Lattice Boltzmann model for free-surface flow and its application to filling process in casting

Irina Ginzburg[*], Konrad Steiner

Fraunhofer Institut für Techno- und Wirtschaftsmathematik, Gottlieb-Daimler-Strasse 49, Kaiserslautern, D-67663, Germany

Abstract

A generalized lattice Boltzmann model to simulate free-surface is constructed in both two and three dimensions. The proposed model satisfies the interfacial boundary conditions accurately. A distinctive feature of the model is that the collision processes is carried out only on the points occupied partially or fully by the fluid. To maintain a sharp interfacial front, the method includes an anti-diffusion algorithm. The unknown distribution functions at the interfacial region are constructed according to the first-order Chapman–Enskog analysis. The interfacial boundary conditions are satisfied exactly by the coefficients in the Chapman–Enskog expansion. The distribution functions are naturally expressed in the local interfacial coordinates. The macroscopic quantities at the interface are extracted from the least-square solutions of a locally linearized system obtained from the known distribution functions. The proposed method does not require any geometric front construction and is robust for any interfacial topology. Simulation results of realistic filling process are presented: rectangular cavity in two dimensions and Hammer box, Campbell box, Sheffield box, and Motorblock in three dimensions. To enhance the stability at high Reynolds numbers, various upwind-type schemes are developed. Free-slip and no-slip boundary conditions are also discussed.
© 2002 Elsevier Science B.V. All rights reserved.

1. Introduction

Numerical modeling of moving interfaces between immiscible fluids is important for many industrial applications. Solving the incompressible Navier–Stokes equations for two-phase flows is a difficult problem since pressure and velocity derivatives may have discontinuities at the interfaces. In addition, a description of the interface motion itself represents a complicated task. Existing methods to treat sharp interfacial problems belong to two main categories: *surface tracking*, a Lagrangian method, and *surface capturing*, an Eulerian one. The former approach explicitly treats the interface as a discontinuity and often the

computational mesh directly tracks the interfaces. The latter one does not consider the exact interface position in the discretization of the governing equations, but takes it into account during the interface advection. These two methods can be combined together in some way when the grids are aligned with the interface (e.g. [17,22,42,53,55,73]). Excellent overviews on various methods to treat interfaces are given by Unverdi and Tryggvason [80], Sussman and Smereka [78], Rider and Kothe [67], Kothe [47], Rudman [68,69], and Scardovelli and Zaleski [74].

Free-surface phenomena are ubiquitous in nature and in many industrial applications. Metal casting is such an example. In this case, the density ratio between the melt metal and the air is of such a disparity that the influence of the air on the melt metal can be ignored. Hence the problem of two-fluid flow with interfaces reduces to the problem of one-fluid flow with free-boundaries. The formulation of the free-boundary flow avoids the steep variations of physical quantities in the interfacial region. Several methods to treat the free-surface problem have been developed. Among volume tracking methods, the most popular one is a volume of fluid (VOF) method due to Hirt and Nicholls [34]. This method has been successfully applied to simulate mold filling with regular grids [2,3,35,52] and has been extended to unstructured grids (e.g. [17,59,60,64,79]). The free Lagrangian method of Fyfe et al. [15] and the grid free smoothed particle hydrodynamics (SPH) methods [11,57] are representative examples of the extension of front tracking methods to treat free-surface problems. In the former case, the computational grid itself is advected by the Lagrangian equations. Whereas in the latter case, the interface is represented by a set of particle positions. A front tracking technique has also been applied to strongly deformable geometries by Galaktionov et al. [16]. A review on free-surface methods based on their applicability to the simulations of the mold filling process is given by Kothe et al. [48].

Intended for solving the Navier–Stokes equation, the lattice Boltzmann (LB) models [33,38] do not involve any global linear or nonlinear systems of equations. Their locality and linearity with respect to computational mesh are absolutely essential for the applications of interfacial problems where the conventional CFD solvers may fail to because of the stiffness and/or large dimensions of the problem. Furthermore, the kinetic nature of the LB method provides the physical basis to deal with such complicated physical phenomena as fluids segregation, diffusion, wetting, evaporation, etc., in a elegant manner. A comparison of two-phase Lattice Boltzmann and VOF methods is available in [75]. An extensive literature on the LB multi-phase and multi-component models can be found in [9,32,54,70,71].

One early LB model for immiscible fluids has been proposed by Gunstensen et al. [28]. In this immiscible lattice Boltzmann (ILB) model, the collision and propagation rules are modified on interfacial grid points in order to introduce the desirable interfacial behavior. The ILB model of Gunstensen and Rothman has been originally designed to simulate flows of two immiscible fluids differentiated only by their colors. In addition to the usual collision and advection steps, the ILB model has an additional "recoloring" step, which preferentially redirects each fluid to the neighboring sites of the same color. This step is accomplished by computing the gradient of the local mass fraction. The "recoloring" step must also preserve the conservation laws. This "recoloring" scheme actually mimics a mechanism of segregation among the two components marked by their colors. Owing to the explicit nature of the LB algorithms, the LB models has difficulty to deal with fluids with large density ratio. In addition, the effective interface boundary conditions implicitly imposed by the LB multi-phase methods have not been given sufficient attention in the past (cf. [19]). The purpose of this paper is to propose a new volume tracking LB method to simulate hydrodynamics with free-surfaces. It can be seen as a modified immiscible lattice Boltzmann model in which one species is the fluid and the other one is considered as vacuum. The proposed LB method is applied to simulate the process of filling a die cavity in metal casting [17,29,52], which is a crucial step in casting process since it determines the quality of the final product.

The new LB method is different from the existing LB multi-phase and multi-component models because the collision only occurs on the "active" cells which are fully or partially filled with fluid. The mass fraction of a cell filled with fluid, which is between zero and one, is an additional variable used in the method.

A ''recoloring operator'', similar to that in the ILB models, determines the redistribution of fluid mass carried by each particle population. The macroscopic variables propagate together with the particle distribution functions in the advection step, according to the usual LB evolution equation. The unknown particle distributions at the front of free-surfaces, which cannot be obtained by the usual LB method, are constructed by using the first-order Chapman–Enskog expansion of the distribution functions. The free-surface boundary conditions are directly met by matching the coefficients of the series solutions of the distribution functions with the boundary conditions. Due to the rotational invariance, the first-order Chapman–Enskog solutions of the distribution functions are naturally expressed in a local coordinates which are normal and tangent to the free-surface. Therefore, the curvilinear interfacial boundaries can be handled easily without reconstruction of mesh geometries.

The solutions for the distribution functions at free-surface are completely specified by the local hydrodynamic variables, i.e., the fluid density and velocity, in two dimensions. In three dimensions, derivatives of velocity tangential to the free-surface are also needed to determine the solutions. The hydrodynamic variables at the free-surface nodes are not determined, however, until all the distributions in a given cell are obtained. The idea here is to derive the hydrodynamic information at free-surface locally, from the *known populations* arriving at a given front node from the neighboring active nodes. In such a way, the resulting solution for unknown distributions at front nodes is implicitly expressed in the form of a linearized function of the known distributions. We obtain unknown quantities from the local least-square solution of a linearized system. This approach follows the basic philosophy of the local second-order boundary (LSOB) method [20] where all first- and second-order momentum derivatives, necessary to impose Dirichlet boundary conditions, are extracted from locally known populations. Therefore, the proposed method is entirely based on a self-consistent kinetic theory, the Chapman–Enskog analysis being consistently applied in the solid and interface boundary regions.

The lattice Boltzmann equation used here is based on the framework of the generalized lattice Boltzmann equation (GLBE) due to d'Humières [38]. In this approach, the collision operator is computed in the space spanned by a basis in momentum space. The basis is presented in a unique form [21] and is suitable for any $DdQb$ model in d dimensions with q velocities [66]. In particular, we present the method in this paper using $D2Q9$ and $D3Q15$ models as examples in two and three dimensions, respectively. The boundary conditions at fluid–solid interfaces are realized by using local reflections of the distribution functions in the spirit of the bounce-back scheme for the no-slip boundary condition. Because the effective accuracy of bounce-back and specular reflections to model no-slip/free-slip boundary conditions (or their combination) depends on the actual choice of all eigenvalues of the collision operator [18,39], we pay a special attention to address this issue. We also pay special attention to the stability of the LB method [50]. Like any LB model, the method develops instabilities at high Reynolds numbers, even if the free eigenvalues are chosen to improve the stability (cf. [50]). This leads us to design schemes with better stability characteristics. Although upwind approach is widely used in finite-difference/finite-elements schemes, it has not yet been adapted for Lattice-Bolzmann methods. Indeed, one of the merits of the LB methods is that their numerical diffusion (at least at second order) is already accounted in the diffusion coefficient. We show using linear convective-diffusion problem as an example that higher-order terms in Chapman–Enskog expansion can introduce negative numerical diffusion into the scheme. In order to compensate it, artificial numerical diffusion can be added. In so far, various upwind LB schemes are constructed in [23], in analogy to one-dimensional upwind [10], full-upwind, and streamline-type multi-dimensional upwind schemes [4]. In the present work, we use the most crude but robust *explicit upwind* approach for free interface simulations at high *Re* numbers.

The paper is structured as follows. In Section 2, the LB framework and basic macroscopic relations are given. In Section 3.1, first-order Chapman–Enskog expansion at interface is presented. In Section 3.2, interface advection with recoloring step is discussed. Reconstruction of unknown populations at the interface is described in Section 3.3. Brief outline of the numerical algorithm is given in Section 3.4. Boundary conditions are discussed in Section 4. Overview of the algorithm is in Section 5. Explicit upwind scheme is

discussed in Section 6. Section 7 considers the different aspects of the algorithm using 2D cavity filling and benchmark 3D simulations in injected molding. Concluding remarks are in Section 8. Details to implementation of generalized LB equation are sketched in Appendix A.

2. Basic theory of lattice Boltzmann equation

2.1. Lattice Boltzmann equation

The lattice Boltzmann equation (LBE) is often written in the following form [77]:

$$\widetilde{N}_i(\boldsymbol{r},t) = N_i(\boldsymbol{r},t) + \sum_{j=0}^{b_m} \mathsf{A}_{ij}[N_j(\boldsymbol{r},t) - N_j^{\mathrm{eq.}}(\boldsymbol{r},t)] + t_p^{\star}(\boldsymbol{C}_i \cdot \boldsymbol{F}), \tag{1a}$$

$$N_i(\boldsymbol{r}+\boldsymbol{C}_i, t+1) = \widetilde{N}_i(\boldsymbol{r},t), \quad i \in \{0,\ldots,b_m\}, \tag{1b}$$

where N_i is the population of the particle moving with D-dimensional velocity $\boldsymbol{C}_i$ ($\boldsymbol{C}_0$ is a zero vector), A is the collision matrix, $\boldsymbol{F}$ is an external force; weight coefficient $t_p^{\star}$ depends on the discrete velocity set $\boldsymbol{C}_i$, and the index p is equal to c_i^2 ($c_i^2 = \|\boldsymbol{C}_i\|^2$). Equilibrium function $\mathbf{N}^{\mathrm{eq.}}$ is introduced by Eq. (10) and the coefficients $t_p^{\star}$ are given in Table 1. They satisfy the following equations:

$$\sum_{i=1}^{b_m} t_p^{\star} C_{i\alpha}^2 = 1 \quad \forall \alpha = 1,\ldots,D, \quad \text{and} \quad t_0^{\star} = 3 - \sum_{p \neq 0} t_p^{\star}. \tag{2}$$

There are two essential steps in Eq. (1a): *collision* (a) and *propagation* (b). Density ρ and momentum $\boldsymbol{j}$ are defined as

$$\rho(\boldsymbol{r},t) = \sum_{i=0}^{b_m} N_i(\boldsymbol{r},t), \tag{3a}$$

$$\boldsymbol{j}(\boldsymbol{r},t) = \boldsymbol{J} + \frac{1}{2}\boldsymbol{F}, \quad \boldsymbol{J} = \sum_{i=1}^{b_m} N_i(\boldsymbol{r},t)\boldsymbol{C}_i. \tag{3b}$$

The reason to modify the momentum in the presence of the forcing term is discussed in Section 4 and can also be found in a number of references [5,18,39,49]. The mass and momentum conservation laws impose the following conditions on the collision matrix A:

$$\mathsf{A} \cdot \mathbf{1} = \mathsf{A} \cdot \mathbf{C}_\alpha = \mathbf{0} \quad \forall \alpha = 1,\ldots,D, \tag{4}$$

where $\mathbf{1} = \{1,\ldots,1\}$ and the (b_m+1)-vector $\mathbf{C}_\alpha$ is built from the components of the (b_m+1) population velocities in direction α.

