \quad \text{on } \partial\Omega_i \setminus \Gamma, \tag{1c}$$

$$u^{(i)}(x,0) = u_{\text{ini}}^{(i)}(x), \quad \text{in } \Omega_i. \tag{1d}$$

In the equations, $u^{(i)}$ and $p^{(i)}$ are the velocity and pressure of the Navier–Stokes solution for fluids with kinematic viscosities $\nu^{(i)}$ and mass densities $\varrho^{(i)}$. Exterior forces, e.g. volume forces like gravitation, are modeled by the acceleration $a^{(i)}$. Furthermore, $u_b^{(i)}$ is the velocity of the solid boundary part and $u_{\text{ini}}^{(i)}$ is the initial velocity for subdomain Ω_i.

The interface between fluid phases is denoted Γ. At the interface, when surface tension is present, the following jump conditions for velocity and stress are imposed:

$$[u] = 0, \tag{2a}$$

$$[T] \cdot n = 2\sigma\kappa n. \tag{2b}$$

Here, brackets denote the jump of a quantity, say q, across the surface defined by $[q](x) = \lim_{\epsilon \to 0}(q(x+\epsilon n) - q(x-\epsilon n))$. The stress tensor is denoted $T^{(i)} = -p^{(i)} I + 2\mu^{(i)} S^{(i)}$, where $S_{kl}^{(i)} = \frac{1}{2}(\partial_k u_l^{(i)} + \partial_l u_k^{(i)})$ and $\mu^{(i)} = \varrho^{(i)} \nu^{(i)}$ is the dynamic viscosity. Moreover, σ is the surface tension coefficient and κ the mean curvature.

Since we consider two-phase flows with movable interfaces, the sets Ω_i and Γ depend on time. The algorithm we propose therefore contains three ingredients:

(1) a solver for the flow equations,
(2) a scheme for computing the motion of the surfaces, and
(3) the coupling of fluid flow and surface evolution.

2.2. The lattice Boltzmann method

We use the lattice Boltzmann method to compute the flow field solving the incompressible Navier–Stokes equations. This method discretizes the Boltzmann equation for the particle density $f(x, v, t)$ (distribution function) in phase space by choosing a finite number of velocities $v \in \{c_0, \ldots, c_n\}$ on a regular grid in space. We implement the *D3Q15 model* for 3D flow, which has a cubic grid and 15 velocities $c_0 = 0$, and $c_1, \ldots, c_7$ given by the columns of the matrix

$$\begin{bmatrix} 1 & 0 & 0 & 1 & -1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & -1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 1 & -1 \end{bmatrix} \tag{3}$$

as well as $c_i = -c_{i-7}$, for $i = 8, \ldots, 14$. We focus in our work on the simple and efficient *lattice BGK model*, which uses the BGK collision operator and a single relaxation time [37]. A more general model using multiple relaxation times is presented e.g. in [4]. It has better stability properties but needs more operations per time step. In the lattice BGK method, the collision operator is a diagonal matrix such that the evolution is given by the *lattice Boltzmann equation*

$$f_i(x + c_i, t + 1) = f_i(x, t) - \frac{1}{\tau}(f_i(x, t) - f_i^{\mathrm{eq}}(x, t)). \tag{4}$$

From the populations $f_i(x, t) = f(x, c_i, t)$ we can deduce density and velocity by taking the moments

$$\rho(x, t) = \sum_{i=0}^{14} f_i(x, t), \qquad u(x, t) = \sum_{i=0}^{14} f_i(x, t) c_i. \tag{5}$$

The parameter τ is the relaxation parameter for the BGK collision operator and controls the kinematic viscosity $\nu = \frac{1}{6}(2\tau - 1)$. Furthermore, the desired Navier–Stokes equations are obtained on the macroscopic level using the equilibrium distribution

$$f_i^{\mathrm{eq}}(f) \equiv f_i^{\mathrm{eq}}(\rho, u) = f_i^* \left(\rho + 3c_i \cdot u + \frac{9}{2}(c_i \cdot u)^2 - \frac{3}{2}u^2 \right) \tag{6}$$

with the corresponding D3Q15 weight factors

$$f_i^* = \begin{cases} \dfrac{2}{9}, & |c_i|^2 = 0, \\ \dfrac{1}{9}, & |c_i|^2 = 1, \\ \dfrac{1}{72}, & |c_i|^2 = 3. \end{cases} \tag{7}$$

We note that with the equilibrium above the physical density variations in the lattice BGK method are given by $\varrho(1 + \rho)$, and the method is not completely incompressible. Moreover, pressure is related to the lattice density ρ by $p = \frac{\varrho}{3}\rho$, where ϱ is the mass density in (1).

The LBM evolution (4) can be divided into the collision step

$$f_i^+(x, t) = f_i(x, t) - \frac{1}{\tau}(f_i(x, t) - f_i^{\mathrm{eq}}(x, t)), \tag{8}$$

and the propagation step

$$f_i(x + c_i, t + 1) = f_i^+(x, t). \tag{9}$$

This is the way the method is usually implemented in code, alternating purely local collisions with propagations conveying data to nearest neighbors, and the code can efficiently run on parallel computers. Here we have to take into account that the latter step is not always feasible. When a point x_b lies near a solid boundary and a neighbor along the link c_i is outside of the fluid domain, $x = x_b - c_i \notin \Omega_i$, the corresponding population has to be prescribed artificially: $f_i(x_b, t + 1) = \tilde{f}_i(x_b, t)$. We use the boundary condition of Bouzidi et al. [14] at solid walls :

$$\tilde{f}_i(x_b, t) = \begin{cases} 2q f_{i^*}^+(x_b, t) + (1 - 2q) f_{i^*}^+(x_b + c_i, t), & q < 0.5, \\ \dfrac{1}{2q} f_{i^*}^+(x_b, t) + \dfrac{1 - 2q}{2q} f_{i^*}^+(x_b + c_i, t), & q \geq 0.5. \end{cases} \tag{10}$$

(Here, i^* refers to the index of the opposite direction, $c_{i^*} = -c_i$.) This is a second-order accurate no-slip boundary condition for arbitrary walls. When the wall moves with velocity w, we have to add the additional term

$$\Delta\tilde{f}_i = \begin{cases} 6f_{i^*}c_i \cdot w, & q < 0.5, \\ \frac{3}{q} f_{i^*}c_i \cdot w, & q \geq 0.5. \end{cases} \tag{11}$$

The parameter $q \in [0, 1]$ denotes the distance from the node to the surface normalized by the link length.

To generate the flow, a vector-valued volume force $F(x, t)$ can be applied in the domain by adding a term of the form $\Delta f_i^+ = 3f_i^* c_i \cdot a$ in the collision step. For example, gravity in x_3-direction uses $a = -ge_3$, where g is the gravitation constant and e_3 the corresponding unit vector. Moreover, there are also pressure or velocity boundary conditions available to create flows in various situations [15–18].

Remark 1. We have presented the lattice BGK method in non-dimensional form in lattice units such that $\Delta x = 1$ and $\Delta t = 1$ to simplify the notation. To establish the connection with the presentation in terms of physical quantities, we have to introduce the lattice sound speed $c_s = \sqrt{\frac{1}{3}\frac{\Delta x}{\Delta t}}$. Then the viscosity is $\nu = \frac{c_s^2 \Delta t}{2}(2\tau - 1)$ and the equilibrium reads

$$f_i^{\mathrm{eq}}(f) \equiv f_i^{\mathrm{eq}}(\rho, u) = f_i^* \left(\rho + \frac{c_i \cdot u}{c_s^2} + \frac{(c_i \cdot u)^2}{2c_s^4} - \frac{u^2}{2c_s^2} \right). \tag{12}$$

The pressure is $p = c_s^2 \varrho \rho$ and force terms have the form $\frac{\Delta t}{c_s^2} f_i^* c_i \cdot a$. Moreover, we note that in any case the lattice density ρ in our model is a dimensionless quantity without physical units.

