ELSEVIER

An adaptive scheme using hierarchical grids for lattice Boltzmann multi-phase flow simulations

Jonas Tölke *, Sören Freudiger, Manfred Krafczyk

Institut für Computeranwendungen im Bauingenieurwesen, TU Braunschweig, Germany

Abstract

The lattice Boltzmann (LB) method is extended and adapted to simulate multi-phase flows on non-uniform tree-type grids. Our model is an extension of the model developed by Gunstensen [Gunstensen AK, Rothman D. Lattice Boltzmann model of immiscible fluids. Phys Rev A 1991;43(8):4320–4327], which is based on the Rothman–Keller model [Rothman DH, Keller JM. Immiscible cellular automaton fluids. J Stat Phys 1988;52:1119–1127]. A first approach we use an a priori grid refinement. We find that the maximum number of possible grid levels for problems with dominant capillary forces is very restricted, if the physical interface is allowed to pass over grid interfaces. Thus a second approach based on adaptive grids was developed, where the physical interface is always discretized on the finest grid level. Efficient and flexible data structures have been developed to manage the remeshing. The application of the scheme for a rising bubble in three dimensions shows very good agreement with the semi-analytical solution and demonstrates the efficiency of our approach.
© 2005 Elsevier Ltd. All rights reserved.

1. Introduction

The efficient simulation of engineering problems governed by partial differential equations requires the use of non-uniform meshes and adaptive approaches. The lattice Boltzmann equation method has been successfully implemented for single-phase flows on block structured grids using a priori refinement [17,18] and on unstructured tree-type grids using a priori refinement [30] as well as adaptive approaches [14,21].

There are three existing approaches used in the simulation of immiscible multi-phase flows with LB methods: the Rothman–Keller model (RK) [1–3], the Shan–Chen model (SC) [4,6,8] and the free-energy approach (FE) [5]. The advantage of the RK-model is the independent adjustment of the surface tension, of the ratio of densities and of the ratio of viscosities. The advantage of the SC model is that high density ratios can be obtained and the disadvan-

tage is that the parameters cannot be independently adjusted. The drawback of the FE model is that the viscous terms in the resulting macroscopic equations are not Galilean invariant [7].

Macroscopic approaches are either based on the surface tracking method, where the interface is modeled explicitly or on a surface capturing method, where the interface is handled implicitly by using a phase field or an index function. An example of a surface tracking method based on a Lagrangian approach with marker particles and interpolating curves can be found in [9]. Surface capturing methods are very convenient for handling topological changes. Two examples are the volume of fluid method [10,29] and the level set method [11]. Furthermore a coupled version of the volume of fluid and level set method have been developed [28].

In this paper we adapt the Rothman–Keller type model introduced by Gunstensen for the LB method [2] for the simulation of immiscible two-phase flow on unstructured tree-type grids. The paper is organized as follows: In Section 2 we give a short review of the LB method on tree-type

---

* Corresponding author.
  E-mail address: toelke@cab.bau.tu-bs.de (J. Tölke).

grids for single-phase problems. In Section 3 we adapt the method for immiscible two-phase problems. In Section 4 we discuss some limitations and drawbacks of the approach on fixed grids. In Section 5 we propose an adaptive approach and in Section 6 our approach is validated for a rising bubble in three dimensions.

2. The LB Method for single-phase flow on non-uniform grids

We use a smoothed tree-type grid (quadtrees and octrees) allowing completely unstructured grids. Typical quadtrees are shown in Fig. 1. We use smoothed trees, meaning that the neighboring cells can only differ by at most one grid level, for the sake of algorithmic simplicity. In Fig. 1 a non-smoothed quadtree, a smoothed quadtree and a smoothed quadtree with a minimum width of three cells per grid level are shown. Regions can be refined locally with respect to arbitrarily shaped objects.

In the following section $\mathbf{x}$ represents a 3D vector in space and $\mathbf{f}$ a b-dimensional vector, where b is the number of microscopic velocities. The subsequent description is given in physical units and not in 'lattice' units. We discuss the $d3q19$ model [16] with the following microscopic velocities:

$$\{\mathbf{e}_i, i = 0, \ldots, 18\}$$
$$= \left\{ \begin{matrix} 0 & c & -c & 0 & 0 & 0 & 0 & c & -c & c & -c & c & -c & c & -c & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & c & -c & 0 & 0 & c & -c & -c & c & 0 & 0 & 0 & 0 & c & -c & c & -c \\ 0 & 0 & 0 & 0 & 0 & c & -c & 0 & 0 & 0 & 0 & c & -c & -c & c & c & -c & -c & c \end{matrix} \right\},$$

where c is a constant velocity determining the speed of sound. We use $c_s^2 = c^2/3$ in this paper. The lattice Boltzmann equation using the multi-relaxation time model [25,26] for the grid level l is then given by

$$f_{i,l}(t + \Delta t_l, \mathbf{x} + \mathbf{e}_i \Delta t_l) = f_{i,l}(t, \mathbf{x}) + \Omega_{i,l}, \quad i = 0, \ldots, b-1, \tag{1}$$

where Δt_l is the time step and the collision operator on grid level l is given by

$$\mathbf{\Omega}_l = \mathsf{M}^{-1} \mathsf{S}_l [(\mathsf{M}\mathbf{f}) - \mathbf{m}^{eq}]. \tag{2}$$

Matrix M is the transformation matrix given in Appendix B, composed of the 19 orthogonal basis vectors $\{\mathbf{\Phi}_i, i = 0, \ldots, b-1\}$ given in Appendix A. They are orthogonal with respect to the inner product $\langle \mathbf{\Phi}_i, \mathbf{\Phi}_j w \rangle$ (in contrast to

[27], where $\langle \mathbf{\Phi}_i, \mathbf{\Phi}_j \rangle = 0$, if $i \neq j$). The vector $\mathbf{w}$ is composed of the weights $\{w_i, i = 0, \ldots, b-1\}$:

$$\mathbf{w} = \left(\frac{1}{3}, \frac{1}{18}, \frac{1}{18}, \frac{1}{18}, \frac{1}{18}, \frac{1}{18}, \frac{1}{18}, \frac{1}{36}, \frac{1}{36}, \frac{1}{36}, \frac{1}{36}, \frac{1}{36}, \frac{1}{36}, \frac{1}{36}, \frac{1}{36}, \frac{1}{36}, \frac{1}{36}, \frac{1}{36}, \frac{1}{36} \right).$$

The moments $\mathbf{m} = \mathsf{M}\mathbf{f}$ are labeled as

$$\mathbf{m} = (\rho, e, \epsilon, j_x, q_x, j_y, q_y, j_z, q_z, 3p_{xx}, 3\pi_{xx}, p_{ww}, \pi_{ww}, p_{xy}, p_{yz}, p_{xz}, m_x, m_y, m_z).$$