Table 1
Equilibrium weights $t_p^{\star}$ and $r_p^{\star}$

Model	$t_0^{\star}$	$t_1^{\star}$	$t_2^{\star}$	$t_3^{\star}$	$r_0^{\star}$	$r_1^{\star}$	$r_2^{\star}$	$r_3^{\star}$
D2Q9	4/3	1/3	1/12	–	$(3-5c_s^2)/3$	$c_s^2/3$	$c_s^2/12$	–
D3Q15	2/3	1/3	–	1/24	$(3-7c_s^2)/3$	$1c_s^2/3$	–	$c_s^2/24$

The collision matrix is fully determined by the choice of its non-zero eigenvalues and the corresponding eigenvectors. To satisfy the linear stability conditions [33], the non-zero eigenvalues must lie in the interval $]-2, 0[$. Mass vector $\mathbf{1}$ and the vectors $\mathbf{C}_\alpha$ are the eigenvectors associated with the zero eigenvalues—they are the conserved modes in the model. Let $\{\mathbf{e}_k\}$, $k = 0, \ldots, b_m$, denote the orthonormal basis in momentum space, constructed as the polynomials of the vectors $\mathbf{C}_\alpha$. Let us assume that this basis represents the set of the eigenvectors of the matrix A, associated with the eigenvalues $\{\lambda_k\}$. Following [38], we rewrite Eq. (1a) as its projection on this basis

$$\widetilde{N}_i(\boldsymbol{r}, t) = N_i(\boldsymbol{r}, t) + \sum_{k=0}^{b_m} \lambda_k (\mathbf{N} - \mathbf{N}^{\text{eq.}}, \mathbf{e}_k) e_{ki} + t_p^{\star}(\boldsymbol{C}_i, \boldsymbol{F}), \tag{5}$$

$$N_i(\boldsymbol{r} + \boldsymbol{C}_i, t + 1) = \widetilde{N}_i(\boldsymbol{r}, t), \quad i \in \{0, \ldots, b_m\}. \tag{6}$$

Note that Eq. (5) replaces the explicit use of the collision matrix A. The eigenvalues can also be easily adjusted during computations, if necessary, provided that they satisfy the stability constraints. When all non-zero eigenvalues $\{\lambda_k\}$ are set to be equal to $-1/\tau$, Eq. (5) reduces to the lattice BGK model [8,66]:

$$N_i(\boldsymbol{r} + \boldsymbol{C}_i, t + 1) = N_i(\boldsymbol{r}, t) - \frac{1}{\tau}(N_i - N_i^{\text{eq.}}) + t_p^{\star}(\boldsymbol{C}_i, \boldsymbol{F}). \tag{7}$$

In the case of the lattice BGK equation (7), ρ and $\boldsymbol{J}$ are conserved provided that the equilibrium function satisfies the following conditions:

$$(\mathbf{N} - \mathbf{N}^{\text{eq.}}, \mathbf{1}) = 0, \quad (\mathbf{N} - \mathbf{N}^{\text{eq.}}, \mathbf{C}_\alpha) = 0 \quad \forall \alpha = 1, \ldots, D. \tag{8}$$

The labeling of the discrete velocity sets for $D2Q9$ and $D3Q15$ models in this paper, their basis vectors and associated eigenvectors are given in Appendices A.1 and A.2, respectively.

2.2. Hydrodynamics equations

The solution for the population function N_i is usually obtained in a perturbative form of Chapman–Enskog expansion [14] in the powers of small perturbation parameter $\epsilon = \delta_x/L$ (L is the characteristic length):

$$N_i(\boldsymbol{r}, t) = N_i^{\text{eq.}}(\boldsymbol{r}, t) + \epsilon N_i^{(1)}(\boldsymbol{r}, t) + \epsilon^2 N_i^{(2)}(\boldsymbol{r}, t) + \mathrm{O}(\epsilon^3), \quad i = 0, \ldots, b_m. \tag{9}$$

The equilibrium population $N_i^{\text{eq.}}(\boldsymbol{r}, t)$ can take a form (see [61,66])

$$N_i^{\text{eq.}} = r_p^{\star}\rho + t_p^{\star}\left[J_\alpha C_{i\alpha} + \rho \frac{u_\alpha u_\beta}{2}(3C_{i\alpha}C_{i\beta} - \delta_{\alpha\beta})\right], \quad \boldsymbol{u} = \frac{\boldsymbol{j}}{\rho}, \quad \boldsymbol{J} = \boldsymbol{j} - \frac{1}{2}\boldsymbol{F}. \tag{10}$$

Parameter $r_p^{\star}$ is related with t_p as

$$r_p^{\star} = t_p^{\star} c_{\text{s}}^2 \quad \text{when } p \neq 0 \quad \text{and} \quad r_0^{\star} = 1 - \sum_{p \neq 0} r_p^{\star}, \tag{11}$$

where c_{s} is the sound speed of the model, which is to be discussed later. The first-order correction to the equilibrium, $N_i^{(1)}$, in standard coordinate system is given in details in Appendix A.5). One crucial ingredient is that $\epsilon \mathbf{N}^{(1)}$ is isotropic (rotational invariant) and can be written in any orthogonal coordinate system $\{x', y', z'\}$ (z' is omitted in two dimensions) as

$$\epsilon N_i^{(1)} = \frac{1}{\lambda_\psi} \frac{\partial j_{\alpha'}}{\partial \beta'} Q_{i\alpha'\beta'} + \frac{1}{\lambda_e}(\nabla \cdot \boldsymbol{j}) E_i^{\text{im}}, \quad \{\alpha', \beta'\} \in \{x', y', z'\}, \tag{12}$$

where

$$Q_{i\alpha'\beta'} = t_p^\star \left(C_{i\alpha'} C_{i\beta'} - \frac{c_i^2}{D} \delta_{\alpha'\beta'} \right) \quad \text{and} \quad E_i^{\text{im}} = t_p^\star \frac{c_i^2}{D} - r_p^\star. \tag{13}$$

Projection of the vectors $\mathbf{N}^{\text{eq.}}$ of Eq. (10) and $\epsilon \mathbf{N}^{(1)}$ of Eq. (12) written in the standard coordinate system on the basis vectors $\mathbf{e}_k$ are given by formulas (A.4) and (A.10). The hydrodynamic equations for ρ and $\boldsymbol{j}$ derived from the model are

$$\partial_t \rho + \nabla \cdot \boldsymbol{j} = 0, \tag{14a}$$

$$\partial_t \boldsymbol{j} + \nabla \cdot \left(\frac{\boldsymbol{j} \otimes \boldsymbol{j}}{\rho} \right) = -c_s^2 \nabla \rho + \nabla \cdot (v \nabla \boldsymbol{j}) + \nabla (\nabla \cdot v_\xi \boldsymbol{j}) + \boldsymbol{F}, \tag{14b}$$

where the kinematic viscosity v and the bulk viscosity viscosity v_ξ are related to two non-zero eigenvalues of the collision matrix

$$v = \frac{1}{3}\left(\tau - \frac{1}{2} \right), \quad \tau \equiv -\frac{1}{\lambda_\psi}, \tag{15a}$$

$$v_\xi = [v(2 - 3\mathscr{C}) + \xi], \quad \xi = -(\mathscr{C} - c_s^2)\left(\frac{1}{\lambda_e} + \frac{1}{2} \right), \tag{15b}$$

and for $D2Q9$ and $D3Q15$ models, the coefficient $\mathscr{C}$ is given by

$$\mathscr{C} = \frac{D+2}{3D}. \tag{16}$$

For the athermal LBE models, the pressure P is given by the equation of state for an ideal gas,

$$P = c_s^2 \rho.$$

By introducing characteristic LB velocity U and assuming the density fluctuates about its average, ρ_0, the pressure can be rescaled in the following dimensionless form:

$$P \rightarrow \frac{(P - P_0)}{\rho_0 U^2}, \quad P_0 \equiv c_s^2 \rho_0. \tag{17}$$

Using the following scalings:

$$x \rightarrow x/L, \quad t \rightarrow tU/L, \quad \boldsymbol{u} \rightarrow \boldsymbol{u}/U, \tag{18}$$

we can write the hydrodynamic equations in the following dimensionless form:

$$\text{M}^2 \partial_t P + \nabla \cdot \rho \boldsymbol{u} = 0, \tag{19a}$$

$$\partial_t \rho \boldsymbol{u} + \nabla \cdot (\rho \boldsymbol{u} \otimes \boldsymbol{u}) = -\rho_0 \nabla P + \frac{1}{Re} \nabla \cdot (\nabla(\rho \boldsymbol{u})) + \nabla \left(\frac{v_\xi}{UL} \nabla \cdot (\rho \boldsymbol{u}) \right) + \rho_0 \frac{\vec{1}_g}{Fr}, \tag{19b}$$

where the Reynolds number Re, the Froude number Fr, and the Mach number M are defined as the following:

$$Re = \frac{LU}{v}, \quad Fr = \frac{U^2}{FL}, \quad M = \frac{U}{c_s}. \tag{20}$$

Accordingly, the density can also be written in terms of the dimensionless pressure

$$\rho = \rho_0(1 + M^2 P). \tag{21}$$

If we choose to neglect the density fluctuation, which is second order in the Mach number M, as indicated in Eq. (21), we obtain the incompressible Navier–Stokes equations

$$\nabla \cdot \boldsymbol{u} = 0, \tag{22a}$$

$$\partial_t \boldsymbol{u} + \boldsymbol{u} \cdot \nabla \boldsymbol{u} + \nabla P = \frac{1}{Re} \nabla^2 \boldsymbol{u} + \frac{\vec{1}_g}{Fr}. \tag{22b}$$

In particular, Stokes equation can be obtained by omitting the nonlinear term in the equilibrium population of Eq. (10). The sound speed is a free parameter in the equilibrium. The restriction $c_s^2 < \mathscr{C}$ comes from the condition $\xi > 0$ (see Eq. (15b)). Based on linear stability analysis, the choice of $c_s^2 = 1/3$ is the optimal choice (see [50]) and it also corresponds to the LB models derived by another approaches (cf. [31,43]). Small Mach number M means that $U \ll c_s$.

2.3. Interface boundary conditions

In absence of surface tension between two fluids, one heavy and one light, the balance of momentum at the interface according to the Navier–Stokes equation (14b) leads to the following equation at the interface:

$$[2\nu \mathrm{D} j \cdot \boldsymbol{n} - P\boldsymbol{n}]_S = 0, \quad \mathrm{D} j = \frac{1}{2} (\partial_\alpha j_\beta + \partial_\beta j_\alpha). \tag{23}$$

If the density ratio among the two fluids is so disparate so that the dynamic viscosity of the light fluid (e.g. gas) is negligible, the above equation (23) reduces to the following free interface conditions for the heavy fluid (e.g. liquid):

$$P - 2\nu \frac{\partial j_n}{\partial n} = P_0, \tag{24a}$$

$$\frac{\partial j_\tau}{\partial n} + \frac{\partial j_n}{\partial \tau} = 0, \quad \tau \in \{\tau_1, \tau_2\}. \tag{24b}$$

Here j_n and $j_\tau = \{j_{\tau_1}, j_{\tau_2}\}$ are the normal and tangential momentum components of the viscous (heavy) fluid; and P and P_0 are the pressures of the heavy and the light fluids at the interface S, respectively.

3. Lattice Boltzmann model for free interface

The populations $\{N_i(\boldsymbol{r}, t)\}$ and the total mass of one-fluid $\rho_f(\boldsymbol{r}, t)$ represent the main independent variables of the LB free-surface algorithm. In empty cells, $\rho_f = 0$; in the cells fully filled with fluid, $m_f = \rho$ and in partially filled cells, $0 < \rho_f < \rho$. The collision step is performed only in the "active" cells where $\rho_f \neq 0$. Similar to the VOF method, gravitation force is computed in proportion to current occupation of the cell:

$$\boldsymbol{F} = \frac{\rho_f}{\rho} (\rho_0 \boldsymbol{g}). \tag{25}$$

The unknown populations at nodes adjacent to interface, which cannot be determined from the lattice Boltzmann equation, are constructed by using Chapman–Enskog analysis up to the first order in the expansion in ϵ.