Fluid nodes can change their type (e.g. from fluid 1 to fluid 2, and vice versa) when the interface Γ_t moves. Then the populations in these nodes have to be reinitialized or refilled. We choose the equilibrium/non-equilibrium refill described in [19,20]. The refill procedure consists of four substeps:

(1) Interpolate density and velocity from neighbors;
(2) Compute the corresponding equilibrium;
(3) Copy the non-equilibrium neighbor;
(4) Add equilibrium and non-equilibrium parts.

At a grid point x near the surface we choose an inward pointing direction c_i, e.g. the direction of the smallest angle with the surface normal. Then the density is interpolated using exiting densities (at time step t) of the three nearest neighbors in the interior of the subdomain (see Fig. 2):

$$\hat{\rho} = 3\rho(x - c_i, t) - 3\rho(x - 2c_i, t) + \rho(x - 3c_i, t). \tag{13}$$

Interpolation of the velocity uses two neighbors and the velocity u_s at the interface point x_s, which is obtained from the level set method below or, in the case of solid obstacles, from other suitable sources:

$$\hat{u} = \frac{2}{q^2 + 3q + 2} u_s + \frac{2q}{q+1} u(x - c_i, t) + \frac{2q}{q+2} u(x - 2c_i, t). \tag{14}$$

The new populations are

$$f_i(x, t) = f_i^{\mathrm{eq}}(\hat{\rho}, \hat{u}) + f_i^{\mathrm{neq}}(x - c_i, t), \tag{15}$$

where $f_i^{\mathrm{neq}} = f_i - f_i^{\mathrm{eq}}$ is the non-equilibrium distribution.

2.3. The level set method

The interface data necessary for boundary conditions or refill procedures above, e.g. distance q, normal n, curvature κ etc., is created from the boundary representation handled by the level set method.

At the beginning, the initial signed distance function $\varphi = \varphi(x, 0)$ is constructed from a triangulation of the surface. To reduce computation time and memory requirements, we apply the narrow band technique, which only stores points near the surface and merely retains several bands of nodes around the actual interface (usually 5 to 10 bands are stored).

To better maintain the signed distance property, the velocity field v is constructed using the constant velocity extensions method of Adalsteinsson and Sethian [21], i.e:

$$v(t) = u(t) \quad \text{on } \Gamma_t, \tag{16}$$

$$\nabla v(t) \nabla \varphi(t) = 0 \quad \text{in } \Omega. \tag{17}$$

For each grid node adjacent to the zero iso-surface, the nearest point on the surface is determined. Then, the velocity at this projection point is computed from the discrete flow velocity field u, taken from the lattice Boltzmann method,

in the neighborhood by extrapolation using a linear or quadratic polynomial ansatz. We note that u is continuous at the interface while this may not be true for its derivatives. The extrapolation is therefore done separately on both sides of Γ_t. For the velocity v at the projection point we take the mean value of both extrapolations and store it in the grid node under consideration. Having constructed the velocities adjacent to the surface, the rest of the velocity field in the narrow band is eventually constructed via the fast marching method [22].

The level set equation of Osher and Sethian [23]

$$\varphi_t + v \cdot \nabla\varphi = 0, \tag{18}$$

is solved by the Hamilton–Jacobi WENO scheme [24], which is fifth-order accurate in space. Time discretization is done by a second-order accurate Runge–Kutta method [24].

During surface evolution, the level set $\{\varphi = 0\}$ may touch the boundary of the narrow band. Then a reinitialization of φ and an adjustment of the narrow band is performed. This usually also becomes necessary after some time steps since the function φ can gradually lose the signed distance property when numerical errors accumulate. In our code this is done efficiently using the fast marching method of Adalsteinsson and Sethian [22].

To estimate surface parameters, we use a method developed in [25], where the level set function φ is fitted locally by a polynomial of chosen degree. Interface normal, curvature and other surface parameters are then approximated by taking the corresponding quantities of the polynomial. With this approach it is possible to reach an arbitrarily high order of approximation of surface properties.

The level set method can suffer from mass losses due to numerical errors; for example it is prone to shrinking convex surfaces. To prevent this, we use the method of Smolianski [26] to preserve the volume, of e.g. a ball, by correcting the signed distance function φ with the correction term

$$c_\varphi = \frac{V_{\text{exact}} - V}{S(\Gamma)}. \tag{19}$$

where V_{exact} is the known volume, V the actual volume in the simulation and $S(\Gamma)$ the surface area of the interface Γ. The correction is added to the current level set function $\varphi := \varphi + c_\varphi$. (See also [24,27] for other approaches to control mass in the level set method.) We refer to [28] for details of the lattice Boltzmann and level set methods used in the our algorithm.

3. Interface boundary conditions for LBM

Starting from the initial surface, the level set code creates the signed distance function and generates the surface description for LBM (links intersecting the interface, normalized distance q, normal n, tangential vectors t_1, t_2, principal curvatures κ_1, κ_2, and velocity v). For the update of the geometry it receives the current velocity field from LBM and computes the surface evolution in the narrow band. After the surface motion is completed, the level set code constructs a data set for the new surface and passes it to LBM (Fig. 3).

At the time-dependent interface Γ_t, populations are also undefined in the propagation step. We therefore use a novel LBM interface boundary condition developed in [28]. Assuming an interface boundary point $x_1 \in \Omega_1$ and a neighbor $x_2 = x_1 + c_i \in \Omega_2$ (see Fig. 1) with interface intersection at $x_1 + qc_i$, we define

$$f_i(x_2, t+1) = f_{i^*}^+(x_2, t) + 6f_i^* c_i \cdot \tilde{u} + 6f_i^* \Lambda_i : A, \tag{20}$$

where $\Lambda_i = c_i \otimes c_i - \frac{1}{3}|c_i|^2 I$ is the second-order velocity tensor, $\tilde{u} = (1-q)u^{(1)}(x_1) + qu^{(2)}(x_2)$ and $A = -q(1-q)[S] - (q-0.5)S(x_2)$. The tensor $S_{kl} = \frac{1}{2}(\partial_k u_l + \partial_l u_k)$ can be computed in LBM by evaluating higher velocity moments

$$S = -\frac{3}{2\tau}\sum_i (f_i - f_i^{\text{eq}}) c_i \otimes c_i. \tag{21}$$

Note that the tensors are evaluated after the surface update and also after the subsequent refill. Using the representation $[\mu S] = [\mu]\bar{S} + \bar{\mu}[S]$, with averages $\bar{S} = \frac{1}{2}(S(x_1) + S(x_2))$ and $\bar{\mu} = \frac{1}{2}(\mu^{(1)} + \mu^{(2)})$, we derive relations involving the pressure jump and surface tension:

$$\begin{aligned} [S] : n \otimes n &= \frac{1}{2\bar{\mu}}\left([p] + 2\sigma\kappa\right) - \frac{[\mu]}{\bar{\mu}}\bar{S} : n \otimes n, \\ [S] : n \otimes t &= -\frac{[\mu]}{\bar{\mu}}\bar{S} : n \otimes t. \end{aligned} \tag{22}$$

Note that the pressure difference in our model is given by

$$[p] = \frac{1}{3}\left(\rho(x_2, t)\varrho^{(2)} - \rho(x_1, t)\varrho^{(1)}\right). \tag{23}$$

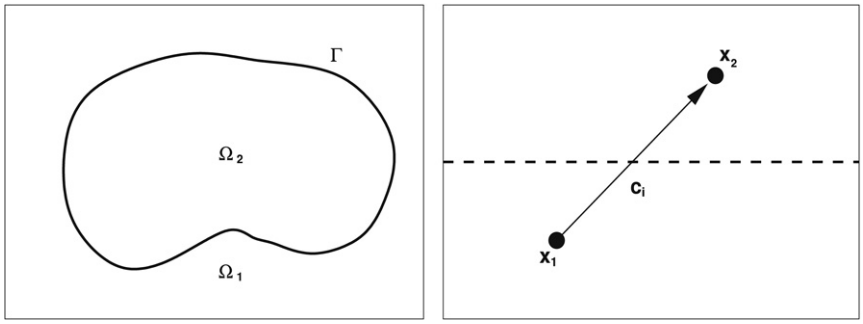

Fig. 1. Left: Partitioning of the domain in the presence of two phases. Right: Link crossing the interface from fluid 1 to fluid 2.