$\mathbf{m}^{eq}$ is the vector composed of the equilibrium moments given in Eqs. (4) and $\mathsf{S}_l = \{s_{l,i,i}, i = 0, \ldots, b-1\}$ is the diagonal collision matrix. The non-zero collision parameters $s_{l,i,i}$ (the eigenvalues of the collision matrix $\mathsf{M}^{-1}\mathsf{S}_l\mathsf{M}$) are

$$s_{l,1,1} = s_{l,a},$$
$$s_{l,2,2} = s_{l,b},$$
$$s_{l,4,4} = s_{l,6,6} = s_{l,8,8} = s_{l,c},$$
$$s_{l,10,10} = s_{l,12,12} = s_{l,d},$$
$$s_{l,9,9} = s_{l,11,11} = s_{l,13,13} = s_{l,14,14} = s_{l,15,15} = -\frac{\Delta t_l}{\tau_l} = s_{l,\omega},$$
$$s_{l,16,16} = s_{l,17,17} = s_{l,18,18} = s_{l,e}.$$

The relaxation time τ_l is

$$\tau_l = 3\frac{v}{c^2} + \frac{1}{2}\Delta t_l, \tag{3}$$

where v is the kinematic viscosity. The parameters s_a, s_b, s_c, s_d and s_e can be freely chosen in the range $[-2, 0]$ and tuned to improve stability [26]. The optimal values depend on the specific system under consideration (geometry, initial and boundary conditions) and cannot be computed in advance. Some reasonable values for these parameters are given in [27]. We choose $s_a = s_b = s_c = s_d = s_e = \max\{s_{l,\omega}, -1.0\}$ on the finest grid level. The non-zero equilibrium distribution functions $\{m_i^{eq}, i = 0, \ldots, 19\}$ are given by

$$m_0^{eq} = \rho, \quad m_3^{eq} = \rho_0 u_x, \quad m_5^{eq} = \rho_0 u_y, \quad m_7^{eq} = \rho_0 u_z, \tag{4a}$$
$$m_1^{eq} = e^{eq} = \rho_0 \left(u_x^2 + u_y^2 + u_z^2 \right), \tag{4b}$$
$$m_9^{eq} = 3p_{xx}^{eq} = \rho_0 \left(2u_x^2 - u_y^2 - u_z^2 \right),$$
$$m_{11}^{eq} = p_{zz}^{eq} = \rho_0 \left(u_y^2 - u_z^2 \right), \tag{4c}$$
$$m_{13}^{eq} = p_{xy}^{eq} = \rho_0 u_x u_y, \quad m_{14}^{eq} = p_{yz}^{eq} = \rho_0 u_y u_z,$$
$$m_{15}^{eq} = p_{xz}^{eq} = \rho_0 u_x u_z, \tag{4d}$$

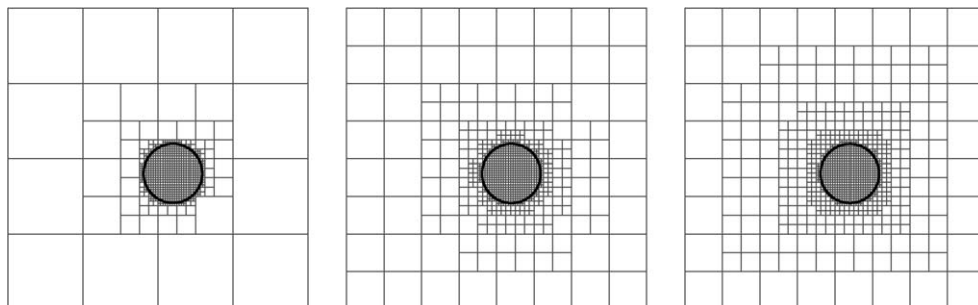where ρ_0 is a constant density and ρ a density variation.

Fig. 1. Quadtree, smoothed quadtree and quadtree with minimum width of three cells.
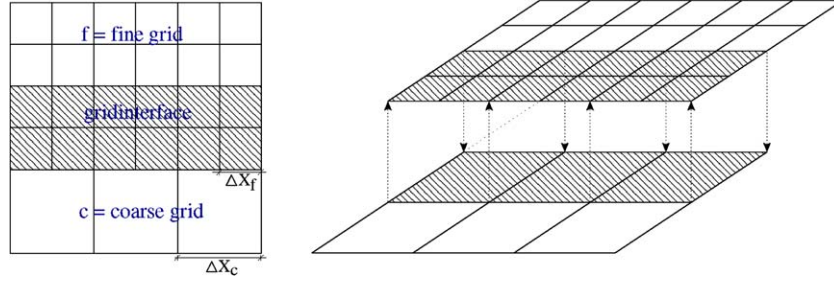
Fig. 2. Grid-interface and scale directions.

We use nested time-stepping: if the speed of sound and the physical viscosity are assumed to be equal on all grid levels, with Mach and Reynolds number equal on all grids, then two time steps on the finer grid have to be done during one time step on the coarser mesh [17]. An approach where also the Mach number is lowered on finer grid levels to ensure a faster convergence to the *incompressible* Navier–Stokes equations can be found in [22]. This approach requires one to do four time steps on the finer grid during one time step on the coarser mesh.

The macroscopic quantities density and momentum and the stress tensor are given by (omitting the index l)

$$\rho = \sum_i f_i, \tag{5a}$$

$$\rho_0 \mathbf{u} = \sum_i \mathbf{e}_i f_i, \tag{5b}$$

$$S_{xx} = -\left(1 - \frac{\Delta t}{2\tau}\right)\left(\frac{1}{3}e + p_{xx} - \rho_0 u_x^2\right), \tag{5c}$$

$$S_{yy} = -\left(1 - \frac{\Delta t}{2\tau}\right)\left(\frac{1}{3}e - \frac{1}{2}p_{xx} + \frac{1}{2}p_{ww} - \rho_0 u_y^2\right), \tag{5d}$$

$$S_{zz} = -\left(1 - \frac{\Delta t}{2\tau}\right)\left(\frac{1}{3}e - \frac{1}{2}p_{xx} - \frac{1}{2}p_{ww} - \rho_0 u_z^2\right), \tag{5e}$$

$$S_{xy} = -\left(1 - \frac{\Delta t}{2\tau}\right)(p_{xy} - \rho_0 u_x u_y), \tag{5f}$$

$$S_{yz} = -\left(1 - \frac{\Delta t}{2\tau}\right)(p_{yz} - \rho_0 u_y u_z), \tag{5g}$$

$$S_{xz} = -\left(1 - \frac{\Delta t}{2\tau}\right)(p_{xz} - \rho_0 u_x u_z). \tag{5h}$$

Using an asymptotic and Taylor expansion, it can be shown that the LB method is a scheme of second order both in space and time for the compressible Navier–Stokes equations in the low Mach number limit using the advective scaling and a scheme of first order in time and second order in space for the incompressible Navier–Stokes equations using the diffusive scaling [19].