3.1. First-order expansion of N_i at interface

Here we write $\mathbf{N}^{(1)}$ given by the relation (12) in the interface coordinate system $\{x', y', z'\} = (n, \tau_1, \tau_2)$, such that

$$\epsilon N_i^{(1)} = \frac{1}{\lambda_\psi} \left(\frac{\partial j_\tau}{\partial n} + \frac{\partial j_n}{\partial \tau} \right) Q_{in\tau} + \frac{1}{\lambda_\psi} \left(\frac{\partial j_n}{\partial n} Q_{inn} + \frac{\partial j_\tau}{\partial \tau} Q_{i\tau\tau} \right) \tag{26}$$

$$+ \frac{\nabla \cdot \boldsymbol{j}}{\lambda_e} E_i^{\text{im}}, \quad \tau \in \{\tau_1, \tau_2\}. \tag{27}$$

In the above equation, the non-diagonal components of the strain-rate tensor Dj are in the prefactor of $\mathbf{Q}_{n\tau}$. From the boundary condition (24b), the terms involving $\mathbf{Q}_{n\tau}$ should be set to zero at the interface. By substitution of the divergence condition $\partial j_\tau / \partial \tau = \nabla \cdot \boldsymbol{j} - (\partial j_n / \partial n)$ and $Q_{i\tau\tau} = -Q_{inn}$, we obtain the first-order solution of N_i at the interface

$$N_i = N_i^{\text{eq.}} + \frac{2}{\lambda_\psi} \frac{\partial j_n}{\partial n} Q_{inn} + \nabla \cdot \boldsymbol{j} \left(\frac{1}{\lambda_e} E_i^{\text{im}} - \frac{1}{\lambda_\psi} Q_{inn} \right) + \mathrm{O}(\epsilon^2) \tag{28}$$

in two dimensions. In three dimensions, we first write

$$\frac{\partial j_{\tau_1}}{\partial \tau_1} Q_{i\tau_1\tau_1} + \frac{\partial j_{\tau_2}}{\partial \tau_2} Q_{i\tau_2\tau_2} = \frac{1}{2} \left(\frac{\partial j_{\tau_1}}{\partial \tau_1} - \frac{\partial j_{\tau_2}}{\partial \tau_2} \right) (Q_{i\tau_1\tau_1} - Q_{i\tau_2\tau_2}) + \frac{1}{2} \left(\frac{\partial j_{\tau_1}}{\partial \tau_1} + \frac{\partial j_{\tau_2}}{\partial \tau_2} \right) (Q_{i\tau_1\tau_1} + Q_{i\tau_2\tau_2}). \tag{29}$$

Then using the divergence condition

$$\frac{\partial j_{\tau_1}}{\partial \tau_1} + \frac{\partial j_{\tau_2}}{\partial \tau_2} = \nabla \cdot \boldsymbol{j} - \frac{\partial j_n}{\partial n}$$

and the $Q_{i\tau_1\tau_1} + Q_{i\tau_2\tau_2} = -Q_{inn}$, we obtain

$$N_i = N_i^{\text{eq.}} + \frac{3}{2\lambda_\psi} \frac{\partial j_n}{\partial n} Q_{inn} + \frac{1}{2\lambda_\psi} \left(\frac{\partial j_{\tau_1}}{\partial \tau_1} - \frac{\partial j_{\tau_2}}{\partial \tau_2} \right) (Q_{i\tau_1\tau_1} - Q_{i\tau_2\tau_2}) + \frac{1}{\lambda_\psi} \left(\frac{\partial j_{\tau_1}}{\partial \tau_2} + \frac{\partial j_{\tau_2}}{\partial \tau_1} \right) Q_{i\tau_1\tau_2}$$

$$+ \nabla \cdot \boldsymbol{j} \left(\frac{1}{\lambda_e} E_i^{\text{im}} - \frac{1}{2\lambda_\psi} Q_{inn} \right) + \mathrm{O}(\epsilon^2) \tag{30}$$

in three dimensions. Since $\nabla \cdot \boldsymbol{j}$ is of order $\mathrm{O}(\mathrm{M}^2)$, it can be neglected in the above equations for incompressible flows. Then using Eq. (24a) for $\partial j_n / \partial n$, Eqs. (28) and (30) become

$$N_i = N_i^{\text{eq.}} + \frac{(P - P_0)}{\nu \lambda_\psi} Q_{inn} + \mathrm{O}(\epsilon^2) \tag{31}$$

in two-dimensions and

$$N_i = N_i^{\text{eq.}} + \frac{3(P - P_0)}{4\lambda_\psi \nu} Q_{inn} + \frac{1}{2\lambda_\psi} (Q_{i\tau_1\tau_1} - Q_{i\tau_2\tau_2}) \left(\frac{\partial j_{\tau_1}}{\partial \tau_1} - \frac{\partial j_{\tau_2}}{\partial \tau_2} \right) + \frac{1}{\lambda_\psi} Q_{i\tau_1\tau_2} \left(\frac{\partial j_{\tau_1}}{\partial \tau_2} + \frac{\partial j_{\tau_2}}{\partial \tau_1} \right) + \mathrm{O}(\epsilon^2) \tag{32}$$

in three dimensions.

The values of the known populations, which propagate from the neighboring active points to the front nodes, are used to compute unknown macroscopic quantities (ρ, $\boldsymbol{j}$, and the derivatives of $\boldsymbol{j}$ tangential to interface) from the above equations by solving the linearized system by means of least-square fitting. We

refer to local computing of unknown populations at front nodes as *reconstruction step* which is further discussed in Section 3.3.

3.2. Recoloring step

In order to describe two-phase behavior, ILB models [28] include two type (color) populations: $\{R_i(\mathbf{r}, t)\}$ and $\{B_i(\mathbf{r}, t)\}$. Indeed, ILB operates only on their sums: $\sum_{i=0}^{b_m} R_i(\mathbf{r}, t)$ and $\sum_{i=0}^{b_m} B_i(\mathbf{r}, t)$. We represent then the ILB model in an equivalent form [24], by using only populations $N_i(\mathbf{r}, t)$ and the total mass of the fluid $\rho_f(\mathbf{r}, t)$ (or mass fraction of one-fluid $m_f = \rho_f(\mathbf{r}, t)/\rho$) as independent variables. Here, $N_i(\mathbf{r}, t)$ can be interpreted as a sum of $R_i(\mathbf{r}, t)$ and $B_i(\mathbf{r}, t)$. Local quantity of the another fluid is equal to $\rho(\mathbf{r}, t) - \rho_f(\mathbf{r}, t)$. Recoloring algorithm is employed to advect the quantity ρ_f between the cells by keeping a sharp front. The operation tends to send as much as possible fluid phase into direction of its bulk. For this purpose, one has to find the solution, denoted $\mathbf{R}^R(\mathbf{r}, t)$, which maximizes the post-collision color flux $\tilde{F}[\mathbf{R}]$ along the normal $\mathbf{n}$ to the interface, with

$$\tilde{F}(\mathbf{R}) = \sum_{i=0}^{b_m} R_i \mathbf{C}_i \cdot \mathbf{n}, \quad \sum_{i=0}^{b_m} R_i = \rho_f(\mathbf{r}, t), \quad 0 \leqslant R_i \leqslant \widetilde{N}_i(\mathbf{r}, t). \tag{33}$$

The solution is constrained by mass conservation. Note that, due to the above inequality, this solution exists only when the post-collision populations are positive. Also it is on common unproven belief that this also is required for stability. Below, we call "good" cells where the post-collision populations $\widetilde{N}_i(\mathbf{r}, t)$ in Eq. (5) are positive. The way we treat other cells is described in Section 3.3. In a simplest way, one can find the solution to (33) as follows. Starting from the population whose velocity is the closest to the normal direction (has maximal $(\mathbf{C}_i \cdot \mathbf{n})$ value), one goes down to those which has the minimal value and put R_i equal to the maximal possible value $(R_i \leqslant \widetilde{N}_i)$. The procedure stops when no more color is available $(\sum_{i=0}^{b_m} R_i = \rho_f)$. Additional efforts can be required to keep the symmetry of the solution. The new value $\rho_f(\mathbf{r}, t + 1)$ is equal to the sum of all incoming fluid quantities $R_i^R(\mathbf{r} - \mathbf{C}_i, t)$:

$$\rho_f(\mathbf{r}, t + 1) = \sum_{i=0}^{b_m} R_i^R(\mathbf{r} - \mathbf{C}_i, t). \tag{34}$$

Since $\rho_f(\mathbf{r}, t + 1)$ can be updated immediately after recoloring steps in neighboring nodes, no additional storage is needed for the solution $\mathbf{R}^R(\mathbf{r}, t)$ and this algorithm reduces the ILB memory requirement by almost a factor two. We compute $\mathbf{n}$ similar as in VOF methods [51,67,83]

$$\mathbf{n} \approx \nabla m_f, \tag{35}$$

with the following central difference approximation

$$\mathbf{n} = \sum_{i=1}^{b_m} s_i \mathbf{C}_i m_f(\mathbf{r} + \mathbf{C}_i). \tag{36}$$

where

$$s = \begin{cases} 1/4 \times \{2, 2, 2, 2, 1, 1, 1, 1\} & \text{for } D2Q9, \\ 1/8 \times \{4, 4, 4, 4, 1, 1, 1, 1, 1, 1, 1, 1\} & \text{for } D3Q15. \end{cases}$$

Following [62,67], the method for normal estimation which reproduces a line (a plane) exactly regardless its orientation with respect to fixed coordinate system is referred to as second-order method. With this cri-
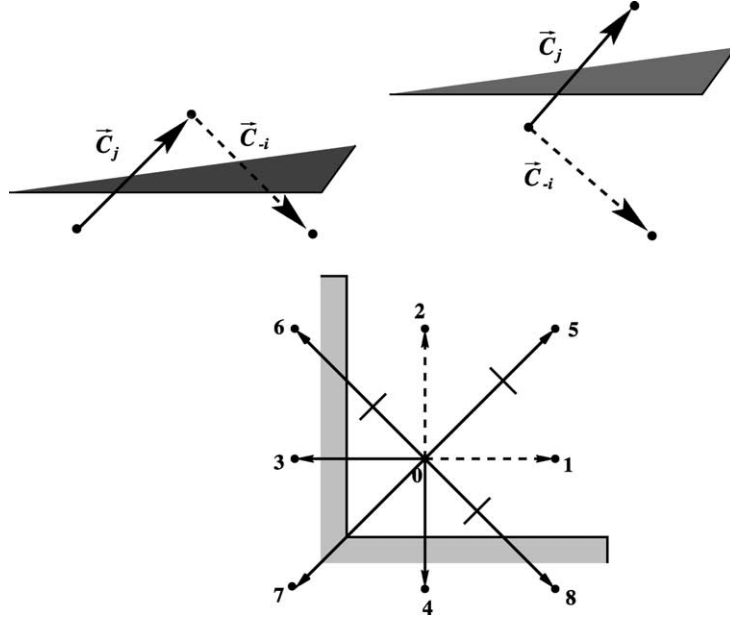
Fig. 1. Left to right: non-local specular reflection, local specular reflexion, 2D corner. In last case, in 2D: $N_1 = \tilde{N}_3$, $N_2 = \tilde{N}_4$, $N_5 = (1/2)(\tilde{N}_6 + \tilde{N}_8)$, $N_6 = (1/2)(\tilde{N}_6 + \tilde{N}_7)$, $N_8 = (1/2)(\tilde{N}_8 + \tilde{N}_7)$.

terion, approximation (36) which is done separately for each cell is only first-order accurate. This is confirmed by simple advection tests in Section 7.1. The possibility to have wetting/non-wetting condition at solid boundary is controlled by the following assignment: $\rho_f(r^{\text{solid}}) = \rho_f \times w$, $w \geqslant 0$ in wetting case and $w \leqslant 0$, otherwise. In computations below we assume mostly that interface is perpendicular to solid boundary: $(\boldsymbol{n}, \boldsymbol{n}^s) = 0$. Here and below, $(n^s, \tau_1^s, \tau_2^s)$ denote the normal and the tangential vector components with respect to the solid wall. In order to model this condition with the relation (36), we define $\rho_f(r^{\text{solid}})$ at smooth enough walls as

$$\rho_f(\boldsymbol{r} + \boldsymbol{C}_j) = \rho_f(\boldsymbol{r} + \boldsymbol{C}_i) \quad \text{if } \boldsymbol{r} + \boldsymbol{C}_j = \boldsymbol{r}^{\text{solid}}, C_{jn^s} = -C_{in^s}, \quad \text{and} \quad C_{j\tau^s} = C_{i\tau^s}. \tag{37}$$

Indeed, the condition (37) uses the same pairs of populations as a specular reflection (56) sketched in Fig. 1.