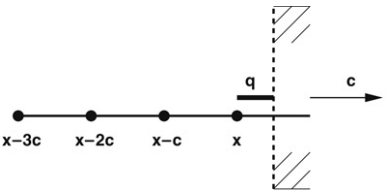

Fig. 2. Refill at the boundary in a point x using the direction $c = c_i$.

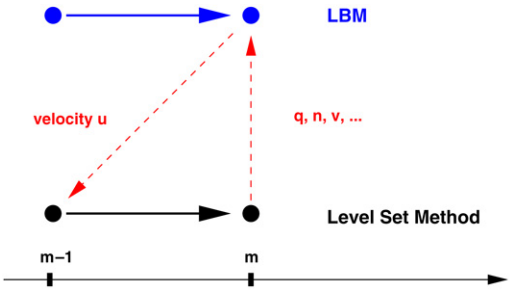

Fig. 3. Data exchange between LBM and the level set method at update time step m. LBM passes the current flow field u, while the level set method computes the description for new surface points (e.g. distance q, normal n, velocity v).

Based on these relations, the product $\Lambda : [S]$, which appears in $\Lambda : A$ of Eq. (20), can be computed using the local basis given by the normal, n, and tangential vectors, t_1, t_2 (see [28])

$$\Lambda_i : [S] = ([S] : n \otimes n)((n \cdot c_i)^2 - |c_i|^2/3) + \sum_{k=1}^{2} 2([S] : n \otimes t_k)(n \cdot c_i)(t_k \cdot c_i). \tag{24}$$

In this way surface tension and the jump conditions (2) are incorporated via (20) in the LBM framework.

The surface parameters are taken from the data generated by the level set method after the interface has been moved. It should be noted that the fluid variables – p, u and S – entering in the interface condition are evaluated after the interface update and the refill. Hence the parameters $\tilde{u}$ and A are computed explicitly.

4. Tests with simple flows

In order to analyze the interface conditions separately, we check its behavior for simple two-phase Couette and Poiseuille flows in axis aligned channels. The flow is in x-direction from left to right with vanishing vertical velocity component.

4.1. Couette flow

In the Couette case, the horizontal component of the piecewise linear velocity field is given by

$$U(y) = \begin{cases} u_0 \left(\dfrac{\mu_1}{\mu_1 + \mu_2} y + \dfrac{\mu_2}{\mu_1 + \mu_2} \right), & y \geq 0, \\ u_0 \left(\dfrac{\mu_2}{\mu_1 + \mu_2} y + \dfrac{\mu_2}{\mu_1 + \mu_2} \right), & y < 0, \end{cases} \tag{25}$$

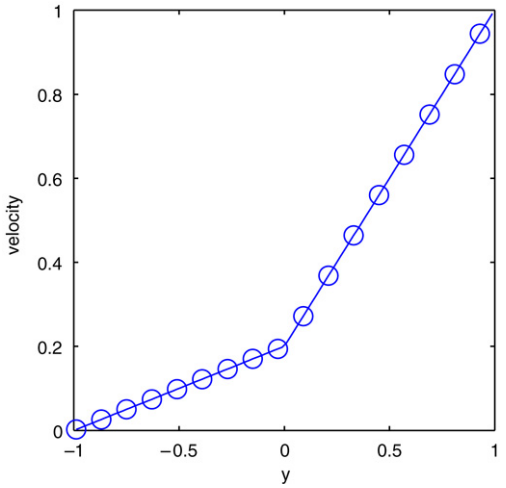

Fig. 4. Analytical (solid) and numerical (circles) velocity profile across the channel of a Couette flow with viscosities $\mu_1 = \frac{4}{6}$ and $\mu_2 = \frac{1}{6}$ below and above the interface.

Table 1
The velocity error with respect to different grid sizes shows that the lattice Boltzmann scheme produces the exact Couette flow field. The viscosity ratio is 1:4.

Grid size	h				
	1/10	1/20	1/30	1/40	1/50
Error	0.01934e−12	0.05018e−12	0.16031e−12	0.18424e−12	0.11685e−12

which is zero at the lower boundary $y = -1$, and u_0 at $y = 1$ with a change of slope at the flat interface at $y = 0$. The pressure is constant.

A numerical experiment shows that the lattice Boltzmann scheme recovers the exact velocity and pressure field (see Fig. 4 and Table 1).

In order to analyze this situation more carefully, we start by considering the single-phase Couette flow. In this case, we can derive an analytic solution of the lattice Boltzmann equation. Following the expansion procedure described in [29,30], we construct the coefficients $f^{(i)}$ in the regular expansion

$$f_i(r) = f_i^{(0)}(r) + hf_i^{(i)}(r) + h^2 f_i^{(2)}(r) + \cdots$$

where r denotes an arbitrary grid point. In general, $f_i^{(0)}$ are the weights of the equilibrium distribution and $f_i^{(1)}$ is the linear part of the equilibrium with zero density, i.e.

$$f_i^{(1)} = 3f_i^* c_i \cdot u$$

where u is a solution of the incompressible Navier–Stokes equation. The corresponding pressure p appears as contribution $3f_i^* p$ in the coefficient $f_i^{(2)}$ which also contains quadratic terms in u as well as the stress components $S[u]$. The remaining coefficients $f_i^{(k)}$ depend on higher derivatives of u and p as well as on solutions to certain linearized Navier–Stokes problems. However, in the Couette case where the velocity is linear, all additional fields vanish and the expansion terminates at order 4. Inserting $u = (U, 0, 0)$ and $p = 0$ in the general expressions (see [31]), we find the coefficients

$$\begin{aligned}
f_i^{(0)} &= f_i^* \\
f_i^{(1)} &= 3f_i^* U c_{ix} \\
f_i^{(2)} &= \frac{9}{2} f_i^* U^2 \left(c_{ix}^2 - \frac{1}{3}\right) - 3\tau f_i^* U' c_{iy} c_{ix} \\
f_i^{(3)} &= -9f_i^* U U' \tau c_{iy} \left(c_{ix}^2 - \frac{1}{3}\right) \\
f_i^{(4)} &= 9f_i^* (U')^2 \tau \left(\tau - \frac{1}{2}\right) c_{iy}^2 \left(c_{ix}^2 - \frac{1}{3}\right).
\end{aligned}$$

In a two-phase situation, we use the same form of the coefficients where U is now piecewise linear and τ jumps at the interface to account for the different viscosities. By construction, the corresponding expansion, which we denote F, satisfies

the lattice Boltzmann equation exactly in each phase. However, inserting F into the interface condition, we find a residue of magnitude h^3. More specifically, the residue $\Delta_i = h^3\delta_i^{(3)} + h^4\delta_i^{(4)}$ has components

$$\begin{aligned}\delta_i^{(3)} &= 9(1-2\tau)f_i^* UU' c_{iy}\left(c_{ix}^2 - \frac{1}{3}\right)\\ \delta_i^{(4)} &= 9(2q-1)(\tau - 1/2)f_i^*(U')^2 c_{iy}^2\left(c_{ix}^2 - \frac{1}{3}\right)\end{aligned} \tag{26}$$

where $i \in I^{(k)}$ denotes the incoming directions in the corresponding subdomains which are defined by $c_{iy} = +1$ in the upper phase ($k = 2$) and $c_{iy} = -1$ in the lower one ($k = 1$).