A typical grid interface is shown in Fig. 2. The overlap of the interface is due to the fact, that missing distributions on one grid level have to be computed from the adjacent grid level. Ensuring the continuity of the pressure, the velocity, and also of their derivatives, we have to rescale the non-equilibrium part f^{neq} [17,18,21] or alternatively $\mathbf{m}^{\text{neq}}$. Doing the rescaling after the propagation step we obtain
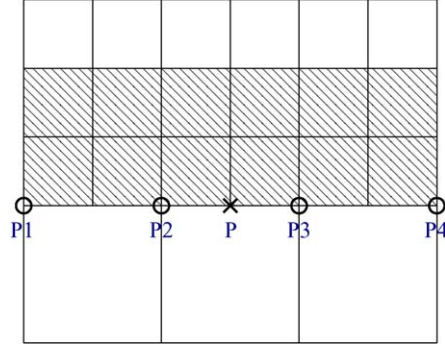

Fig. 3. Hanging node P and nodes P1–P4 required for interpolation.

$$m_{i,l-1}^{\text{neq}} = \frac{s_{l,i,i}}{s_{l-1,i,i}} \frac{\Delta t_{l-1}}{\Delta t_l} m_{i,l}^{\text{neq}}. \tag{6}$$

The relaxation parameters $s_{l,a} = s_{l,b} = s_{l,c} = s_{l,d} = s_{l,e}$ can be chosen in principle arbitrarily in the range $[-2, 0]$, but we use the same scaling as for $s_{l,\omega}$.

$$\frac{s_{l,a}}{s_{l+1,a}} = \frac{s_{l,b}}{s_{l+1,b}} = \frac{s_{l,c}}{s_{l+1,c}} = \frac{s_{l,d}}{s_{l+1,d}} = \frac{s_{l,e}}{s_{l+1,e}} = \frac{s_{l,\omega}}{s_{l+1,\omega}}. \tag{7}$$

We have to use space and time interpolations because of the different mesh spacings Δx_l and time steps Δt_l. A cubic interpolation in space is used for the 'hanging' nodes. Fig. 3 shows the interpolation for the 'hanging' node P using the nodes P1, P2, P3 and P4. Linear interpolation is used in time. Details of the algorithm can be found in [13,14,18].

3. Rothman–Keller model on tree-type grids

The Gunstensen model [2] is based on a phase field approach. The collision operator is extended by terms responsible for the generation of surface tension. The advection of the phase field is done in the lattice Boltzmann framework and a recoloring step is introduced to keep the interface sharp.

We introduce a red and blue density field ρ_r and ρ_b and define an order parameter ϕ:

$$\phi = \frac{\rho_r - \rho_b}{\rho_r + \rho_b} \tag{8}$$

The gradient $\mathbf{C}$ of the phase field is computed as

$$\mathbf{C}_l(t, \mathbf{x}) = \frac{3}{c^2 \Delta t_l} \sum_i w_i \mathbf{e}_i \phi(t, \mathbf{x} + \mathbf{e}_i \Delta t_l). \tag{9}$$

The normalized gradient is

$$n_{l,\alpha} = \frac{C_{l,\alpha}}{|\mathbf{C}_l|}. \tag{10}$$

The equilibrium distribution functions contain additional terms to generate surface tension through a perturbation based on the gradient of the phase field. The altered equilibrium distribution functions $f^{\,\mathrm{eq}}$ can be found in [15]. The additional contributions to the equilibrium moments given in Eqs. (4) are

$$m_1^{\mathrm{st},l} = -\sigma|\mathbf{C}_l|\left(n_{l,x}^2 + n_{l,y}^2 + n_{l,z}^2\right) = \sigma|\mathbf{C}_l|, \tag{11a}$$

$$m_9^{\mathrm{st},l} = \frac{1}{2}\sigma|\mathbf{C}_l|\left(2n_{l,x}^2 - n_{l,y}^2 - n_{l,z}^2\right), \tag{11b}$$

$$m_{11}^{\mathrm{st},l} = \frac{1}{2}\sigma|\mathbf{C}_l|\left(n_{l,y}^2 - n_{l,z}^2\right), \tag{11c}$$

$$m_{13}^{\mathrm{st},l} = \frac{1}{2}\sigma|\mathbf{C}_l|(n_{l,x}n_{l,y}), \tag{11d}$$

$$m_{14}^{\mathrm{st},l} = \frac{1}{2}\sigma|\mathbf{C}_l|(n_{l,y}n_{l,z}), \tag{11e}$$

$$m_{15}^{\mathrm{st},l} = \frac{1}{2}\sigma|\mathbf{C}_l|(n_{l,x}n_{l,z}), \tag{11f}$$

where σ is the surface tension. Note that due to the gradient $\mathbf{C}_l$ the collision is not local anymore. The additional contributions from the surface tension (st) to the pressure tensor are

$$P_{xx}^{\mathrm{st}} = \sigma|\mathbf{C}_l|\left(n_{l,y}^2 + n_{l,z}^2\right), \tag{12a}$$

$$P_{yy}^{\mathrm{st}} = \sigma|\mathbf{C}_l|\left(n_{l,x}^2 + n_{l,z}^2\right), \tag{12b}$$

$$P_{zz}^{\mathrm{st}} = \sigma|\mathbf{C}_l|\left(n_{l,x}^2 + n_{l,y}^2\right), \tag{12c}$$

$$P_{xy}^{\mathrm{st}} = -\sigma|\mathbf{C}_l|(n_{l,x}n_{l,y}), \tag{12d}$$

$$P_{yz}^{\mathrm{st}} = -\sigma|\mathbf{C}_l|(n_{l,y}n_{l,z}), \tag{12e}$$

$$P_{xz}^{\mathrm{st}} = -\sigma|\mathbf{C}_l|(n_{l,x}n_{l,z}). \tag{12f}$$

The advection of an arbitrary scalar phase field ψ, where $\psi = \{\rho_{\mathrm{r}}, \rho_{\mathrm{b}}\}$, is done with a separate LB equation:

$$g_{i,l}(t + \Delta t_l, \mathbf{x} + \mathbf{e}_i\Delta t_l) = g_{i,l}(t, \mathbf{x}) - \frac{\Delta t_l}{\tau_l}\big(g_{i,l}(t, \mathbf{x})$$
$$- g_i^{\mathrm{eq}}(\psi(t, \mathbf{x}), \mathbf{u}(t, \mathbf{x}))\big). \tag{13}$$

Setting $\frac{\Delta t_l}{\tau_l} = 1$ gives the following simplified equation:

$$g_{i,l}(t + \Delta t_l, \mathbf{x} + \mathbf{e}_i\Delta t_l) = g_i^{\mathrm{eq}}(\psi(t, \mathbf{x}), \mathbf{u}(t, \mathbf{x})). \tag{14}$$

The equilibrium distribution function is given by

$$g_i^{\mathrm{eq}}(\psi, \mathbf{u}) = w_i\psi\left(1 + \frac{3}{c^2}\mathbf{e}_i \cdot \mathbf{u}\right). \tag{15}$$

The scheme (14) in combination with (15) results in an advection diffusion equation [15]. By using $\frac{\Delta t_l}{\tau_l} = 1$ on all grid levels we have a different diffusion coefficient on each grid level which is annihilated by the recoloring scheme. Note that for the scheme given by (14) we only need the value ψ of the phase field and the advection velocity $\mathbf{u}$ as input values. So for the multi-phase extension we need to store only the two additional variables ρ_{r} and ρ_{b}. A recoloring step is introduced to eliminate the diffusion effects and to achieve a phase separation The recoloring step redistribute the mass distributions of phase ρ_{r} and ρ_{b} in a way so that the inner product of the gradient $\mathbf{C}$ and momentum of phase ρ_{r} is maximized. The constraints are the conservation of the mass of each phase and the conservation of the momentum of the sum of both phases. We use the recoloring algorithm given in [12,20]. A detailed mathematical analysis shows that this numerical model yields the Navier–Stokes equation for two immiscible phases with surface tension [15]. Note that the extensions to the original method [2], especially the linear advection scheme, the moment method and the altered terms of Eqs. (11a)–(11f) for imposing surface tension, improve the numerical efficiency drastically. The difference between the algorithm for single-phase and multi-phase flows on non-uniform grids are the following:

- introduction of two fields ρ_{r} and ρ_{b},
- generation of surface tension by *non-local* operations (gradient of a phase field),
- advection of fields ρ_{r} and ρ_{b} with a separate LB equation,
- recoloring to keep the interface sharp,
- additional time and space interpolations to compute the gradient $\mathbf{C}_l$ at the grid interface.