3.3. Reconstruction step

According to our algorithm, cell is active if its ρ_f value is positive. We define an interface cell as a cell where the populations are separated into two sets: known and unknown. Known populations are those which arrive from already active cells. Unknown populations are those which would arrive from the non-active cells. We distinguish then two types of interface cells: interface (I) cells are those which have been already active at the previous time step; new interface (N) cells are those which have not been active at the previous time step. Let us denote as I^+ (I^-) the set of indices of locally known (unknown) populations N_i^+ (N_i^-, respectively). The numbers s^+ and s^- of known and unknown populations are related by

$$s^+ + s^- = b_m + 1. \tag{38}$$

One can represent the population expansions (31) and (32) as

$$\mathbf{N} = \mathrm{B}X + \boldsymbol{b}, X = \begin{cases} \{\rho, j_x, j_y\}, & \text{2D}; \\ \{\rho, j_x, j_y, j_z, \frac{\partial j_{\tau_1}}{\partial \tau_1} - \frac{\partial j_{\tau_2}}{\partial \tau_2}, \frac{\partial j_{\tau_1}}{\partial \tau_2} + \frac{\partial j_{\tau_2}}{\partial \tau_1}\}, & \text{3D}. \end{cases} \tag{39}$$

Vector X contains $n_v = 3$ unknown macroscopic quantities in 2D and $n_v = 6$ unknowns in 3D. When we do not neglect the term $\nabla \cdot \boldsymbol{j}$ in Eqs. (28) and (30), we include it into the list of variables X. The elements of the matrix B depend upon the linearization of the equilibrium (10).

Linearization with respect to momentum approximates the nonlinear terms $\rho u_\alpha u_\beta$ in (10) as

$$\rho u_\alpha u_\beta = j_\alpha u_\beta^\star. \tag{40}$$

Approximate velocity ($u_\beta^\star$) and density ($\rho^\star$) values are discussed below. Let us introduce $Q_{i\alpha}$

$$Q_{i\alpha} = \sum_\beta \frac{u_\beta^\star}{2} (3C_{i\alpha}C_{i\beta} - \delta_{\alpha\beta}) \quad \forall \alpha = 1, \ldots, D. \tag{41}$$

Then for the *D2Q9* model the coefficients in (39) are

$$B_{i,1} = r_p^\star + \frac{c_s^2}{v} \frac{1}{\lambda_\psi} Q_{inn}, \tag{42a}$$

$$B_{i,2} = t_p^\star (C_{ix} + Q_{ix}), \tag{42b}$$

$$B_{i,3} = t_p^\star (C_{iy} + Q_{iy}), \tag{42c}$$

$$b_i = -\frac{1}{v} \frac{1}{\lambda_\psi} P_0 Q_{inn} - \frac{1}{2} t_p^\star \frac{\rho_f}{\rho^\star} \rho_0 (\boldsymbol{C}_i, \boldsymbol{g}) \tag{42d}$$

and for the *D3Q15* model:

$$B_{i,1} = r_p^\star + \frac{3c_s^2}{4v} \frac{1}{\lambda_\psi} Q_{inn}, \tag{43a}$$

$$B_{i,2} = t_p^\star (C_{ix} + Q_{ix}), \tag{43b}$$

$$B_{i,3} = t_p^\star (C_{iy} + Q_{iy}), \tag{43c}$$

$$B_{i,4} = t_p^\star (C_{iz} + Q_{iz}), \tag{43d}$$

$$B_{i,5} = Q_{i\tau_1\tau_1} - Q_{i\tau_2\tau_2}, \tag{43e}$$

$$B_{i,6} = Q_{i\tau_1\tau_2}, \tag{43f}$$

$$b_i = -\frac{3}{4v} \frac{1}{\lambda_\psi} P_0 Q_{inn} - \frac{1}{2} t_p^\star \frac{\rho_f}{\rho^\star} \rho_0 (\boldsymbol{C}_i, \boldsymbol{g}). \tag{43g}$$

Approximation to force-term (25) appears in $\boldsymbol{b}$ since we consider here $\boldsymbol{j}$ and not $\boldsymbol{J}$ as an unknown variable in $\boldsymbol{X}$ (cf. (10)). The linearized equations to find $\boldsymbol{X}$ correspond to S^+ known populations N_i^+

$$B_{ij}X_j = r_i, \quad r_i = N_i^+ - b_i, \quad i \in I^+. \tag{44}$$

One can introduce further constraints on $\boldsymbol{X}$. We considerably improve stability and accuracy when the solution is required to fulfill approximate density definition (3a) in a form:

$$\rho - \sum_{i \in I^-} N_i^- = \sum_{i \in I^+} N_i^+. \tag{45}$$

Substitution of the population expansion (39) into (45) for N_i^- yields an additional equation:

$$\sum_j B_{\rho,j}X_j = r_\rho, \tag{46a}$$

$$B_{\rho,1} = 1 - \sum_{i \in I^-} B_{i,1}, \tag{46b}$$

$$B_{\rho,k} = -\sum_{i \in I^-} B_{i,k}, \quad k = 1,\ldots,n_v \tag{46c}$$

$$r_\rho = \sum_{i \in I^+} N_i^+ + \sum_{i \in I^-} b_i. \tag{46d}$$

This completes assembling of the matrix B and the vector $\boldsymbol{b}$. Different from the relation (40) linearizations of the equilibrium can be proposed.

Linearization with respect to density, in particular, treats the nonlinear term as

$$\rho u_\alpha u_\beta = \rho u_\alpha^\star u_\beta^\star. \tag{47}$$

In case (47), one can take $\boldsymbol{J}$ itself as a component in $\boldsymbol{X}$. This avoids approximation of the density (cf. (42a)–(43a)). In 2D, for example, relations (42a) are modified as follows:

$$B_{i,1} = r_p^\star + \frac{c_s^2}{v}\frac{1}{\lambda_\psi}Q_{inn} + t_p^\star \sum_{\alpha\beta} \frac{u_\alpha^\star u_\beta^\star}{2}(3C_{i\alpha}C_{i\beta} - \delta_{\alpha\beta}), \tag{48a}$$

$$B_{i,2} = t_p^\star C_{ix}, \tag{48b}$$

$$B_{i,3} = t_p^\star C_{iy}, \tag{48c}$$

$$b_i = -\frac{1}{v}\frac{1}{\lambda_\psi}P_0 Q_{inn}, \tag{48d}$$

$$\boldsymbol{X} = \{\rho, J_x, J_y\}. \tag{48e}$$

Linearized system of equations (44), (45) contains $m = s^+ + 1$ equations: $2 \leqslant m \leqslant b_m + 1$. The number of variables n_v is equal to the number of components of the vector X. When $n_v \leqslant m$, we solve the linear system by using fast least-square method with permutations. Single Value Decomposition Method [65] can be used as well but it has been found to be much slower for linear systems used here. If the linear solver detects that the system is singular, or when $n_v > m$, we use extrapolations for unknown populations from neighbor "good" active points (as defined in Section 3.2). Similar procedure is employed when appear negative populations after the reconstruction or after collision. We show below that the relative part of "bad" situations is very small in stable calculations. When combinations of tangential derivatives in 3D are not included to X (39), one does not need then to construct tangential vectors τ_1 and τ_2. Moreover, this reduces the number of singular cases since the number of unknowns decreases from $n_v = 6$ to $n_v = 4$. No important impact on the solution was detected because of this approximation.

Approximate values $\rho^\star$ and $u^\star$ are obtained as follows. In already active I-cells, the previous time step solution is used. In new interface N-cells, an extrapolation from the active cells lying as close as possible along the normal n is employed. At least one neighbor active node always exists by the definition of N-cell, otherwise it would not be activated. Since the collision, and hence update of ρ and u, is done first in B- and I-cells, reconstruction step in N-cells can use current solutions in neighbor "good" nodes for extrapolations (see the steps 6–8 of the algorithm). Additionally, other successful N-cells can be used for extrapolations.

The summary of the local reconstruction procedure reads:
1. Compute n. When necessary, compute τ_1 and τ_2.
2. Extrapolate (in time or space) velocity and density values.
3. Compute B and b.
4. Solve linearized system.
5. Compute unknown populations in a form (39).

One can then iterate the steps 3–5 by using velocity/density values obtained at a previous sub-iteration for approximations. We detect, however, only weak influence of this procedure on the accuracy/stability.

3.4. Brief outline of the numerical algorithm

We initialize first ρ_f in all liquid cells at $t = 0$. The populations are then initialized to their equilibrium values and first collision step is performed in active cells where $\rho_f > 0$. We refer to active cells as $A(t)$-cells below. Subsequent steps at time t, $t = 0, \ldots,$ are
1. Compute $\rho_f(r, t + 1)$ in all cells by recoloring technique.
2. Divide all cells into active/non-active cells: $(r, t + 1) \in A(t + 1)$ if $\rho_f(r, t + 1) > 0$; otherwise it is non-active.
3. Propagate populations from $A(t)$ into $A(t + 1)$ cells.
4. Classify known/unknown populations in $A(t + 1)$ cells: $N_i(r, t + 1)$ is marked as *known population* if $(r - C_i, t) \in A(t)$. Otherwise it is marked as *unknown population*.
5. Divide $A(t + 1)$ cells into B/I/N-cells:
   - $(r, t + 1)$ is marked as B(*bulk*)-cell if it has obtained only known populations.
   - $(r, t + 1)$ is marked as I(*interface*)-cell if it has obtained at least one unknown population and if $(r, t) \in A(t)$.
   - $(r, t + 1)$ is marked as N(*new interface*)-cell if $(r, t) \notin A(t)$.
6. Perform collision in B-cells.
7. Perform reconstruction and collision in I-cells.
8. Perform reconstruction and collision in N-cells.
9. $t = t + 1$; Go to 1.

This completes the short description of the LB free interface algorithm.

4. Boundary conditions

4.1. No-slip boundary conditions

While applying the method in very complex geometries, we currently accept its "step-wise" cell-centered discretization on the regular rectangular grids. We apply at boundary nodes the bounce-back rule (b.b.) where the populations leaving the fluid return to the node of departure with the opposite velocity:

$$N_{-i}(\boldsymbol{r}, t+1) = \tilde{N}_i(\boldsymbol{r}, t) \quad \text{if } \boldsymbol{r} + \boldsymbol{C}_i \in \text{solid}, \quad \boldsymbol{C}_{-i} = -\boldsymbol{C}_i. \tag{49}$$

Let us first consider the condition (49) at order $O(\epsilon^0)$, i.e., when

$$N_{-i}^{\text{eq.}}(\boldsymbol{r}, t) = N_i^{\text{eq.}}(\boldsymbol{r}, t) \tag{50}$$

and, therefore, momentum projection on the link $\boldsymbol{C}_i$ should vanish at $\boldsymbol{r}$

$$(\boldsymbol{j} \cdot \boldsymbol{C}_i)(\boldsymbol{r}, t) = 0. \tag{51}$$

Substitution of the first-order expansion (12) written along a link $\boldsymbol{C}_i$ into the b.b condition (49) holds

$$[(\boldsymbol{j} \cdot \boldsymbol{C}_i) + \Delta \times \nabla(\boldsymbol{j} \cdot \boldsymbol{C}_i)\boldsymbol{C}_i](\boldsymbol{r}, t) = 0 + O(\epsilon^2) + O(M^2) + \cdots, \quad \Delta = \frac{1}{2}. \tag{52}$$

Closure relation (52) locates the walls in the middle between the current node $\boldsymbol{r}$ and the neighbor node $\boldsymbol{r} + \boldsymbol{C}_i$. So, at order $O(\epsilon^1)$:

$$(\boldsymbol{j} \cdot \boldsymbol{C}_i)(\boldsymbol{r} + \frac{1}{2}\boldsymbol{C}_i, t) = 0. \tag{53}$$

Condition (53) is exact for linear flow only, similar as its generalizations [13,58], which annihilate $(\boldsymbol{j} \cdot \boldsymbol{C}_i)$ at a given distance $\Delta \boldsymbol{C}_i$ between $\boldsymbol{r}$ and $\boldsymbol{r} + \boldsymbol{C}_i$. When second-order Chapman–Enskog expansion is substituted into b.b. condition, i.e., the second-order momentum derivatives are taken into account, the analysis [18,21,39] shows that effective wall location depends on the choice of the whole set of the eigenvalues. So far, it depends on the kinematic and bulk viscosities values. It depends also upon wall inclination with respect to the lattice. Modification of momentum definition by $(1/2)\boldsymbol{F}$ [cf. Eqs. (3b) and (10)] enables us to analyze obtained closure relations independently on the force term in Eq. (5) since b.b holds

$$-\frac{1}{2}t_p^\star(\boldsymbol{C}_i, \boldsymbol{F}) \overbrace{+}^{\text{force}} t_p^\star(\boldsymbol{C}_i, \boldsymbol{F}) = \frac{1}{2}t_p^\star(\boldsymbol{C}_i, \boldsymbol{F}) = -\frac{1}{2}t_p^\star(\boldsymbol{C}_{-i}, \boldsymbol{F}). \tag{54}$$

In general flows, effective precision of the b.b. rule is something between first and second order. It is only first-order accurate, however, in inclined channels as shown in [20]. In order to improve the precision of the b.b. boundary conditions, we apply *magic solution* for eigenvalues (A.13)–(A.15a). This solution fulfills exactly closure relation (53) for Poiseuille flow for channels parallel to x, y or z lattice axis. Relation (A.13) represents a particular case of the solution [39], since here all eigenvalues associated with the even (odd) order polynomial basis vectors are set equal between them. The solution (A.13) provides viscosity-independent wall location for Stokes flow in any geometry. When $\lambda_\psi \to -2$, the first-order collision (A.16) is not as precise as the magic collision for b.b rule but is still acceptable, since in the limit $v \to 0$, the influence of second-order terms on the effective wall position goes to zero.