The result shows that glueing together the exact single-phase solutions of the lattice Boltzmann equation does not yield the exact solution f of our algorithm with interface condition. Nevertheless, F can be used to construct the solution by setting

$$f_i = F_i + G_i$$

and trying to find an appropriate distribution function G.

In order to motivate the subsequent definition of G, we briefly describe the construction principles. First, because of symmetry and stationarity of the problem, we expect that G is stationary and depends only on the y coordinate. Secondly, based on our numerical experiment, we expect that f has the exact velocity and pressure moments. Since F also produces the exact two-phase velocity and pressure field, we are led to the assumption that G has vanishing moments. This, on the other hand, implies that $f_i^{eq}(F+G) = f_i^{eq}(F)$ and since F satisfies the lattice Boltzmann equation exactly, the sum $F+G$ is a solution of the lattice Boltzmann equation if and only if G satisfies the simpler rule

$$G_i(r + hc_i) = G_i(r) - \frac{1}{\tau}G_i(r). \tag{27}$$

Combined with the assumed y-dependence, this implies a simple exponential decay of G normal to the interface.

Summarizing our considerations, we are led to the ansatz

$$G_i(r) = \begin{cases}(1-1/\tau^{(2)})^n \phi_i^{(2)} & y = (n+0.5)h\\ (1-1/\tau^{(1)})^n \phi_i^{(1)} & y = -(n+0.5)h\end{cases}$$

where $n \geq 0$ is an integer and $\phi_i^{(k)} = 0$ for the outgoing directions $i \notin I^{(k)}$. Moreover, $\phi^{(k)}$ have to satisfy the vanishing moment conditions

$$\sum_i \phi_i^{(k)} = 0, \qquad \sum_i \phi_i^{(k)} c_i = 0, \quad k = 1, 2.$$

Due to the structure of the D3Q15 velocity set, these constraints lead to the condition that, for example, the values $\phi_i^{(2)}$ are equal for the diagonal incoming directions and sum up to the negative value of $\phi_i^{(2)}$ belonging to the normal direction. A similar consideration applies to $\phi^{(1)}$.

By construction G satisfies (27) and has the vanishing moment property so that $F+G$ is a solution of the lattice Boltzmann equation. It only remains to fix the incoming populations of $\phi^{(k)}$ such that the interface condition (20) is also satisfied exactly. Due to the linearity of (20), it suffices to evaluate it separately for F and G, keeping in mind that insertion of F leads to the residue Δ described above.

When evaluating (20) for the function G, we need the velocity moments $u^{(1)}$ and $u^{(2)}$ to compute the interface value $\tilde{u} = (1-q)u^{(1)} + qu^{(2)}$. Since G_i equals $\phi_i^{(k)}$ at the interface nodes in phase k we see that $u^{(k)} = \sum \phi_i^{(k)} c_i = 0$ and hence $\tilde{u} = 0$. Also, since $\phi_{i^*}^{(k)} = 0$ for incoming $i \in I^{(k)}$, the first term on the right-hand side of (20) drops out.

Next, we need the tensor S on both sides of the interface according to formula (21). Noting that $\phi^{(k)}$ equal their non-equilibrium parts because of the vanishing moment property, we find at the interface points in phase ($k = 2$)

$$S^{(2)} = -\frac{3}{2\tau}\sum c_i \otimes c_i \phi_i^{(2)}.$$

Here, the incoming directions are $(\pm 1, 1, \pm 1)$ with $\phi_i^{(2)} = \alpha^{(2)}$ and $(0, 1, 0)$ with $-4\alpha^{(2)}$ as corresponding $\phi_i^{(2)}$. Evaluating the sum yields

$$S^{(2)} = -\frac{6\alpha^{(2)}}{\tau^{(2)}}\begin{pmatrix}1 & 0 & 0\\ 0 & 0 & 0\\ 0 & 0 & 1\end{pmatrix}. \tag{28}$$

With a similar computation for the lower phase, we eventually find the average tensor with some constant β

$$\bar{S} = \beta\begin{pmatrix}1 & 0 & 0\\ 0 & 0 & 0\\ 0 & 0 & 1\end{pmatrix}.$$

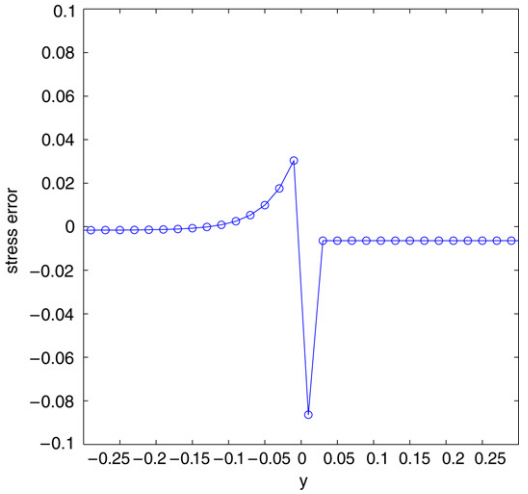

Fig. 5. The error of the S_{11} component divided by h^3 for a Couette channel flow with viscosity $\mu_1 = 4/6$ ($y < 0$) and $\mu_1 = 1/6$ ($y > 0$).

According to (22), the average $\bar{S}$ is multiplied with $n \otimes n$ and $n \otimes t$ respectively, which yields zero in all cases because of the vanishing second row and second column in $\bar{S}$.

Finally, the pressure jump $[p]$ associated to G is also zero because the density moment vanishes on both sides. Altogether, the right-hand side of (20) vanishes upon insertion of G so that the residue is G itself for any valid choice of the building blocks $\phi^{(k)}$. As a consequence, the residue of the sum $F + G$ has the form $G + \Delta$ and it vanishes by setting $\phi_i^{(k)} = -\Delta_i^{(k)}$ for $i \in I^{(k)}$. This choice is in accordance with the structural assumptions on $\phi^{(k)}$ because one can easily check that Δ defined by (26) satisfies

$$\sum_{i \in I^{(k)}} \Delta_i^{(k)} = 0, \qquad \sum_{i \in I^{(k)}} c_i \Delta_i^{(k)} = 0, \quad k = 1, 2.$$

The analytic expression $f = F + G$ obtained in this way thus turns out to be the exact solution of the lattice Boltzmann equation combined with our interface condition. We see that the velocity and pressure are obtained exactly by the numerical scheme in this case, while the tensor S computed with (21) is only a first-order accurate approximation of the exact symmetric velocity gradient. In fact, for the contribution F, a direct computation shows that

$$-\frac{3}{2\tau} \sum_i c_i \otimes c_i (F_i - F_i^{eq}) = h^2 \frac{U'}{2} \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} - h^4 (\tau - 1/2)(U')^2 \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

which is the symmetric velocity gradient in leading order plus a contribution two orders higher (since U' is constant, the diagonal entry is a negative constant in each phase).

The tensor contribution due to G follows from the construction above which shows that the second moment is of the form

$$-\frac{3}{2\tau} \sum_i c_i \otimes c_i (G_i - G_i^{eq}) = -\frac{6\alpha}{\tau} (1 - 1/\tau)^n \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

where n counts the distance of the grid point to the interface and α is a piecewise constant function with values $\alpha^{(k)}$ in the two phases which are determined by $\Delta_i^{(k)}$, evaluated at a diagonal incoming direction $i \in I^{(k)}$ where $f_i^* = 1/72$, $c_{ix}^2 = 1$, and $c_{iy} = (-1)^k$. Specifically, we have

$$\alpha^{(k)} = (-1)^k \frac{h^3}{12} (1 - 2\tau^{(k)}) UU' + (-1)^k \frac{h^4}{12} (2q - 1) \left(\tau^{(k)} - \frac{1}{2} \right) (U')^2$$

where U is evaluated at distance $(-1)^k h/2$ of the interface.