Computing the color gradient at the grid interface requires additional space and time interpolations for the density (see Fig. 4). A flowchart of the algorithm is given in Fig. 5. We used

- constant, linear and quadratic time interpolation of ϕ,
- different stencils to compute the gradient $\mathbf{C}$ at the grid interface.

A linear scheme both in space and time for the interpolation of ϕ yielded satisfactory results, but no improvements were observed when higher order schemes were used. Depending on the values of the surface tension, perturbations occurred when the physical interface passed the grid
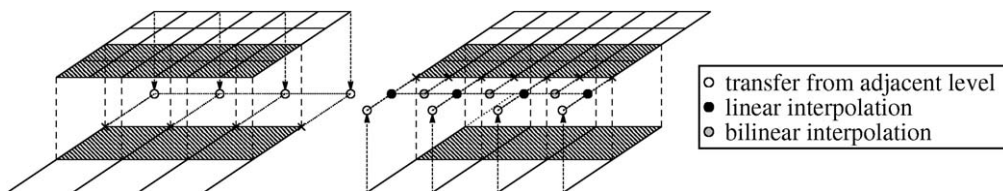

Fig. 4. Interpolation of ϕ for the computation of the gradient $\mathbf{C}$ at the grid interface.

○ transfer from adjacent level
● linear interpolation
◉ bilinear interpolation

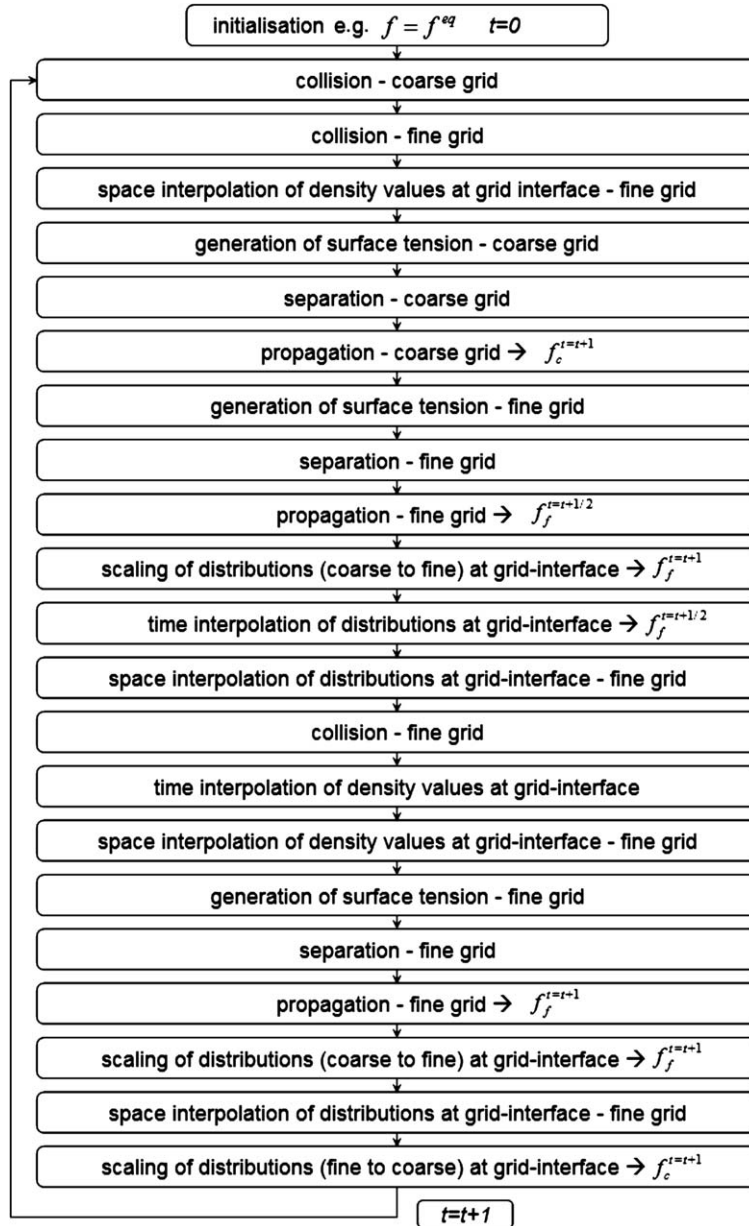| initialisation e.g. $f = f^{eq}$　　$t=0$ |
| collision - coarse grid |
| collision - fine grid |
| space interpolation of density values at grid interface - fine grid |
| generation of surface tension - coarse grid |
| separation - coarse grid |
| propagation - coarse grid → $f_c^{t=t+1}$ |
| generation of surface tension - fine grid |
| separation - fine grid |
| propagation - fine grid → $f_f^{t=t+1/2}$ |
| scaling of distributions (coarse to fine) at grid-interface → $f_f^{t=t+1}$ |
| time interpolation of distributions at grid-interface → $f_f^{t=t+1/2}$ |
| space interpolation of distributions at grid-interface - fine grid |
| collision - fine grid |
| time interpolation of density values at grid-interface |
| space interpolation of density values at grid-interface - fine grid |
| generation of surface tension - fine grid |
| separation - fine grid |
| propagation - fine grid → $f_f^{t=t+1}$ |
| scaling of distributions (coarse to fine) at grid-interface → $f_f^{t=t+1}$ |
| space interpolation of distributions at grid-interface - fine grid |
| scaling of distributions (fine to coarse) at grid-interface → $f_c^{t=t+1}$ |
| $t=t+1$ |

Fig. 5. Flowchart of the algorithm for two grid levels.

interface. Small perturbations were introduced when the surface tension vanished (pure advection of a phase field) causing the shape of the phase field to be slightly distorted. Strong surface tension introduced spurious capillary forces.

4. Some remarks concerning the approach with fixed grids

4.1. Interface width

The width of the physical interface is always $d \cdot \Delta x_l$, where the constant d depends on the recoloring algorithm but is independent of the grid level. The integration of the difference of the normal and tangential component of the pressure tensor over the interface yields the same value of the surface tension for each grid level, but the different

physical widths of the interface on different grid levels lead to perturbations (see Fig. 10). If the physical interface passes the grid interface, then the gradient **C** is different on the coarse and fine grid due to the wider physical interface on the coarse grid level (see Fig. 6).