4.2. Free-slip boundary condition

Free-slip boundary condition has not been so intensively studied as the bounce-back conditionfor the LB models. Following Maxwell [6,56], specular reflections are used in the LB methods to model it: when the population arrives on the solid from a boundary node, it reflects into the node symmetric with the respect to the normal to the wall (see Fig. 1). Using first-order Chapman–Enskog expansion [7,12,27], one can show that *specular* reflection at a solid wall provides *free-slip boundary condition*

$$j_{n^s} = 0, \quad \frac{\partial j_{\tau^s}}{\partial n^s} + \frac{\partial j_{n^s}}{\partial \tau^s} = 0, \quad \tau^s = \{\tau_1^s, \tau_2^s\}. \tag{55}$$

Formally, condition (55) holds up to $O(\epsilon^2)$ only when the flow is invariant along a wall. In general then, *local* specular reflection (see Fig. 1) has approximately the same accuracy. We implement it in a form

$$N_{-i}(\boldsymbol{r}, t+1) = \tilde{N}_j(\boldsymbol{r}, t) \quad \text{if } \{\boldsymbol{r} + \boldsymbol{C}_i, \boldsymbol{r} + \boldsymbol{C}_j\} \in \text{solid} \quad \text{and} \quad C_{-in^s} = -C_{jn^s}, \quad C_{-i\tau^s} = C_{j\tau^s}. \tag{56}$$

Relations (56) mean that all populations return into the node of departure. Unlike to bounce-back, force addition in (5) is not consistent with the condition (56) when $\boldsymbol{F}$ is parallel to the wall. To improve this, one should either omit the corresponding force addition to leaving populations $\tilde{N}_j(\boldsymbol{r}, t)$, or to implement specular reflection in its classical *non-local* form, when the populations are reflected into the neighboring nodes. In geometries more complicated than the point near a solid wall, the solution for an unknown population should involve more than one post-collision population. For instance, in the case of "2D" corner (see Fig. 1), we compute unknown "corner" populations, both in 2D and in 3D, as an arithmetical mean of specular reflections with respect to both walls forming the angle. This provides free-slip condition (55) approximately on the both sides. Useful test of free-slip boundary conditions is a uniform Stokes flow in an infinite (periodic) channel. This solution is maintained exactly by the relations (56) in 2D case and by using mentioned above reflections in corners, in 3D case. Similar, free interface algorithm should provide exact solution with density and velocity equal to those at the inlet when constant flux comes into a channel. Linear combination of free-slip/no-slip boundary conditions with some factor $p/(1-p)$ enables us to model intermediate friction behavior.

4.3. Inlet boundary condition

Inlet boundary condition is not trivial even in the case of a constant incompressible flux $\boldsymbol{j}^{\text{in}} = \rho_0 \boldsymbol{U}^{\text{in}}$ entering the domain. Indeed, the density $\rho^{\text{in}}(\boldsymbol{r}, t)$ at the inlet is not equal to its initial value ρ_0 because of the pressure gradients. So far, $\rho^{\text{in}}(\boldsymbol{r}, t)$ is a priori unknown. Moreover, since mass flux $\boldsymbol{j}$ performs the ρ_f-transport for the recoloring algorithm, $\boldsymbol{j}$ should be proportional to the effective ρ^{in} value and, therefore, cannot be set equal to $\boldsymbol{j}^{\text{in}}$. In order to compute ρ^{in}, we use the same idea as for the reconstruction step: all populations are expressed as a first-order Chapman–Enskog expansion where the velocity is set equal to its inlet value. Known populations, arriving at the inlet from the bulk, are used to derive the unknown quantities. In particular, when the velocity derivatives at inlet are negligible (e.g. constant profile) and continuity condition (19a) is assumed, $\epsilon N_i^{(1)}$ (12) written in non-inclined coordinate system becomes becomes

$$\epsilon N_i^{(1)} = \frac{1}{\lambda_\psi} \frac{\partial \rho}{\partial \beta} U_\alpha^{\text{in}} Q_{i\alpha\beta}, \quad i \in \{0, \ldots, b_m\}. \tag{57}$$

Assuming an approximately hydrostatic (linear) pressure distribution at inlet $c_s^2 \partial \rho / \partial z \approx \rho_0 g$, the populations take the form

$$N_i = \rho B_i + b_i, \tag{58}$$

where

$$B_i = r_p^\star + t_p^\star \left[U_\alpha^{\text{in}} C_{i\alpha} + \frac{U_\alpha^{\text{in}} U_\beta^{\text{in}}}{2} (3C_{i\alpha} C_{i\beta} - \delta_{\alpha\beta}) \right], \quad b_i = \frac{\rho_0 g}{c_s^2} \frac{1}{\lambda_\psi} U_\alpha^{\text{in}} Q_{i\alpha z} - \frac{1}{2} \rho_0 g C_{iz}.$$

Computing a sum of known populations $\sum_{i \in I^+} N_i^+$, we write then Eq. (45) for density

$$\rho = \frac{\sum_{i \in I^+} N_i^+ + \sum_{i \in I^-} b_i}{1 - \sum_{i \in I^-} B_i}. \tag{59}$$

When ρ is computed, incoming populations are imposed in the form (58). In case of not uniform inlet profile, the same approach has to include the first- and, if necessary, second-order momentum derivatives into Chapman–Enskog expansion for inlet populations.

5. Overview of the algorithm

We apply our method to simulate filling processes. The scaling procedure is based on the equality of the Reynolds number Re and Froude number Fr to those of the experiment (see (20)). The magnitude of the inlet LB velocity U^{lb}, $U^{lb} = \|U^{\text{in}}\|$, determines the scaling factor between the LB and the physical velocities. Characteristic length L is set equal to some inlet distance. Consider some regular grid which covers the computational domain. Let the number of its liquid cells be equal to V and their number at inlet be S^{in}. The number of LB time steps to fill the box is $T^{lb} = V/(S^{\text{in}} U^{lb})$. Since the computational time is the inverse of proportional to U^{lb}, its value must be as high as possible. On the other side, the conditions $U^{lb} < 1$ and $U^{lb} < c_s$ should be met. Moreover, $M^2 = U^{lb2}/c_s^2$ should be small enough to avoid high compressibility error. For instance, in case when nearly hydrostatic regime $\nabla P' \approx Fr^{-1}$ is attained in a box of a height H, the density difference $\rho - \rho_0$ between the top and the bottom is about $[\rho_0/c_s^2]g^{lb}H$. When the physical velocity increases in l times but the same inlet velocity is used in different LB simulations, g^{lb} and density variation decreases as l^2. So far, simulations corresponding to high physical velocities are easier for the method from the point of view of the compressibility effects caused by the gravitation.

In simulations below, we mostly use $U^{lb} \leqslant 0.1$ and $c_s^2 = 1/3$, i.e., $M^2 \leqslant 0.03$ at inlet, at least. According to von Neumann analysis of the linearized stability [50,76,82], the minimal stable viscosity value $v^{\min}$ increases with U^{lb}. When $U^{lb} = 0.1$, the LB method without free interface approaches its stability boundary somewhere at $\tau^{\min} \approx 0.5078$ in 32^2 and 64^2 periodic boxes. These data correspond to first-order collision (A.16) according to our stability analysis. In case of simulations with free interface and $U^{lb} \approx 0.1$, we detect a loss of stability at approximately this range of τ values, i.e., at moderate Reynolds numbers ($Re \approx 200$–500 for typical inlet length $L^{lb} \approx 10 l.u$). The development of instabilities manifests itself, for example, in appearance of a large number of negative populations after the reconstruction. In fact, local velocities overhead the inlet velocity several times in real-life simulations. The nonlinearity of the flow (see [82]) and the presence of free boundary can shift the stability bounds to greater viscosity values as well.

When the grid is refined by a factor p, i.e., $V \to p^3 V$, $S \to p^2 S$, and U^{lb} is reduced by a factor k, $k \geqslant 1$, T^{lb} increases as $p \times k$ times and the total computational efforts increase accordingly by a factor about $p^3 \times p \times k$. Since Mach number decreases as k^2, one should not expect decreasing of the compressibility effects when $k = 1$ even if the grid is refined. The stability should improve, however, since v^{lb} increases by a factor p/k. In reality, v^{lb} should take so small values for high Reynolds number simulations that it appears to be un-practical to improve the stability only with a help of the refining procedure. In order to analyze the problem, two approaches have been investigated. The first one is to study different reconstruction strategies, including higher-order extrapolations, iteratively improved linearizations, explicit/implicit time approximations, etc. In spite of some improvement, no important gain in stability has been obtained unless

some numerical diffusion is introduced into the scheme. This represents our second approach to stabilize the LB method as discussed in the next section.

6. Upwind approaches

So far three possibilities have been explored in [23]. As a first (*explicit upwind*) approach we add numerical diffusion explicitly as shown below. In the next approach, in order to reduce crosswind diffusion of such an explicit upwind scheme in multi-dimensions, we represent the equilibrium function of the LB equation in such a form that derived macroscopic equations may include different corrections to diffusion tensor. In this way, we introduce LB analogs of full upwind scheme and different streamline type upwind schemes. As a last possibility to damp the small-scale fluctuations, the simplest turbulent (Smagorinsky) model was considered similar to [37]: $v \rightarrow v + v_T$, $v_T = C_s^2||D||$. The intrinsic locality of the LB method is maintained in almost all new schemes since all components of strain-rate tensor $Dj = \rho D$ are derived from non-equilibrium part of the population solution. When a spectrum of global evolution operator is improved for the first and third approaches, the LB method becomes robust and stable. *Explicit upwind scheme* has been found to be the most robust for free interface simulations. Robustness means here that very different realistic problems can be modeled using nearly equal upwind parameters without loss of stability.