The theoretical predictions on the upper diagonal component of the second-order moment tensor completely agree with the numerical results. In Fig. 5, the values of the 11-component are displayed over a small y-interval around the interface. Since $\tau^{(2)} = 1$, the decay factor $(1 - 1/\tau^{(2)})$ is actually zero so that the error persists only in the first node above the interface. At the bulk nodes with $y > 0$, one can see a small negative constant value which is the fourth-order contribution from the second-order moment of F. For $y < 0$, the quick exponential decay of the leading order error is clearly visible where the asymptotic value is again the negative constant value of the fourth-order F-contribution. The factor $(-1)^k$ in the definition of $\alpha^{(k)}$ explains the jump in sign when crossing the interface.

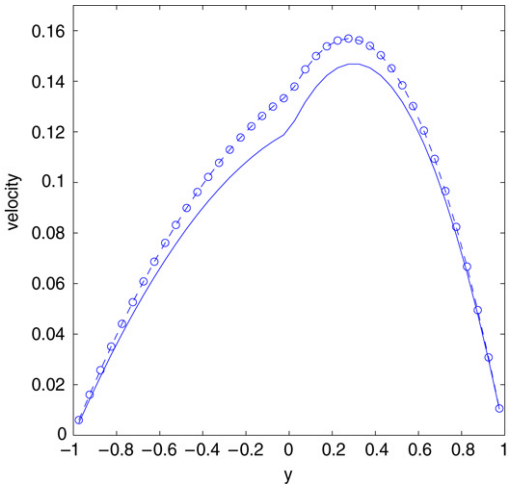

Fig. 6. Velocity profile for a Poiseuille channel flow with viscosity $\mu_1 = 4/6\,(y < 0)$ and $\mu_1 = 1/6\,(y > 0)$.

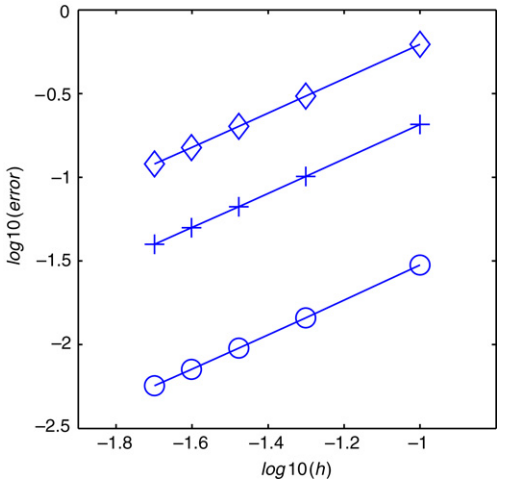

Fig. 7. Grid convergence for three Poiseuille channel flows with different viscosity ratios $1:4\,(\circ)$, $1:20\,(+)$, $1:100\,(\diamond)$.

4.2. Poiseuille flow

In the Poiseuille case, the wall velocity is zero and the piecewise parabolic flow field is driven by a constant pressure gradient leading to

$$U(y) = \begin{cases} \dfrac{\Delta p_0}{2\mu_2}\left(y^2 - \dfrac{\mu_1-\mu_2}{\mu_1+\mu_2}y - \dfrac{2\mu_2}{\mu_1+\mu_2}\right), & y \geq 0, \\ \dfrac{\Delta p_0}{2\mu_1}\left(y^2 - \dfrac{\mu_1-\mu_2}{\mu_1+\mu_2}y - \dfrac{2\mu_1}{\mu_1+\mu_2}\right), & y < 0. \end{cases} \tag{29}$$

In this case, a numerical experiment shows that the velocity is recovered only with first-order accuracy, as indicated by the convergence study in Fig. 7 (see Fig. 6 for a comparison of exact and approximate velocity profiles). The reason for the difference to the Couette flow is that the second-order derivatives of the velocity field (which vanish in the Couette but not in the Poiseuille case) act as sources in a stress condition for the equation determining the velocity error. It is possible to change the interpolation for the interface velocity $\tilde{u} = (1-q)u^{(1)} + qu^{(2)}$ in such a way that these source terms disappear. However, this approach reduces the stability of the method and increases the effort because additional neighbor nodes are required to compute the interpolation. In general, it turns out that velocity, pressure and symmetric velocity gradient (and thus the full stress tensor) are recovered with first-order accuracy. For further details we refer to [28].

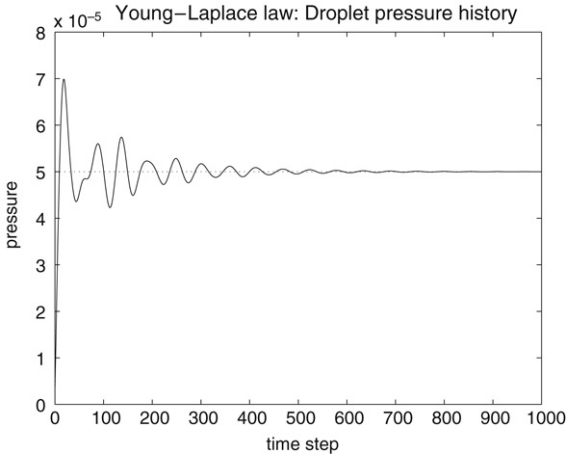

Fig. 8. Evolution of the pressure for $N = 32$ over 1000 time steps.

5. Further numerical results

To demonstrate the overall feasibility of the approach, we apply the proposed method to several well-known two-phase flow problems. We simulate the Young–Laplace experiment for the pressure in a bubble and present results for Hele–Shaw viscous fingering and rising bubbles.

5.1. The Young–Laplace law

The Young–Laplace experiment for the pressure difference inside a bubble is a benchmark problem for the implementation of surface tension. The pressure p inside a spherical bubble of radius r surrounded by a second fluid is proportional to the surface tension coefficient σ according to the Young–Laplace law

$$p = \frac{2\sigma}{r} = 2\sigma\kappa, \qquad \text{mean curvature } \kappa = \frac{1}{2}(\kappa_1 + \kappa_2). \tag{30}$$

We consider a unit cube $[0, 1]^3$ containing a bubble of radius $r = 0.25$ at the center. For the simulations we used an equidistant grid of N nodes in each coordinate direction and periodic boundary conditions at the sides of the cube. The viscosities were chosen equal $\nu_i = \frac{1}{6}$ and the densities were $\rho_1 = 1$ and $\rho_2 = 10$. Simulations were started with zero pressure difference and stopped when the initial fluctuations were reduced sufficiently.

Fig. 8 shows a time series of initial pressure oscillations, which are in general quickly damped until the final value is attained when the viscosities are not exceedingly high. For our choice of viscosities it could be assumed that oscillations had sufficiently subsided when simulations were stopped at $t = 500$ on a grid with $N = 16$ (correspondingly, according to the diffusive scaling, $t = 2000$ for $N = 32$, and $t = 8000$ for $N = 64$). Grid convergence of the pressure error is shown in Fig. 9.

The numerical method leads to spurious currents at the interface of the bubble (Fig. 10), an artifact also observed with many other schemes. To differentiate between the error introduced by the LBM itself and the error added by the level set method, we present the magnitude of the spurious velocities, U, in non-dimensional form $U\mu/\sigma$ in Table 2. The first column lists the currents when the level set method was used to create position, curvature and normal, while the second column is based on simulations using the analytical values. It reveals that the currents are four to five orders of magnitude larger when the level set method is used. Nevertheless the currents are very small and they are comparable with other values reported in the literature [26]. On the one hand, in the analytical case we get first-order convergence. This error is attributed to interface coupling used in our LBM implementation. On the other hand, when the level set method is coupled we find second-order convergence because on the grids used here the surface reconstruction error prevails. The order can be explained by the fact that we used third-order polynomial reconstruction in the level set method. In [25] it was shown that the corresponding curvature error is of second order.