4.2. Maximum number of grid levels

The collision factor ω is defined as

$$\omega = \frac{\Delta t}{\tau}, \quad \Delta t = \Delta t_{\text{ref}}(1/2)^{l-l_0}, \quad \tau = 3\frac{v}{c^2} + \frac{1}{2}\Delta t, \tag{16}$$

where l_0 is the coarsest grid level. Thus we can compute ω in terms of the grid level l, the coarsest grid level l_0 and the kinematic viscosity v as follows [14]:
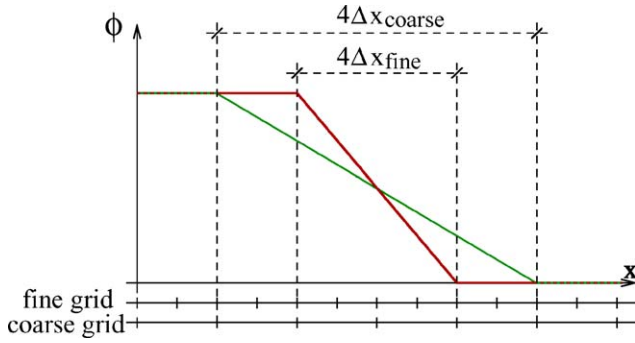
Fig. 6. Interface width (phase field) for coarse and fine grid level.

$$\omega = \frac{\Delta t}{\tau} = \frac{\left(\frac{1}{2}\right)^{l-l_0}}{3\frac{v}{\Delta t_{\text{ref}} c^2} + \left(\frac{1}{2}\right)^{l-l_0+1}}. \tag{17}$$

Solving for l gives the following relation:

$$l = l_0 - \log_2 \frac{6v\omega}{\Delta t_{\text{ref}} c^2 (2-\omega)}. \tag{18}$$

Due to stability restrictions and convergence issues (Knudsen number), the collision factor is limited to a range

$$\omega_{\text{lower}} < \omega < \omega_{\text{upper}}, \tag{19}$$

where ω_{lower} is the limit due to the requirement of small Knudsen numbers and ω_{upper} is the limit due to stability restrictions. The values ω_{lower} and ω_{upper} for typical simulations are: single time relaxation approximation (BGK) $\omega_{\text{lower}} = \frac{2}{3}$ and $\omega_{\text{upper}} \approx 1.976$, and multi-time relaxation approximation [26,27] $\omega_{\text{lower}} = \frac{2}{3}$ and $\omega_{\text{upper}} \approx 1.999$. These limits are approximate values and differ for specific flow problems. Eq. (18) combined with the values for ω_{lower} and ω_{upper}, the maximum number of levels can be computed as functions of the viscosity. Fig. 7 shows the relation for the MRT, resulting in a maximum of 11 grid levels. For multi-phase problems with *strong* capillary forces due to high surface tension the limits are much more severe: the collision factor has to be in the range $0.8 < \omega < 1.5$. These limits result in a restriction to only three grid levels (see Fig. 8 and Table 1), so that the possibilities for an appropriate grid refinement are severely restricted.

5. Simulation of multi-phase flows using adaptive, nested tree-type data structures

The basic idea of the adaptive approach is to resolve the physical interface always on the finest grid level. This is reasonable, since the strongest variations of the physical
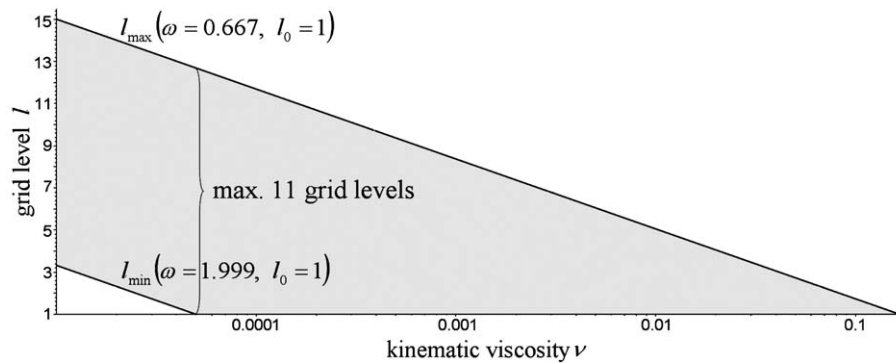

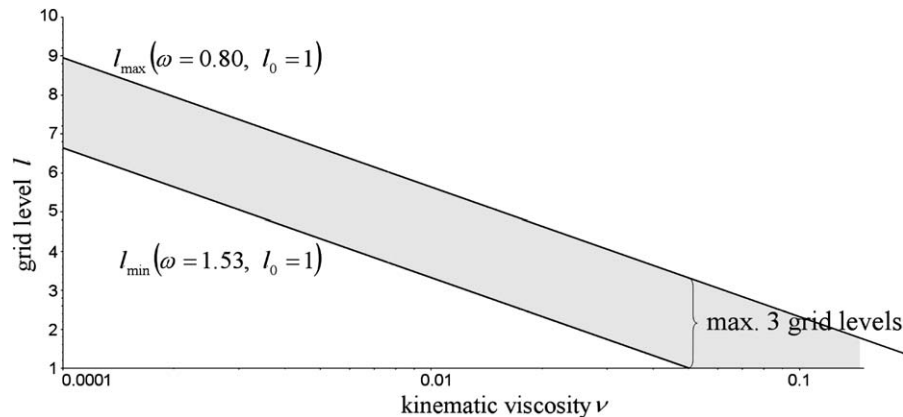Fig. 7. Maximum number of grid levels for single-phase problems (MRT).

Fig. 8. Maximum number of grid levels for multi-phase problems with strong capillary forces.

Table 1
Maximum number of grid levels

Method	ω_{lower}	ω_{upper}	Maximum number of grid levels
SRT-SP	$\frac{2}{3}$	≈ 1.976	≈ 7
MRT-SP	$\frac{2}{3}$	≈ 1.999	≈ 11
SRT-MP	≈ 0.8	≈ 1.5	3