The idea of *Pe*-dependent numerical diffusion borrowed from the framework of the conventional approaches in case of 1D convection–diffusion problem (see [10]):

$$v\frac{\partial^2\phi}{\partial y'^2} - UH\frac{\partial\phi}{\partial y'} = 0, \quad U > 0, \ 0 \leqslant y' = y/H \leqslant 1, \ \phi(0) = 1, \ \phi(1) = 0. \tag{60}$$

The exact solution is

$$\phi(y') = \frac{e^{ky'} - e^k}{1 - e^k}, \quad k = \frac{UH}{v}. \tag{61}$$

Exact LBE solution to Eq. (60) is discussed in [30]

$$\phi(y' = jh) = \frac{\chi^j - \chi^n}{1 - \chi^n}, \quad \chi = \frac{1 + Pe}{1 - Pe}, \quad Pe = \frac{Uh}{2v}, \quad h = \frac{H}{n}, \quad j = 0,\ldots,n. \tag{62}$$

This solution will coincide with the solution (61) if

$$\chi^n = \exp^k, \ \text{i.e} \ Pe(v^{\text{eff}}) = \frac{\exp^{k/n} - 1}{\exp^{k/n} + 1}, \ \text{i.e} \ v^{\text{eff}} = \frac{Uh}{2}\frac{\exp^{k/n} + 1}{\exp^{k/n} - 1}. \tag{63}$$

Relation (63) means that the effective LB viscosity should be changed ($v \rightarrow v^{\text{eff}}$) to obtain exact solution for linear convection–diffusion problem. In other words, high-order terms in this flow population solution add the negative diffusion to the viscosity coefficient v, computed from the first-order Chapman–Enskog expansion. In order to compensate it, one can introduce artificial numerical diffusion v^{num}:

$$v^{\text{num}} = v^{\text{eff}} - v = v \times Pe\left(\frac{\exp^{2Pe} + 1}{\exp^{2Pe} - 1} - \frac{1}{Pe}\right) = v \times Pe \times f(Pe), \quad f(Pe) = \coth(Pe) - \frac{1}{Pe}, \quad Pe = \frac{k}{2n}. \tag{64}$$

Relation (63) means that when exactly the same quantity of the numerical diffusion ("optimal rule") as for finite-difference methods [10] is added, LB solution to the problem (60) becomes exact.

With the "explicit upwind approach", we adjust locally λ_ψ so that v becomes equal to v^{eff}:

$$v^{\text{eff}} = v + v^{\text{num}}, \quad v^{\text{num}} = v \times F(Pe), \quad F(Pe) = C \times Pe \times f(Pe), \quad Pe = \frac{\|u\|h}{2v}. \tag{65}$$

Here, the local Peclet number Pe (or grid Reynolds number) controls the quantity of the numerical diffusion; $\|u\|$ is magnitude of local velocity; space step h is equal to 1 l.u.; C is some free parameter. In order to reduce v^{num} at least at small and intermediate Pe numbers, we introduce *modified critical approximation* (cf. [4]):

$$f(Pe) = 0, \quad Pe < Pe^{\text{crit.}}, \quad \text{and} \quad f(Pe) = \left(1 - \frac{Pe^{\text{crit.}}}{Pe}\right), \quad Pe > Pe^{\text{crit.}}. \tag{66}$$

We assume here that an estimation obtained from the stability analysis for maximal stable Peclet number Pe^{max} enables us to fix $Pe^{\text{crit.}}$ value, $Pe^{\text{crit.}} \leqslant Pe^{\text{max}}$. Let $v^{\text{crit.}}(\|u\|)$ corresponds locally to Pe^{crit}: $v^{\text{crit.}} = \|u\|h/2Pe^{\text{crit.}}$. Then we can rewrite relations (65) and (66) as

$$\begin{aligned} v^{\text{num}}(v) &= 0, \quad v^{\text{eff}}(v) = v \quad \text{if } v > v^{\text{crit.}}, \\ v^{\text{num}}(v) &= (v^{\text{crit.}} - v)[C \times Pe^{\text{crit.}}], \\ v^{\text{eff}}(v) &= [C \times Pe^{\text{crit.}}]v^{\text{crit.}} + [1 - C \times Pe^{\text{crit.}}]v \quad \text{if } v < v^{\text{crit.}}. \end{aligned} \tag{67}$$

Relations (67) mean that the numerical diffusion manifests itself only when the kinematic viscosity is less than critical viscosity value at a given velocity. Effective viscosity (67) is represented as a linear combination of $v^{\text{crit.}}$ and v. Its magnitude depends on a product of two values: $Pe^{\text{crit.}}$ and C. When $C \times Pe^{\text{crit.}} = 1$, v^{eff} takes its minimal value $v^{\text{crit.}}(v)$. We study results obtained with a help of explicit upwind scheme in case of one-phase examples (1D convection–diffusion, driven cavity) and benchmark free interface simulations (see [23]). Based on these results, we conjecture that the choice $Pe^{\text{crit.}} = D$ and $C = 1/D$ is close to limit of the admissible numerical diffusion on fine enough grids. In this way, v^{eff} approaches critical approximation [4] to solution [10] in 1D case. Note that in case $U^{lb} = 0.1$, choice $Pe^{\text{crit.}} = 3$ corresponds to $\tau = 0.55 \gg \tau_{\min}$. The assumption that the LB stability limits can be estimated in terms of Pe number is currently under study.

7. Numerical results

7.1. Advection tests

As a simple advection test of the recoloring algorithm, we advance a bubble with a given constant velocity U by using free interface LB method for grids with different resolution. Fig. 2 demonstrates that the bubble shape is maintained. Initially, the active points are those lying inside the bubble ($\rho_{\text{f}} = \rho_0$). In all active points, the population solution stays equal to Stokes equilibrium $N_i^{\text{eq.}}(\rho_0, U)$ after each propagation step. Consequently, the density ρ and velocity u stay equal to their initial values ρ_0 and U. For the convergence study, regular grids $(2^n)^D$, $n \geq n_0$ are used. Space step is set equal to 1 l.u. for every grid. The initial radius of bubble is $7 \times 2^{(n-n_0)}$ l.u., $n_0 = 5$ and its initial position is $r_0 = (2^{n-1})^D$. The position of bubble center $\mathbf{R}(t)$ is approximated as ρ_{f}/ρ-weighted sum of the active cells centers. The error err in bubble center position is computed as the arithmetical mean of its coordinate values

$$\text{err} = \frac{2^{n_0-n}}{D} \sum_{\alpha=1,\dots,D} \sum_t \|(R_\alpha(t) - R_\alpha^{\text{e.}}(t))\|, \quad \mathbf{R}^{\text{e.}}(t) = \mathbf{R}_0 + \boldsymbol{u}t. \tag{68}$$

Error is measured in discrete time moments $t = \{10 \times k \times 2^{(n-n_0)}\}$, $k = 1, \dots, 7$. The norm of the difference $\text{rel}(n) = (\text{err}(n)/\text{err}(n+1))^{1/2}$ is computed for each pair of the consequently refined grids. Linear
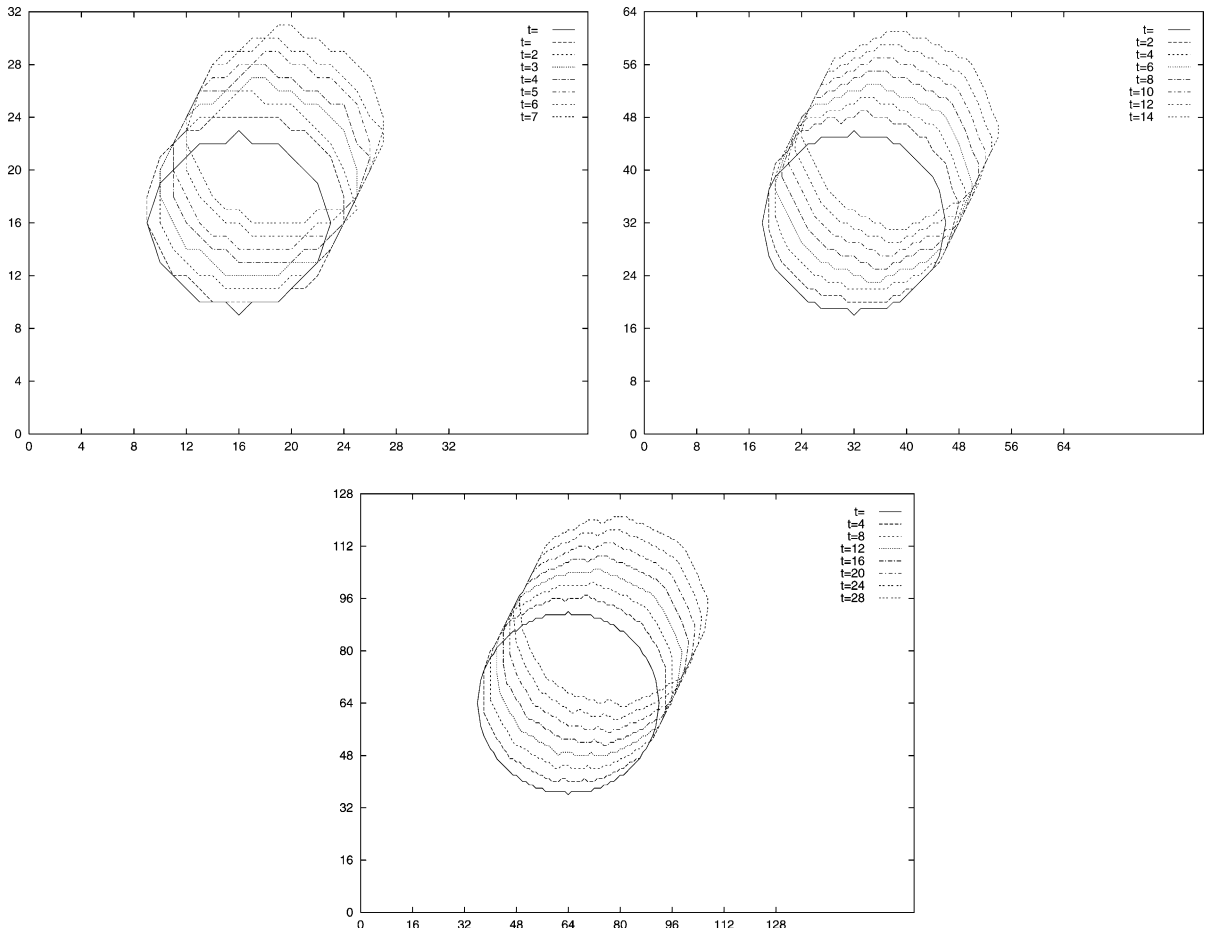
Fig. 2. Advection of 2D bubble with $U = (0.05, 0.1)$. Top to bottom: 32^2, 64^2, and 128^2 grids.

(quadratic) convergence should correspond to $rel(n) = \sqrt{2}$ $(rel(n) = 2)$, respectively. Results obtained in case of different advection velocities and $D3Q15$ model are found in Tables 2 and 3. Convergence rates between first and second order are agree with the prediction of Section 3.2. As it could be expected, the error is anisotropic. Similar results are obtained for other U values and for $D2Q9$ model.

Although analysis of the ILB solutions for simple interfaces has been done (see [1,19]), recoloring step deserves to be studied more attentively. In order to demonstrate how the recoloring step can be analyzed independently from the whole LB algorithm, translation test is included as an example. More complicated

Table 2
Error norm in 3D for bubble advection in case (36)

Velocity U	32^3	64^3	128^3
$U = (0.1, 0, 0)$	0.2464	0.0603	0.0168
$U = (0.1, 0.1, 0.1)$	0.2011	0.0696	0.0321
$U = (0.05, 0.1, 0)$	0.2489	0.1031	0.0301
$U = (0.05, 0.1, 0.05)$	0.2727	0.0923	0.0271

Table 3
Convergence results for data in Table 2

Velocity U	32–64	64–128
$U = (0.1, 0, 0)$	2.0216	1.8953
$U = (0.1, 0.1, 0.1)$	1.7001	1.4733
$U = (0.05, 0.1, 0)$	1.5537	1.8509
$U = (0.05, 0.1, 0.05)$	1.7186	1.8443

advection tests defined by Rider and Kothe [67] are in study [46]. We limit then ourselves by the translation test because of the following reasons.

The benchmark simulations used for conventional interface tracking schemes are more difficult to handle correctly for the recoloring step. The reason for this is that recoloring needs to set population distribution functions. In case of the translation test, the equilibrium function represents exact population solution. For more complicated tests, distribution series requires not only local pressure and velocity values at given time but also their space derivatives. Depending on the truncation order, approximated populations may introduce an additional error at the advection stage. Alternatively, one may model the velocity/pressure benchmark solution with the LB method itself. In this case, the recoloring step is based on the LB population solution similar as in the present paper. One should have in mind, however, that the obtained velocity solution may differ from the benchmark solution due to the numerical error in the LB method itself. When necessary, one should then separate the advection error from the velocity error. Theoretical analysis of the recoloring step could reveal at which order the Chapman–Enskog expansion of the population distribution can be truncated in a consistent manner. Moreover, it could help to understand how the recoloring could be efficiently coupled with the other interface solvers.