The numerical experiments also showed that the results depend on the details of the implementation, e.g. the grid refinement of the level set grid with respect to the LBM grid or the order of curvature reconstruction for least squares estimation of curvature in the level set method. Our experience indicates that using third-order curvature and a level set grid that coincides with the LBM grid is appropriate. We note, furthermore, that simulations with density ratios of 1:1000 or kinetic viscosity ratios of 1:100 could be successfully performed. However, accuracy deteriorates and spurious currents increase when the ratio attains excessively large values.

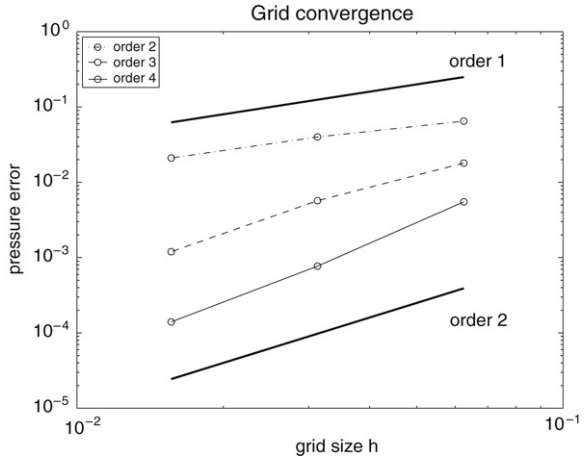

Fig. 9. Grid convergence of the pressure in the Young–Laplace experiment. Three different orders of reconstruction polynomial in the level set method are used: (-.), (- -) order 3, (-) order 4.

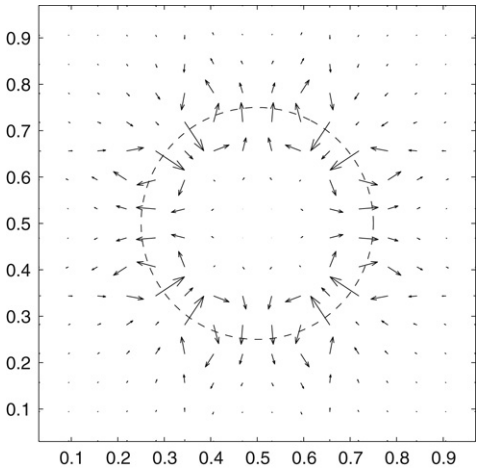

Fig. 10. Spurious currents at the interface of the bubble in the Young–Laplace experiment.

Table 2
Comparison of the spurious currents in the Young–Laplace experiment using the level set values or the analytical values for surface representation. The non-dimensional velocities $U\mu/\sigma$ are presented for three different grids of grid size h.

h	Level set method $U\mu/\sigma$	Analytical method $U\mu/\sigma$
$\frac{1}{16}$	4.037×10^{-3}	5.460×10^{-8}
$\frac{1}{32}$	1.184×10^{-3}	3.070×10^{-8}
$\frac{1}{64}$	2.429×10^{-4}	1.529×10^{-8}

5.2. Hele–Shaw viscous fingering

In viscous fingering experiments, a fluid of low viscosity pushes a second fluid of high viscosity owing to a pressure gradient or gravity. The interface is prone to instabilities which lead to the penetration of the first fluid into the domain of the second in the form of a finger. This is a model problem for flow with movable interfaces that includes surface tension and which is used in analytical studies as well as for benchmarking numerical simulations [32–34].

In our simulations we considered the flow in the narrow gap between two plates and investigated the finger by assuming that the tip can be fitted by an exponential shape as described in [33]. We used a channel $[0, 32] \times [-1, 1]$ with aspect ratio 16:1 using $16N$ grid points in x-direction, and N in y-direction. At the inlet (outlet), a pressure $p_{in} = 0.032$ ($p_{\text{out}} = -0.032$) was applied in x-direction to simulate a pressure gradient. At the walls, $y = \pm 1$, we imposed the usual no-slip boundary condition. An initial interface perturbation in the form of a sine shape was prescribed $y(x) = 1+0.5\cos(\pi x)$ at the beginning. We assumed a driving fluid of viscosity $\nu_1 = 0.1$ and a second fluid of higher viscosity $\nu_2 = 1$. The densities of the fluids

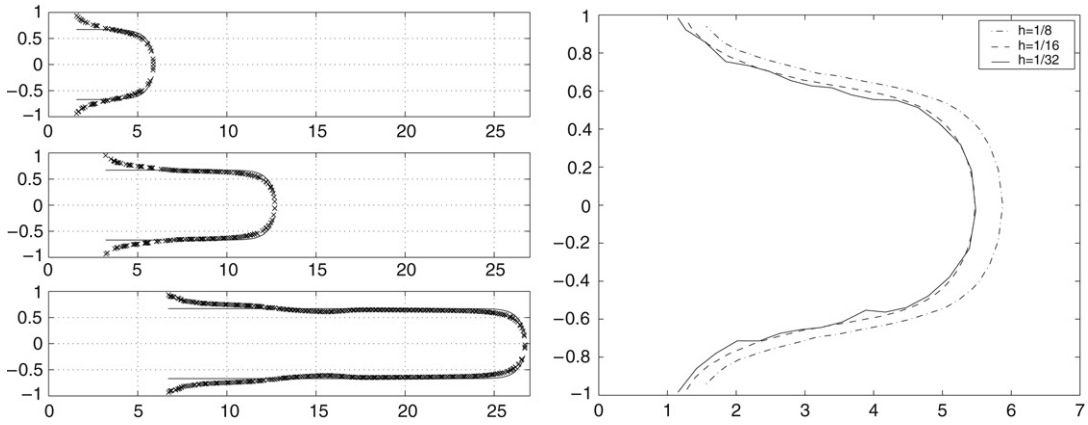

Fig. 11. Left: Evolution of the finger. Right: Grid convergence of the finger shape.

were the same, $\rho_1 = \rho_2 = 1$, and a surface tension coefficient $\sigma = 0.016$ was chosen. We used $N = 16$ for the grid such that the grid size parameter was $h = 1/8$.

In this simulation the interface undergoes considerable deformation and stretching. Therefore mass losses produced by the level set method might occur. We compared the volume of the two fluid phases over the course of the fingering experiment with the volume influx at the inlet. It revealed that there was excellent agreement of the recorded volume with the predicted volume from the influx with a deviation of less than 1% and no mass correction was necessary.

To check the correctness of the shape obtained on this rather coarse grid, we computed the results on two finer grids with $N = 32$ and $N = 64$, respectively, and confirmed the grid convergence of the finger shape (Fig. 11).

5.3. Rising bubbles

Studies of bubble dynamics are of practical interest for example in the design of chemical reactors in chemical engineering. The development of numerical schemes that can simulate these phenomena has been the subject of many investigations. We therefore applied our algorithm to the test problem of a rising bubble.

We considered a box of dimension $48 \times 48 \times 240$ containing a bubble of diameter $D = 16$ at position $(24, 24, 48)$ rising under buoyant forces. The heavy surrounding fluid had density $\rho_1 = 1$ and the light fluid in the bubble had density $\rho_2 = 0.1$. We fixed the acceleration due to gravity $g = 10^{-4}$ in negative z-direction. For the surface tension we chose the value $\sigma = 10^{-4}$. Both fluids had the same viscosity $\nu_1 = \nu_2 = 0.1$.