quantities are in the region of the physical interface. The adaptation of the grid is done after each coarse time step by collapsing and expanding nodes using appropriate indicators based on the value of the phase field and/or its gradient. Ensuring that the moving physical interface is always kept on the finest grid level, the neighborhood of the phys-
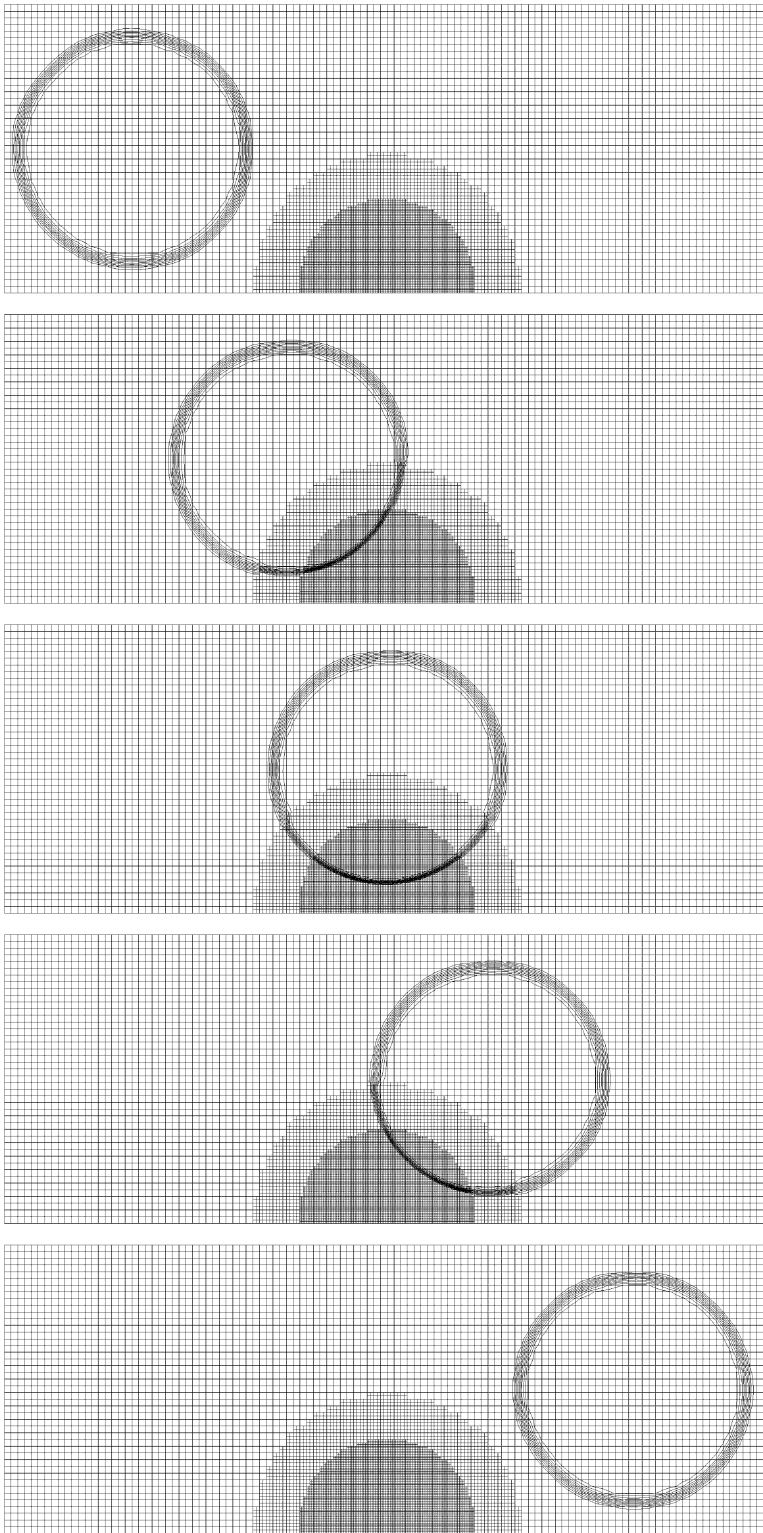

Fig. 9. Isolines of the phase field of a single bubble in an initially uniform flow field for $\sigma = 0.01$ (N/m).

ical interface has to be discretized by a sufficiently wide layer of fine grid cells. Suppose for example we use seven grid levels, we have 64 time steps on the finest grid level during one time step on the coarsest grid level. If we choose $\Delta t = 1$ s and $\Delta x = 1$ m on the finest grid level and the physical interface moves with 0.05 m/s, it can pass $0.05 \cdot 64 = 3.2$ fine grid cells in one coarse time step. So we have to ensure, that a layer of four fine grid cells is present in any direction from the interface. The maximum number of possible grid levels with this approach is almost the same as for single-phase problems, since the restriction $0.8 < \omega < 1.5$ is only valid on the finest grid level, although on the other grid levels $0.667 < \omega < 1.999$ holds. The implementation was done in 2D using the $d2q9$ model and in 3D using the $d3q19$ model [16] based on tree-type grids.

6. Numerical examples

6.1. Two-dimensional bubble on a fixed nested grid in an initially uniform flow field

The motivation of this section is to qualitatively show the distortions introduced by the algorithm if a bubble passes different grid levels. A part of the simulation domain and the grid is shown in Fig. 9. A bubble (represented by isolines of the phase field) in an initially uniform flow field is advected over a locally refined grid with three different grid levels. The boundary conditions are periodic in both directions (the lower half of the domain is missing in the figures). The bubble passes the grid interfaces with no visible distortions in this parameter range. The grid dependent width of the interface can be clearly seen. Fig. 10 shows the shape of the bubble after passing the refined zone for three different values of surface tension. The first figure shows the resulting shape from the previous case $\sigma = 0.01$ (N/m), the second one shows the resulting shape for no surface tension $\sigma = 0$ (pure advection of the phase field), where the third one shows the resulting shape for strong surface tension $\sigma = 0.1$ N/m.

6.2. Simulation of a three-dimensional rising bubble using an adaptive grid

Extensive theoretical and experimental work on rising bubbles is found in the books of Clift et al. [24] and Zapryanov and Tabakova [23]. The case of the free rising
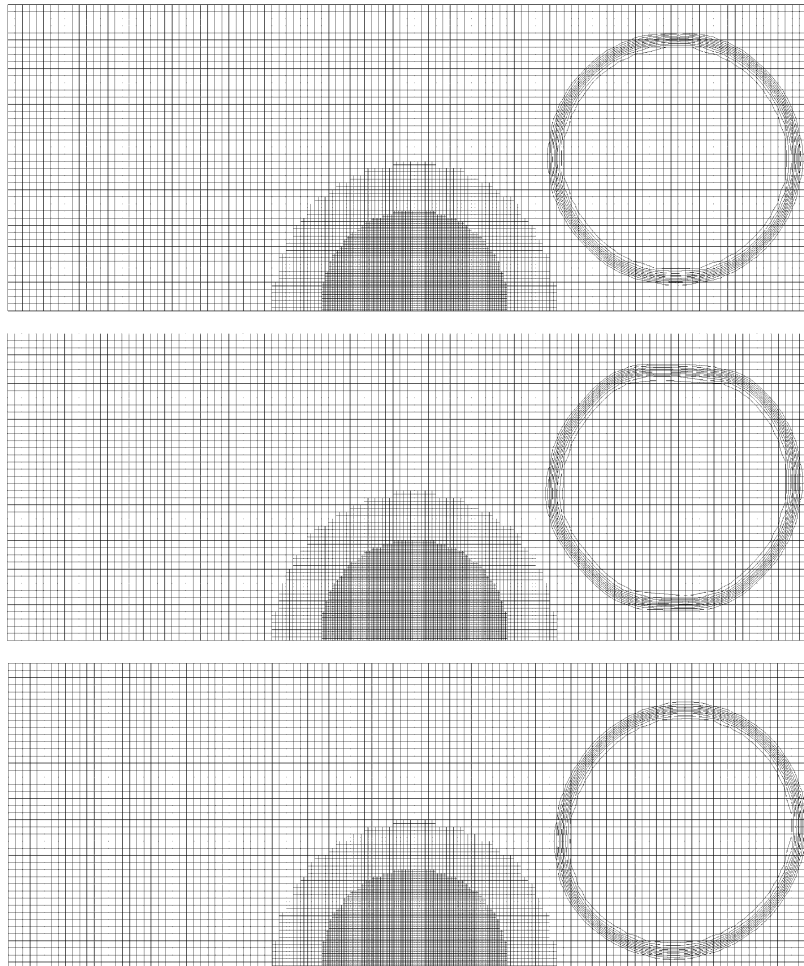

Fig. 10. Isolines of the phase field of a single bubble in a uniform flow field after passing the grid interface for $\sigma = 0.01$, 0.0, and 0.1 (N/m).