7.2. Filling in 2D cavity

We consider first filling simulations in 2D cavity with expansion 1:5. Inlet section is 2 cm × 7.8 cm, the cavity is 10 cm × 20 cm; inlet velocity is equal to 100 cm/s, filling time T is 1.08 s. Gravitation is absent: $g = 0$. Density of fluid ρ^{exp} is 1 g/cm^3. We vary Reynolds number Re with viscosity. No special efforts to maintain the symmetry is done. We show the obtained results in Figs. 3–6 for $Re = 0.2, 2, 50, 500$, respectively. In whole, filling patterns are in agreement with the theoretical and the numerical analysis [2]. At $Re = 0.2$, the "mound filling" is observed. At $Re = 2$, the filling behavior is changed and "disk pattern"

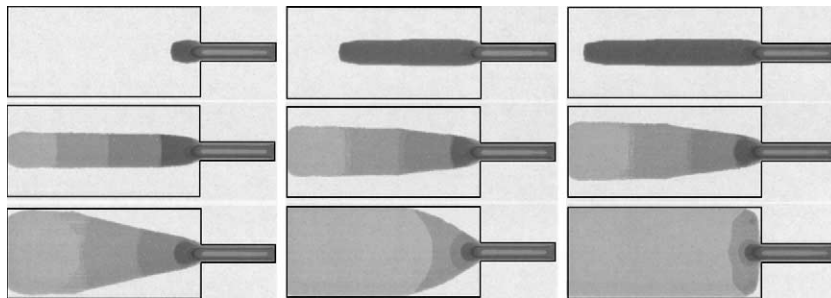
Fig. 3. LB simulation of filling of a 2D cavity at $Re = 0.2$. The figure shows the time-evolution of the velocity magnitude distribution in the cavity (left to right then top to bottom, $t = 0.11, 0.27, 0.32, 0.43, 0.54, 0.59, 0.75, 0.92$, and 1.02 s). Physical parameters are: $U = 100$ cm/s, $v^{exp} = 1.0$ cm^2/s, $T = 1.08$ s, and $L^{exp} = 2$ cm. The number of cells is 86,240. The parameters used in the simulation are: $U^{lb} = 0.00625$, $v^{lb} = 1.25$, $\tau = 4.25$, $L^{lb} = 40$, and $T^{lb} = 344,960$. No-slip boundary conditions are applied at walls and the magic collision of Eq. (A.13) is also used. Colors: light gray in inlet section (214–232 cm/s), dark gray in inlet section (125–143 cm/s), and gray (<71 cm/s).

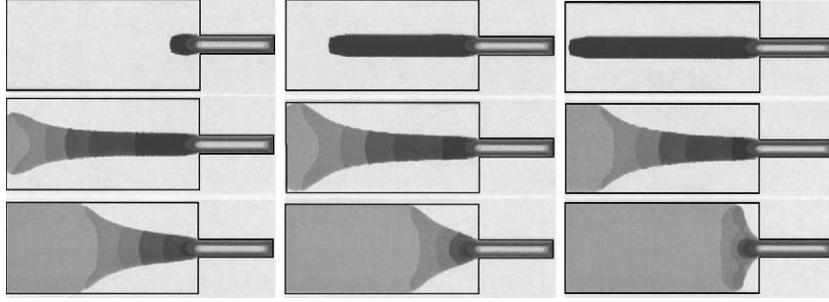Fig. 4. LB simulation of filling of a 2D cavity at $Re = 2$. The viscosity is $v^{exp} = 0.1 \text{ cm}^2/\text{s}$ ($\tau = 0.875$). Other parameters and conditions remain the same as in Fig. 3. Colors: light gray (157–171 cm/s), dark gray (100–114 cm/s), and gray (<43 cm/s).
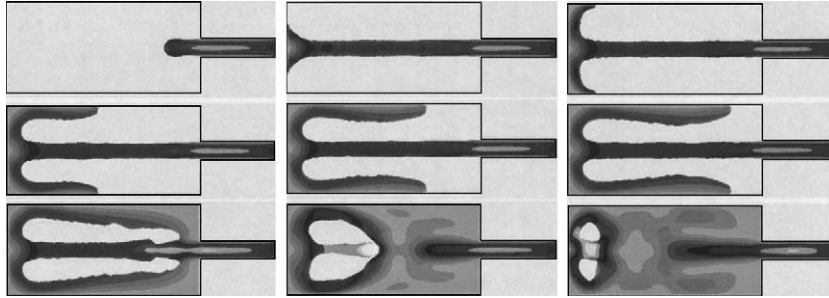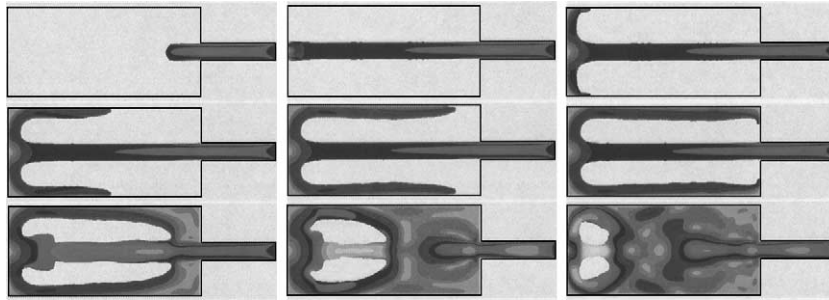

Fig. 5. LB simulation of filling of a 2D cavity at $Re = 50$. The viscosity is $v^{exp} = 0.04 \text{ cm}^2/\text{s}$ ($\tau = 0.74$), $U^{lb} = 0.1$, and $T^{lb} = 21,560$. First-order collision (A.16) is used. Other parameters and conditions remain the same as in Fig. 4. Colors: light gray (196–232 cm/s), dark gray (125–143 cm/s), and gray (<71 cm/s).

Fig. 6. LB simulation of filling of a 2D cavity at $Re = 500$. The viscosity is $v^{exp} = 0.004 \text{ cm}^2/\text{s}$ ($\tau = 0.524$). Friction factor p is set to 1/2. Other parameters and conditions remain the same as in Fig. 5. Colors: light gray (196–232 cm/s), dark gray (125–143 cm/s), and gray (<71 cm/s).

develops. Relatively small LB velocities are used in both cases in order to decrease LB viscosities and, therefore, to improve an accuracy of boundary conditions.

At intermediate and high Re, when inertia dominates, filling patterns change drastically and so-called "shell" type filling is obtained at $Re = 50$ and $Re = 500$. At $Re = 50$, viscous boundary layers are rather thick (see Fig. 5). At $Re = 500$, the boundary patterns are much thinner and they develop almost parallel to adjacent wall, in according to the analysis of inviscid flow [2]. Similar solution are obtained at $Re > 500$,
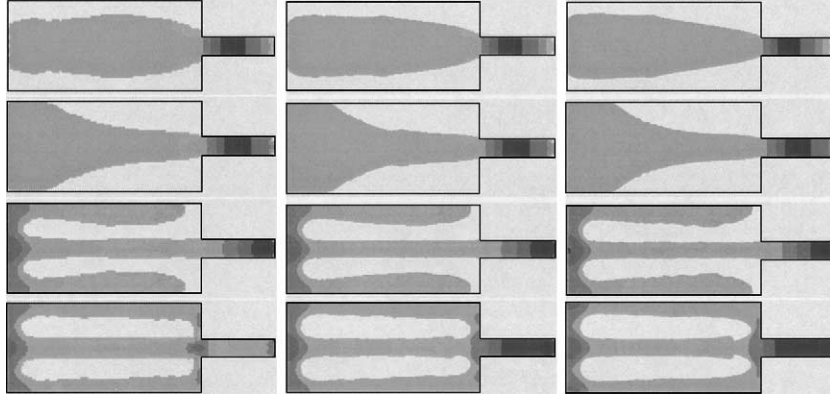
Fig. 7. Pressure solutions in 2D cavity at $t = 0.65$ s with different space resolutions (left to right, space step is: $h = 0.2$; $h = 0.1$; $h = 0.05$ cm) and different Re numbers (top to bottom, $Re = 0.2, 2, 50, 500$). Reference pressure of gas phase at interface is $P_0 = 1013$ mbar. Colors: $Re = 0.2$, light gray near the inlet (3622–4096 mbar), dark gray (2673–2910 mbar), and gray (<1724 mbar); $Re = 2$, light gray near the inlet (1327–1356 mbar), dark gray (1213–1242 mbar), and gray (<1099 mbar); $Re = 50$, dark gray (1036–1033 mbar), and gray (<1025 mbar); $Re = 500$, dark gray (1048–1043 mbar), and gray (<1033 mbar).

when we use explicit upwind scheme (67) with $C = 1$ and $Pe^{crit.} = 1$. When this scheme is applied in case $Re = 50$, no influence on the solution is detected since the actual Pe-numbers are less than $Pe^{crit.}$. We conclude then that for chosen parameters of the upwind scheme, the quantity of the numerical diffusion is acceptable.

The convergence behavior of the algorithm with respect to the space resolution is checked by considering three consequently refined grids. In so far, the solution above corresponds to finest grid of the sequence. The results are displayed in Fig. 7 for pressure solution. At given Re number, equal inlet LB velocities are used for simulations at every grid. Then the CFL value is constant (CFL $= U^{lb}\Delta t^{lb}/\Delta x^{lb}$, $\Delta t^{lb} = 1$, $\Delta x^{lb} = 1$) and the value of time step in physical units decreases together with the space step when the grid is refined. Fig. 8 displays point-wise difference $L_{h,2h}(t)$ between the solution obtained on the grid with step $2h$ and its projection from the finer grid, measured in L_1 norm: $L_{h,2h}(t) = \|f_h(t) - f_{2h}(t)\|$. Projection is set equal to an arithmetical mean of the four fine cells lying inside one coarse cell. The solution is put equal to zero in non-filled cells. The results are given for pressure (mbar), velocity (cm/s), and phase-distribution variable $\rho_f/\rho : f = \{P, \|u\|, \rho_f/\rho\}$. The last figure in each row plots the error ratio $L_{h,2h}(t)/L_{2h,4h}(t)$ for these variables. The error is measured each 5% of filling. Mean ratio value is about 2 what corresponds to first-order convergence. Table 4 displays integrated over the whole period of filling time convergence rates $L_{h,2h} = \sum_t L_{h,2h}(t)$ vers $L_{2h,4h} = \sum_t L_{2h,4h}(t)$.

The results for convergence rates reflect quite well main features of the current algorithm with respect to mesh refinement. First, only first-order convergence is observed. In 2D case, since no tangential derivatives are neglected, second-order accuracy is met by Chapman–Enskog expansion at interface as well as in bulk. However, we can not expect effective second-order behavior from the boundary conditions at solid walls used here. Also, the advection scheme and the calculations of the normal are only first-order accurate. Second, the difference between the solutions obtained on three grids is smaller for intermediate Reynolds numbers, $Re = 2$ and $Re = 50$. At $Re = 0.2$, i.e., at high τ values ($\tau > 1$), the difference between coarse/fine solutions at the stagnation point is quite significant. We relate this to inaccuracy of the boundary conditions which grows together with v (see [18,39]). For $Re = 500$, if equal parameters of upwind scheme ($Pe^{crit.} = 1$, $C = 1$) are used on every grid, the numerical diffusion on coarse grids becomes excessive. This can be understood from the relation (67): since $Pe^{crit.}$ and U^{lb} values are equal at all grids, $v^{crit.}$ values are also close, whereas the imposed viscosity values v increases with the refinement. This implies higher v^{num} values for