Fig. 12 shows the rising bubble at three different time steps and a close-up of the bubble shape taken from the level set iso-surface $\{\phi = 0\}$ (cut through the center in the xz-plane). The velocity field is plotted in the bubble's frame of reference. It shows the typical internal recirculations in the left and right halves of the bubble (see e.g. Clift et al. [35]). We can compare the observed rise velocity $U = 1.22 \times 10^{-2}$ with a simple approximate formula for a single bubble in an infinite domain ([35, eq. 8-5])

$$U_{\text{ref}} = \frac{2}{3}\sqrt{gR\frac{\Delta\rho}{\rho_1}}. \tag{31}$$

In our case $U_{\text{ref}} = 1.79 \times 10^{-2}$ which is satisfactory given the fact that we actually simulate a periodic array of bubbles and not a single bubble. (Also, the shape of the bubble influences U_{ref} and the factor $2/3$ can be replaced by a shape function that accommodates this.)

Other shapes could be obtained by varying the acceleration due to gravity and thereby the buoyancy and the rise velocity. Fig. 13 displays skirted and spherical bubbles. The shapes observed in these simulations correspond well to shapes expected from the classical diagram of Clift et al. [35], Fig. 2.5, where different regimes are characterized by the non-dimensional Reynolds, Eötvös and Morton numbers

$$\text{Re} = \frac{DU}{\nu}, \qquad \text{Eo} = \frac{gD^2\Delta\rho}{\sigma}, \qquad \text{Mo} = \frac{g\nu_1^4\rho_1^2\Delta\rho}{\sigma^3}. \tag{32}$$

Also the rise velocities coincide reasonably well, although it seems the fast bubbles are better approximated by the formula above. We measured $U = 3.81 \times 10^{-2}$ ($U_{\text{ref}} = 4 \times 10^{-2}$) for the skirted bubble and $U = 2.97 \times 10^{-3}$ ($U_{\text{ref}} = 8 \times 10^{-3}$) for the spherical bubble, respectively.

Finally, we also simulated bubbles with large density contrast $1 : 1000$ and higher viscosity ratios $\nu_1 = 0.1,\ \nu_2 = 1$. An example is shown in Fig. 14. In this case the bubble rose with velocity $U = 1.35 \times 10^{-2}$ and we find $U_{\text{ref}} = 1.88 \times 10^{-2}$.

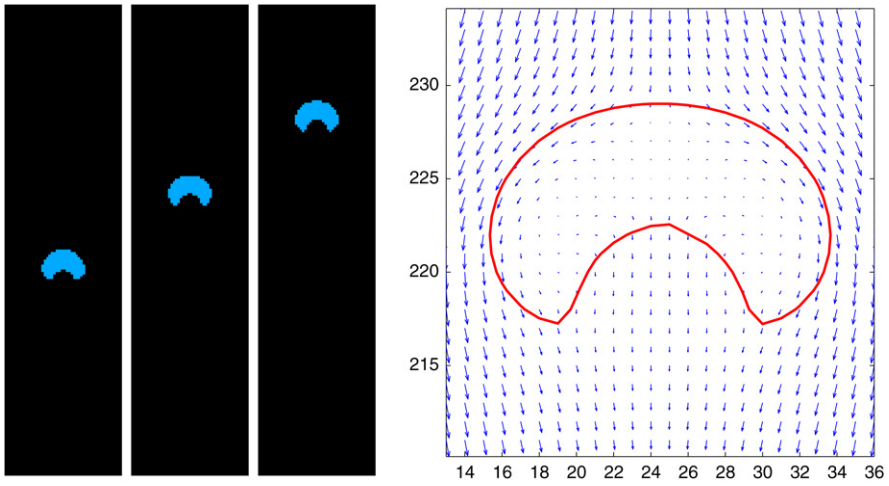

Fig. 12. Left: Three snapshots of a rising bubble at time steps 7500, 10 000 and 12 500. Right: Bubble shape at $t = 12\,500$ (non-dimensional parameters $Re = 1.95$, $Eo = 230.4$, $Mo = 9000$.).

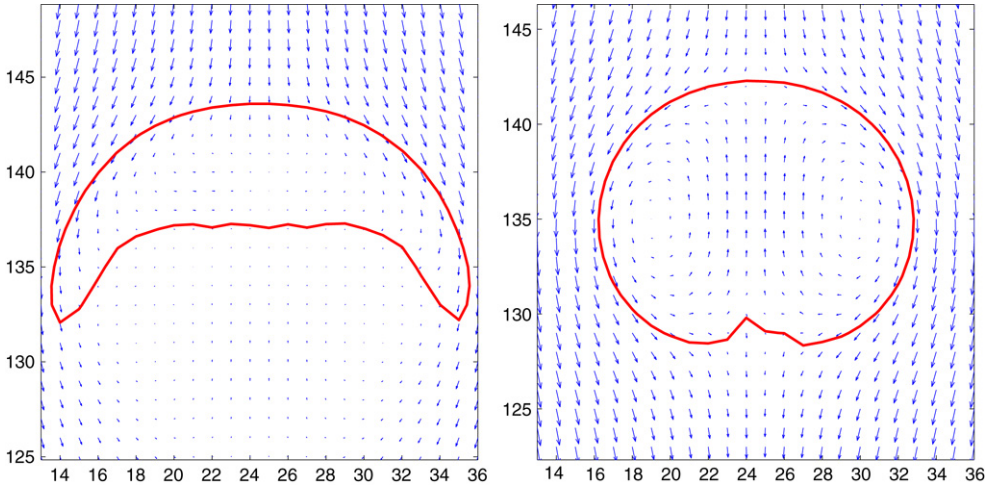

Fig. 13. Left: Dimpled bubble for $g = 5 \times 10^{-4}$ at $t = 3000$ ($Re = 6.09$, $Eo = 1100$, $Mo = 45\,000$). Right: Spherical bubble for $g = 2 \times 10^{-5}$ at $t = 30\,000$ ($Re = 0.476$, $Eo = 46.1$, $Mo = 1800$).

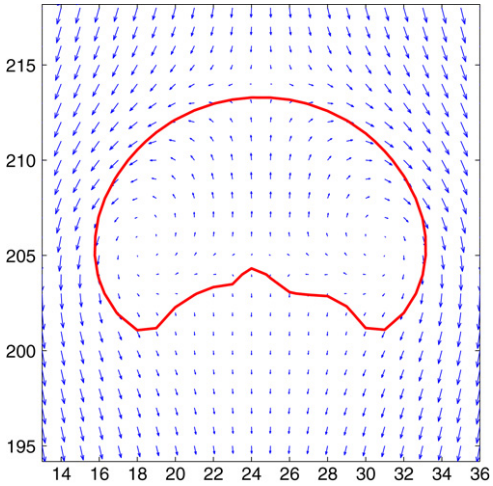

Fig. 14. Bubble shape in the case of large density differences $\rho_1 = 1$, $\rho_2 = 0.001$ at $t = 12\,500$ ($Re = 2.15$, $Eo = 255$, $Mo = 1000$).

Remark 2. We mention that the parameter range for the surface tension coefficient was restricted to small values up to $\sigma \approx 10^{-3}$. Nevertheless it was possible to obtain shapes ranging from spherical and dimpled to skirted by varying σ in the range $\approx 10^{-5}$–10^{-3} and/or the acceleration due to gravity $g \approx 10^{-6}$–10^{-4}. The corresponding Reynolds numbers ranged from approximately 0.1 to 10. In our numerical investigations we could simulate density ratios up to 1000 and viscosity ratios up to 100. For the simulation of typical physical situations on a grid of reasonable size of $O(10^2)$ nodes per direction this would yield small physical surface tension coefficients.