Eo	M	Re	U_t	U_{num}
6062.5	202939	19.4	0.050	0.048

bubble in an infinite medium, the bubble motion can be characterized by the viscosity ratio γ, the Eötvös number Eo, the Morton number M and the Reynolds number Re [24]:

$$\gamma = \frac{\mu_i}{\mu}, \quad Eo = \frac{g\Delta\rho d^2}{\sigma}, \quad M = \frac{g\mu^4\Delta\rho}{\rho^2\sigma^3}, \quad Re = \frac{U_T d\rho}{\mu},$$

(20)

where μ_i and μ are the dynamic viscosity of the inner and outer phase, $\Delta\rho$ the density difference, d the sphere diameter, ρ the density of the outer phase, U_T the terminal velocity, and g the acceleration due to gravity. The shape of the bubble can be spherical, ellipsoidal, "skirted", "dimpled" or a "spherical cap", depending on these parameters.

When $Eo > 40$ and $M > 200$ the terminal velocity of a rising bubble can be computed by solving the following equation for the Reynolds number Re [24]:

$$2Re^2 + 6Re\frac{2 + 3\gamma}{1 + \gamma} - Eo^{\frac{3}{2}}M^{-\frac{1}{2}} = 0.$$

We used the parameters given in Table 2 and six grid levels for the simulation. The initial setup, the resulting shape of the bubble, the velocity vectors and the grid are shown in Fig. 11. A comparison of the theoretical terminal velocity U_T and the velocity U_{num} resulting from the simulation shows a good agreement (see Table 2). The number of grid nodes for this simulation was about 660,000, whereas using a uniform mesh with the finest resolution would result in a
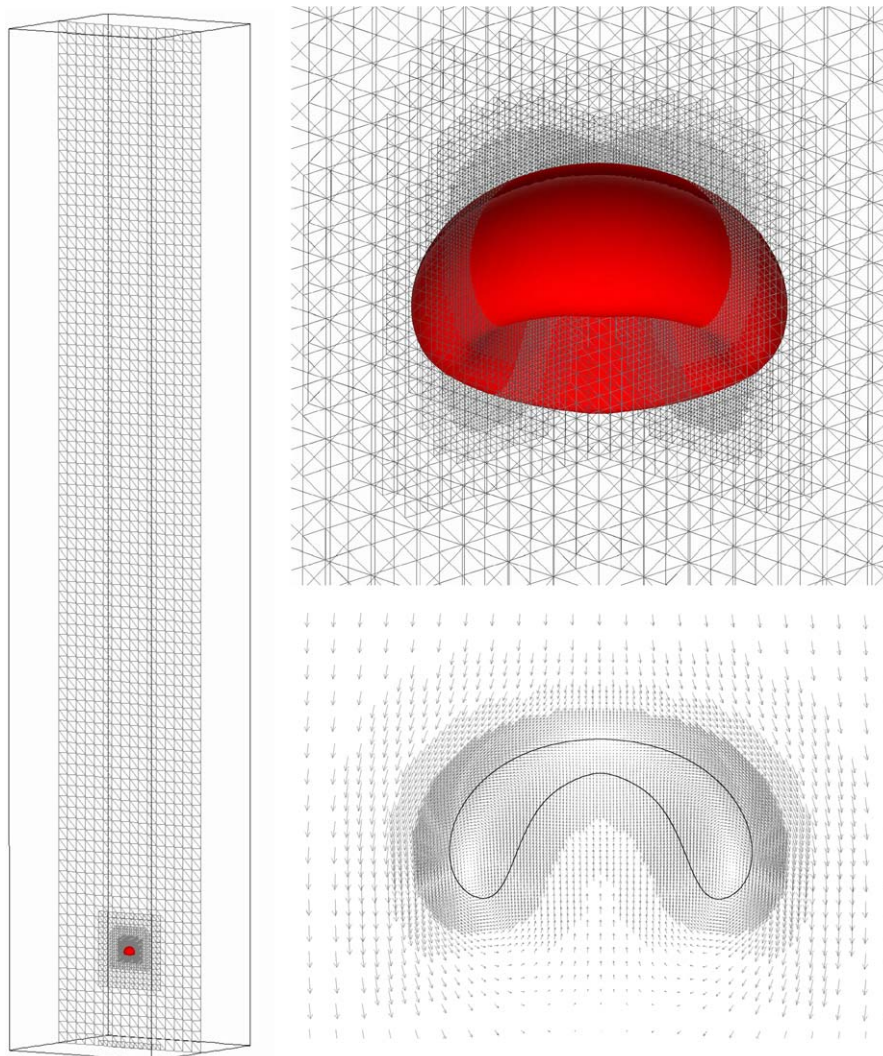

Fig. 11. Rising bubble, simulation domain (left, six grid levels), plots of the shape (right, top) and the velocity vectors (right, bottom).

mesh with 6.5 billion nodes and correspondingly larger CPU times by a factor of $O(10^4)$.

7. Conclusions and outlook

The simulation of multi-phase problems using the LB method with a priori refined and fixed grids works in a limited parameter range. The number of possible grid levels is severely restricted when used on problems with strong capillary forces. These limitations are due to the lower stability when a physical interface passes a grid interface. These problems are eliminated by using an adaptive approach, where the physical interface is always resolved on the finest grid level and the remeshing follows the physical interface. Both approaches resolve the physical interface explicitly resulting in strong gradients. Another possibility is to reconstruct the physical interface locally using appropriate algorithms to compute the approximate curvature and to couple both phases through appropriate boundary conditions (inducing the correct pressure jump). First investigations in this direction show very encouraging results and will be published in the near future.

Acknowledgments

Appendix A. Orthogonal basis vectors $\{\Phi_i, i = 0,\ldots,b-1\}$

$$\Phi_{0,\alpha} = n1, \quad \Phi_{1,\alpha} = \mathbf{e}_\alpha^2 - c^2,$$

$$\Phi_{2,\alpha} = 3(\mathbf{e}_\alpha^2)^2 - 6\mathbf{e}_\alpha^2 c^2 + c^4, \tag{A.1}$$

$$\Phi_{3,\alpha} = e_{\alpha x}, \quad \Phi_{5,\alpha} = e_{\alpha y}, \quad \Phi_{7,\alpha} = e_{\alpha z}, \tag{A.2}$$

$$\Phi_{4,\alpha} = (3\mathbf{e}_\alpha^2 - 5c^2)e_{\alpha x}, \quad \Phi_{6,\alpha} = (3\mathbf{e}_\alpha^2 - 5c^2)e_{\alpha y},$$

$$\Phi_{8,\alpha} = (3\mathbf{e}_\alpha^2 - 5c^2)e_{\alpha z}, \tag{A.3}$$

$$\Phi_{9,\alpha} = 3e_{\alpha x}^2 - \mathbf{e}_\alpha^2, \quad \Phi_{11,\alpha} = e_{\alpha y}^2 - e_{\alpha z}^2, \tag{A.4}$$

$$\Phi_{13,\alpha} = e_{\alpha x}e_{\alpha y}, \quad \Phi_{14,\alpha} = e_{\alpha y}e_{\alpha z}, \quad \Phi_{15,\alpha} = e_{\alpha x}e_{\alpha z}, \tag{A.5}$$

$$\Phi_{10,\alpha} = (2\mathbf{e}_\alpha^2 - 3c^2)(3e_{\alpha x}^2 - \mathbf{e}_\alpha^2), \quad \Phi_{12,\alpha} = (2\mathbf{e}_\alpha^2 - 3c^2)(e_{\alpha y}^2 - e_{\alpha z}^2), \tag{A.6}$$