6. Summary and conclusions

We have presented a lattice Boltzmann method for immiscible two-phase flows with movable interfaces that uses the level set method to compute interface evolution. To incorporate surface tension and the macroscopic jump conditions at the interfaces, new boundary conditions were applied and included into the LBM framework. The algorithm was successfully validated in simple test cases with channel flows and a stationary bubble. When simulating surface evolution, the level set method has to account for the correct movement and deformation. This has been demonstrated by simulating a Hele–Shaw viscous fingering experiment. Moreover, the method could be applied to experiments with rising bubble. The range of applications and the fact that large density and viscosity differences can be simulated show that the scheme is an alternative tool for immiscible two-phase flow simulations in the realm of lattice Boltzmann methods.

Acknowledgments

M. Junk, K. Steiner and A. Klar acknowledge the support of the Deutsche Forschungsgemeinschaft (Germany) for this work by the grants *JU 440/1-3*, *STE 871/1* and *KL 1105/9*, respectively.

References

[1] S. Chen, G.D. Doolen, Lattice Boltzmann method for fluid flows, Ann. Rev. Fluid Mech. 30 (1998) 329–364.
[2] Sauro Succi, The Lattice Boltzmann Equation for Fluid Dynamics and Beyond, Clarendon Press, Oxford, 2001.
[3] D. d'Humières, Generalized Lattice-Boltzmann Equations, in: B.D. Shizgal, D.P. Weaver (Eds.), Rarefied Gas Dynamics: Theory and Simulations, Progress in Astronautics and Aeronautics, AIAA 159 (1992), 450–458.
[4] D. D'humières, I. Ginzburg, M. Krafczyk, P. Lallemand, L.-S. Luo, Multiple-relaxation-time lattice Boltzmann models in three dimensions, Phil. Trans. R. Soc. A 360 (2002) 437–451.
[5] D.H. Rothman, J.M. Keller, Immiscible cellular-automaton fluids, J. Stat. Phys. 53 (3) (1988) 119–1127.
[6] A.K. Gunstensen, D.H. Rothman, S. Zaleski, G. Zanetti, Lattice Boltzmann model of immiscible fluids, Phys. Rev. A 43 (8) (1991) 4320–4327.
[7] S. Geller, M. Krafczyk, J. Tölke, S. Turek, J. Hron, Benchmark computations based on lattice-Boltzmann finite element and finite volume methods for laminar flow, Comput. & Fluids 35 (8–9) (2006) 888–897.
[8] M.R. Swift, W.R. Osborn, J.M. Yeomans, Lattice Boltzmann simulation of nonideal fluids, Phys. Rev. Lett. 75 (1995) 830–833.
[9] T. Inamuro, T. Ogata, S. Tajima, N. Konishi, A lattice Boltzmann method for incompressible two-phase flows with large density differences, J. Comput. Phys. 198 (2004) 628–644.
[10] X. Shan, H. Chen, Lattice Boltzmann model for simulating flows with multiple phases and components, Phys. Rev. E 47 (3) (1993) 1815.
[11] M.R. Swift, E. Orlandini, W.R. Osborn, J.M. Yeomans, Lattice Boltzmann simulations of liquid-gas and binary fluid systems, Phys. Rev. E 54 (5) (1996) 5041–5052.
[12] X. Shan, H. Chen, Simulation of nonideal gases and liquid-gas phase transition by the Lattice Boltzmann equation, Phys. Rev. E 49 (4) (1994) 2941.
[13] P. Lallemand, L.S. Luo, Y. Peng, A lattice Boltzmann front-tracking method for interface dynamics with surface tension in two dimensions, J. Comput. Phys. 226 (2) (2007) 1367–1384.
[14] M. Bouzidi, M. Firdaouss, P. Lallemand, Momentum transfer of a Boltzmann-lattice fluid with boundaries, Phys. Fluids 13 (11) (2001) 3452–3458.
[15] Z. Guo, C. Zheng, B. Shi, An extrapolation method for boundary conditions in lattice Boltzmann method, Phys. Fluids 14 (6) (2002) 2007–2010.
[16] M. Junk, Z. Yang, A one-point boundary condition for the lattice Boltzmann method, Phy. Rev. E 72 (2005) 066701.
[17] M. Junk, Z. Yang, Outflow boundary conditions for the lattice Boltzmann method, Prog. Comput. Fluid Dyn. 8 (2008) 38–48.
[18] Q. Zou, X. He, On pressure and velocity boundary conditions for the lattice Boltzmann BGK model, Phys. Fluids 9 (6) (1997) 1591–1597.
[19] A. Caiazzo, Asymptotic Analysis of lattice Boltzmann method for Fluid-Structure interaction problems, Ph.D. Thesis, Kaiserslautern (Germany) and Pisa (Italy), 2007, http://kluedo.ub.uni-kl.de/volltexte/2007/2079.
[20] A. Caiazzo, Analysis of lattice Boltzmann nodes initialisation in moving boundary problems, Prog. Comput. Fluid Dyn. 8 (2008) 3–10.
[21] D. Adalsteinsson, J.A. Sethian, The fast construction of extension velocities in level set methods, J. Comput. Phys. 148 (2) (1999) 2–22.
[22] D. Adalsteinsson, J.A. Sethian, A fast level set method for propagating interfaces, J. Comput. Phys. 118 (2) (1995) 269–277.
[23] S. Osher, J. Sethian, Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton–Jacobi formulations, J. Comput. Phys. 79 (1988) 12–49.
[24] S. Osher, R. Fedkiw, Level Set Methods and Dynamic Implicit Surfaces, Springer, New York, 2003.
[25] A.K. Vaikuntam, A. Wiegmann, Accurate estimation of surface parameters from the level set functions by a least squares approach, J. Scientific Comput., 2007, (submitted for publication).
[26] A. Smolianski, Finite-Element/Level-Set/Operator-Splitting (FELSOS) approach for computing two-fluid unsteady flows with free moving interfaces, Internat. J. Numer. Methods Fluids 48 (3) (2005) 231–269.
[27] M. Sussmann, A second order coupled level set and volume of fluid for computing growth and collapse of vapour bubbles, J. Comput. Phys. 187 (2003) 110–136.
[28] G. Thömmes, J. Becker, M. Junk, A.K. Vaikuntam, D. Kehrwald, A. Klar, K. Steiner, A. Wiegmann, A lattice Boltzmann method for immiscible multiphase flow simulations using the level set method, J. Comput. Phys. 228 (2009) 1139–1156.
[29] M. Junk, A. Klar, L.-S. Luo, Asymptotic analysis of the lattice Boltzmann equation, J. Comput. Phys. 210 (2) (2005) 676–704.
[30] M. Junk, Z. Yang, Asymptotic Analysis of lattice Boltzmann boundary conditions, J. Stat. Phys. 121 (2005) 3–35.
[31] Z. Yang, Analysis of lattice Boltzmann boundary conditions, Ph.D. Thesis, Konstanz, Germany, 2007.
[32] J. Casademunt, Viscous fingering as a paradigm of interfacial pattern formation: Recent results and new challenges, Chaos 14 (3) (1997) 809–824.
[33] D.A. Reinelt, P.G. Saffman, The penetration of a finger into a viscous fluid in channel and tube, SIAM J. Sci. Stat. Comput. 6 (3) (1985) 542–561.
[34] P.G. Saffman, G.I. Taylor, The penetration of a fluid into a porous medium or Hele–Shaw cell containing a more viscous liquid, Proc. Roy. Soc. A 245 (1958) 312–329.
[35] R. Clift, J.R. Grace, M.E. Weber, Bubbles Drops and Particles, Academic Press, 1978.
[36] I. Ginzburg, Lattice Boltzmann modelling with discontinuous collision components. Hydrodynamic and advection–diffusion equations, J. Stat. Phys. 126 (2007) 157–203.
[37] Y.H. Qian, D. D'humières, P. Lallemand, Lattice BGK models for the Navier Stokes equation, Europhys. Lett. 17 (1992) 479–484.