$$\Phi_{16,\alpha} = (e_{\alpha y}^2 - e_{\alpha z}^2)e_{\alpha x}, \quad \Phi_{17,\alpha} = (e_{\alpha z}^2 - e_{\alpha x}^2)e_{\alpha y},$$

$$\Phi_{18,\alpha} = (e_{\alpha x}^2 - e_{\alpha y}^2)e_{\alpha z}. \tag{A.7}$$

Appendix B. Transformation matrix M

$$
\begin{bmatrix}
1\cdot & (1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1) \\
c^2\cdot & (-1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1) \\
c^4\cdot & (1 & -2 & -2 & -2 & -2 & -2 & -2 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1) \\
c\cdot & (0 & 1 & -1 & 0 & 0 & 0 & 0 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 0 & 0 & 0 & 0) \\
c^3\cdot & (0 & -2 & 2 & 0 & 0 & 0 & 0 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 0 & 0 & 0 & 0) \\
c\cdot & (0 & 0 & 0 & 1 & -1 & 0 & 0 & 1 & -1 & -1 & 1 & 0 & 0 & 0 & 0 & 1 & -1 & 1 & -1) \\
c^3\cdot & (0 & 0 & 0 & -2 & 2 & 0 & 0 & 1 & -1 & -1 & 1 & 0 & 0 & 0 & 0 & 1 & -1 & 1 & -1) \\
c\cdot & (0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1) \\
c^3\cdot & (0 & 0 & 0 & 0 & 0 & -2 & 2 & 0 & 0 & 0 & 0 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1) \\
c^2\cdot & (0 & 2 & 2 & -1 & -1 & -1 & -1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & -2 & -2 & -2 & -2) \\
c^4\cdot & (0 & -2 & -2 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & -2 & -2 & -2 & -2) \\
c^2\cdot & (0 & 0 & 0 & 1 & 1 & -1 & -1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & 0 & 0 & 0 & 0) \\
c^4\cdot & (0 & 0 & 0 & -1 & -1 & 1 & 1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & 0 & 0 & 0 & 0) \\
c^2\cdot & (0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0) \\
c^2\cdot & (0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & -1 & -1 & 0 & 0 & 0 & 0) \\
c^2\cdot & (0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & -1 & -1) \\
c^3\cdot & (0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 0 & 0 & 0 & 0) \\
c^3\cdot & (0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 1 & -1 & 0 & 0 & 0 & 0 & 1 & -1 & 1 & -1) \\
c^3\cdot & (0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1)
\end{bmatrix}.
$$

References

[1] Rothman DH, Keller JM. Immiscible cellular automaton fluids. J Stat Phys 1988;52:1119–27.

[2] Gunstensen AK, Rothman D. Lattice Boltzmann model of immiscible fluids. Phys Rev A 1991;43(8):4320–7.

[3] Grunau D, Chen S, Eggert K. A lattice Boltzmann model for multiphase flow. Phys Fluids A 1993;5(10):2557–61.

[4] Schelkle M. LB-Verfahren zur Simulation dreidimensionaler Zweiphasen-Strömungen mit freien Oberflächen. PhD thesis, Univ. Stuttgart, 1996.

[5] Swift MR, Orlandini E, Osborn WR, Yeomans JM. Lattice Boltzmann simulation of liquid–gas and binary fluid systems. Phys Rev E 1996;54:5041–52.

[6] Shan X, Chen H. Lattice Boltzmann model for simulating flows with multiple phases and components. Phys Rev E 1993;47:1815–9.

[7] Luo L-S. Unified theory of the lattice Boltzmann models for nonideal gases. Phys Rev Lett 1998;81(8):1618–21.

[8] Inamuro T, Ogata T, Ogino F. Numerical simulation of bubble flows by the lattice Boltzmann method. FGCS 2004;20(6):959–64.

[9] Popinet S, Zaleski S. A front-tracking algorithm for accurate representation of surface tension. Int J Numer Meth Fluids 1999;30:775–93.

[10] Hirt C, Nichols B. Volume of fluid method for the dynamics of free boundaries. J Comput Phys 1981;10:201–25.

[11] Osher S, Fedkiw R. Level set methods and dynamic implicit surfaces. New York: Springer; 2003.

[12] Tölke J. Gitter-Boltzmann-Verfahren zur Simulation von Zweiphasenströmungen. PhD thesis, TU München, 2001.

[13] Yu D. Viscous flow computations with the lattice Boltzmann equation method. PhD thesis, University of Florida, 2002.

[14] Crouse B. Lattice-Boltzmann Strömungs-simulationen auf Baumdaten-strukturen. PhD thesis, TU München, 2003.

[15] Kehrwald D. Numerical analysis of immiscible lattice BGK. PhD thesis, Universität Kaiserslautern, 2003.

[16] Quian YH, d'Humières D, Lallemand P. Lattice BGK models for Navier–Stokes equations. Europhys Lett 1992;17(6):479–84.

[17] Filippova O, Hänel D. Boundary-fitting and local grid refinement for LBGK models. Int J Mod Phys C 1998;8:1271.

[18] Yu D, Mei R, Shyy W. A multi-block lattice Boltzmann method for viscous fluid flows. Int J Numer Methods Fluids 2002;39(2):99–120.

[19] Junk M, Klar Axel, Luo L-S. Asymptotic analysis of the lattice Boltzmann equation. J Comput Phys 2005;210(2):676–704.

[20] Tölke J, Krafczyk M, Schulz M, Rank E. Lattice Boltzmann simulations of binary fluid flow through porous media. Philos Trans Roy Soc Lond A 2002;360(1792):535–45.

[21] Crouse, Rank E, Krafczyk M, Tölke J. A LB-based approach for adaptive flow simulations. Int J Mod Phys B 2002;17:109–12.

[22] Rheinländer M. A consistent grid coupling method for lattice-Boltzmann schemes. In: Krafczyk M, Ladd AJC, Luo L-S, editors. Special issue on mesoscopic methods in engineering and science. J Stat Phys 2005;121(1–2):49–74.

[23] Zapryanov Z, Tabakova S. Dynamics of bubbles drops and rigid particles. Dordrecht: Kluwer; 1999.

[24] Clift R, Grace JR, Weber ME. Bubbles, drops and particles. New York: Academic Press; 1978.

[25] d'Humières D. Generalized lattice Boltzmann equations. Rarefied gas dynamics: theory and simulations. Prog Aeronaut Astronaut 1992;159:450–8.

[26] Lallemand P, Luo LS. Theory of the lattice Boltzmann method: dispersion, dissipation, isotropy, Galilean invariance, and stability. Phys Rev E 2000;61(6):6546–62.

[27] d'Humières D, Ginzburg I, Krafczyk M, Lallemand P, Luo L-S. Multiple-relaxation-time lattice Boltzmann models in three dimensions. Philos Trans Roy Soc Lond A 2002;360:437–51.

[28] Sussman M. A second order coupled level set and volume-of-fluid method for computing growth and collapse of vapor bubbles. J Comput Phys 2003;187:110–36.

[29] Gueyffier D, Li J, Nadim A, Scardovelli S, Zaleski S. Volume of Fluid interface tracking with smoothed surface stress methods for three-dimensional flows. J Comput Phys 1999;152:423–56.

[30] http://www.exa.com/.