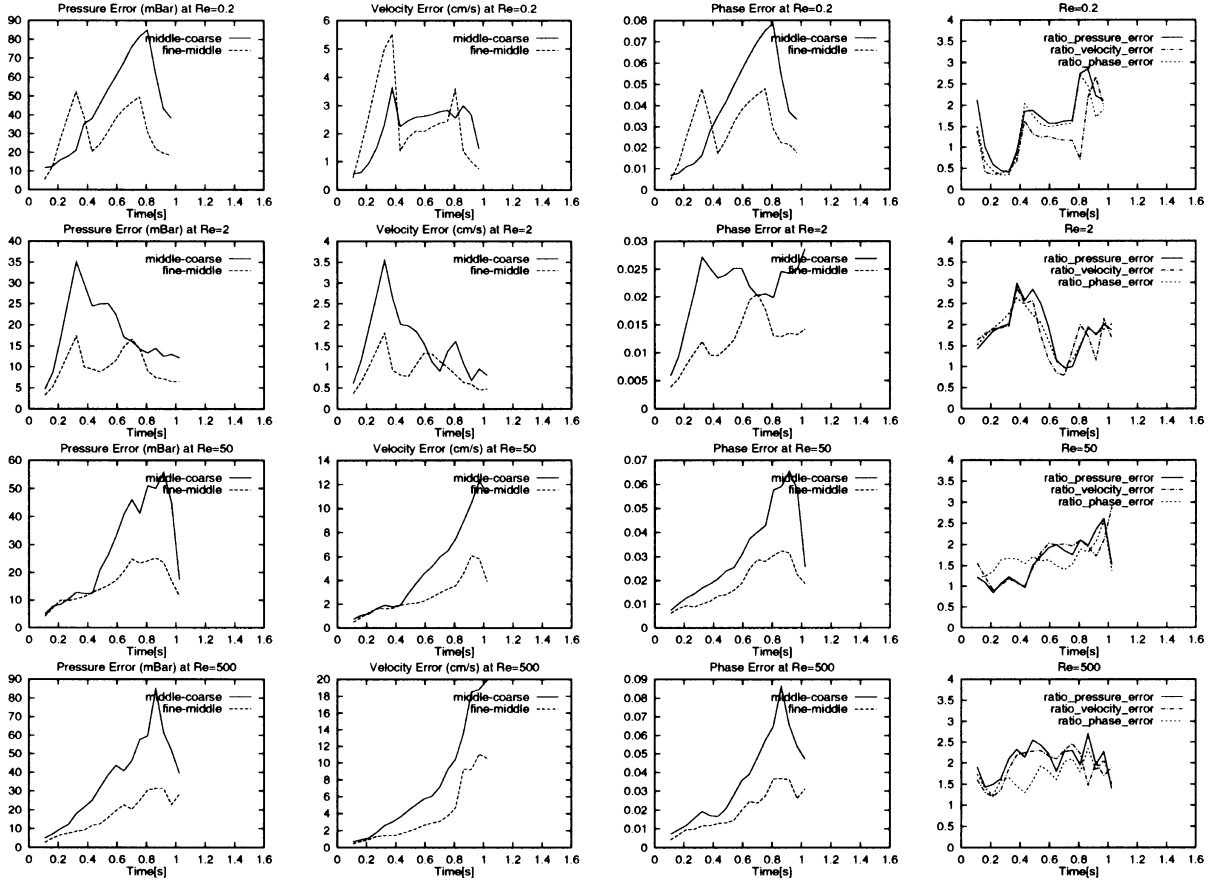Fig. 8. Left to right, magnitude of error value for pressure, velocity magnitude, and fluid quantity ρ_{f}/ρ between coarse-middle and middle-fine grids in 2D cavity simulations. The last figure in each row plots the error ratio (coarse-middle to middle-fine) for these variables. Top to bottom: $Re = 0.2, 2, 50, 500$. Data correspond to previous picture.

Table 4
Error norm $L_{h,2h}/L_{2h,4h}$ in 2D cavity

Re	Pressure (mbar)	$\|u\|$ (cm/s)	ρ_{f}/ρ
0.2	1295.9/553.4	49.44/41.7	0.705/0.51
2.0	330.7/180.4	28.45/16.5	0.39/0.22
50	496.9/282.5	89.6/48.6	0.57/0.33
500	654.68/311.5	144.1/69.09	0.98/0.36

smaller v values, i.e., for coarse grids. One could assume then that the parameter C should be reduced together with v. The calculations on the bottom of the Fig. 7 are performed with increasing C values ($C = 1/4, 1/2, 1$), from coarser to finest grids. The results improve then according to our predictions (see at the end of Section 6). We note, however, that some thickening appears when the boundary flux drains into the inlet column (see the bump at the left of the inlet on right bottom picture in Fig. 7); this thickening continues to travel with the fluid (see the left bottom pictures in Figs. 5 and 6). We conjecture that this is related to coupling of the populations which carry fast and slow momentum values at interface cells. This
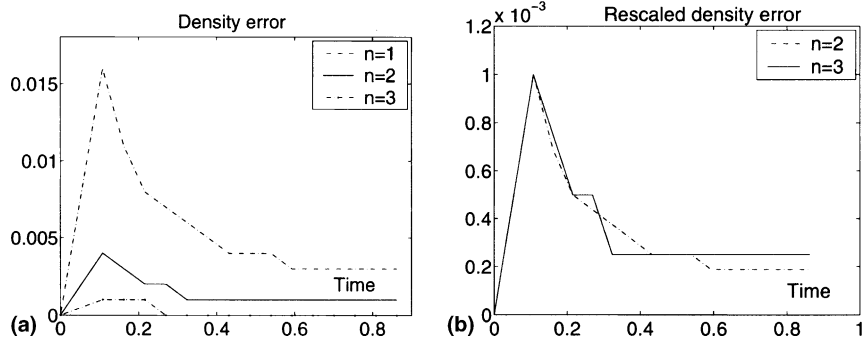
Fig. 9. Density deviations from the reference value at $Re = 2$, $L^{lb} = 40$. Results are plotted for LB inlet velocities $U^{lb} = 0.1 \times 2^{-n}$, where $n = 2, 3, 4$. Accordingly, $v^{lb} = 2 \times 2^{-n}$, $\tau = \{2, 1.25, 0.875\}$. (a) $\delta\rho = (\bar{\rho} - \rho_0)$; (b) $\delta_\rho^s = (\bar{\rho} - \rho_0) \times 4^{4-n}$.

effect is less visible on coarse grids (see bottom pictures in Fig. 7) due to excessive numerical diffusion and/ or lack of space resolution. Similar results are obtained with the SPH approach by Kuhnert and Tiwari [44]. We suppose that adding surface tension would smooth the solution.

Compressibility study is performed for $Re = 2$ when inlet velocities varies: $U^{lb} = 0.1 \times 2^{-n}$, $n = 0, 1, 2, 3, 4$. In two first cases, i.e., at high τ values ($\tau = \{7.5, 3.5\}$), the solution is neither accurate nor stable. In the three other cases, we compute mean density value $\bar{\rho}(t)$ over all active points and compare it with the reference value ρ_0. We plot in Fig. 9(a) obtained results for $\delta\rho$ value, $\delta\rho = (\bar{\rho} - \rho_0)$. In order to check if $\delta\rho(n)$ scales as $U^{lb^{2n}}$, we rescale $\delta\rho$ with respect to its value at $n = 4$. Fig. 9(b) displays δ_ρ^s values, $\delta_\rho^s = (\bar{\rho} - \rho_0) \times 4^{4-n}$. When $n = 4$, $U^{lb} = 0.1/2^4$, $M^2 \approx 1.2 \times 10^{-4}$, $\delta\rho \approx 0$, i.e., incompressible regime is practically reached. After rescaling, density deviations $\delta_\rho^s(n)$ approaches to zero, similar to the results for $n = 4$. This confirms that the compressibility errors scales as M^2, in agreement with the theoretical predictions.

7.3. Three-dimensional simulations

Benchmark simulations: Hammer box [36], Campbell box [72], and Sheffield box [3] are presented. The influence of inlet velocity on the compressibility is considered in ''Motorblock'' simulations. Density of fluid ρ^{exp} is 1 g/cm^3 unless specially indicated. Since very high Re numbers are modeled, free-slip boundary conditions are mostly used. No-slip boundary conditions correspond to high local velocities in narrow channels and lead to further increase of the compressibility. Filled volume is computed as a sum of vof-type value $m_f = \rho_f/\rho$. The deviation of the obtained filling state in time from the exact linear solution is controlled. In benchmark simulations below, corresponding compressibility error lies within 5%.

Regular computational grids used here include from 10^5 to 2×10^6 liquid cells. Similar results are obtained by using both linearizations discussed in Section 3.3. The code is parallelized using Dynamical Load Balance strategy [45]. Since the non-local operations (compared with one-phase LB method) are concentrated at interface cells only (e.g. calculation of normal vectors, advection of fluid mass, extrapolations), the method keeps its advantages for parallelization.

Hammer box [36]. We show in Figs. 10 and 11, pressure and velocity fields during mold filling simulations of steel hammer head casting at $Re = 53{,}417$, $Fr = 5.1$. LB simulations at $U^{lb} = 0.1$ are done with explicit upwind scheme (67), where $Pe^{crit.} = 3$ and $C = 1/3$. The filling sequence agrees quite well with the other results [36,52]. The stream reaches first the right wall at a the point which lies approximately at a height equal to 2/3 of the distance between the runner and the bottom. The jet of failing steel attains the velocity 250–260 cm/s, then it slows down at the bottom and raises slow into the casting box. During the
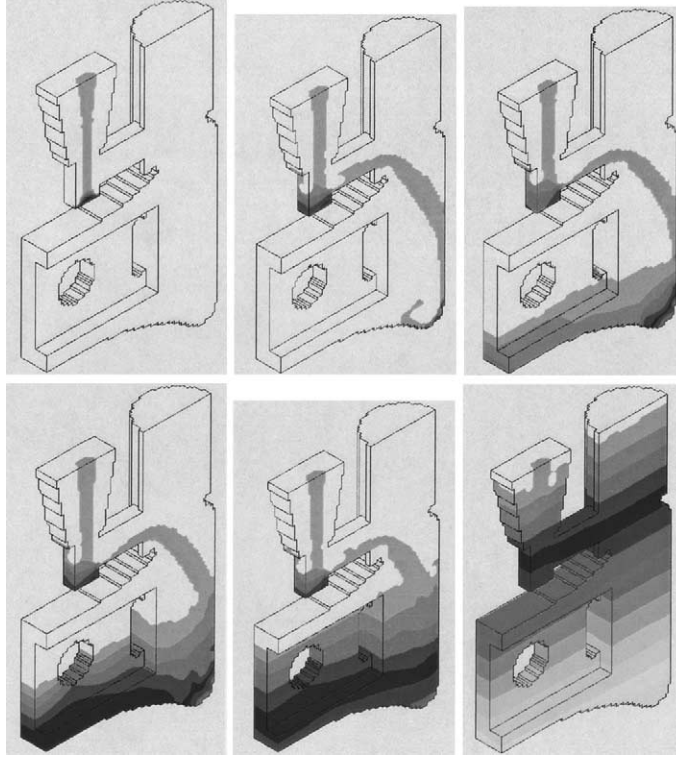
Fig. 10. Filling sequence of pressure distribution in Hammer Box, close to symmetry plane, at $Fr = 5.1$, $Re = 53,417$ (left to right and top to bottom, 1.25%, 7.5%, 25%, 35%, 50%, 100% of exact filling). Physical parameters are: $U = 122.859$ cm/s, $v^{exp} = 6.9 \times 10^{-7}$ m^2/s, $T = 15$ s, $L^{exp} = 3$ cm. Grid: 110,573 liquid cells. LB: $U^{lb} = 0.1$, $v = 1.1 \times 10^{-5}$, $\tau = 0.500034$, $L^{lb} = 6$, $T^{lb} = 36,858$. Colors: light gray (1339–1366 mbar), dark gray (1203–1230 mbar), and gray (<1122 mbar).

rise, the pressure reaches the hydrostatic distribution. When the numerical diffusion increases and viscous/gravitation effects prevail over the inertia, the stream comes into the runner and falls down (see [23]). In so far, this test can be used as a measure of the effective Reynolds numbers. Also, because of very small LB viscosity values used in this experiment, local Pe numbers take mostly high values. Indeed, $Pe^{crit.} = 3$ corresponds to $\|u\| \approx 7 \times 10^{-5}$ in this experiment. One can assume that numerical diffusion can be switched on at higher $Pe^{crit.}$ numbers. For instance, the results at $Pe^{crit.} = 150$, $C = 1/3$ (i.e., $\tau^{crit.}_{\|u\|=0.1} = 0.501$) are still similar to those presented in Figs. 10 and 11. On the other hand, the stabilization is not strong enough when $Pe^{crit.} = 10^3$, i.e., $\tau^{crit.}_{\|u\|=0.1} = 0.50015$). This agrees with the stability values mentioned above. The work in progress should help to estimate a priori effective τ and/or Pe stable values in function of other physical parameters.

Analysis of the algorithm is applied to current example. The number of interface points is of order of several thousands at each time step (see Fig. 12(a)). Figs. 12(b)–(d) display the number of points where at least one unfavorable situation mentioned at caption happens. The number of "bad" cases is negligible compared with the total number of points where the reconstruction takes place. Extrapolations of populations from neighboring "good" cells are performed when situations (b), (c) or (d) happen. Figs. 12(e) and (f) display the number of such points. If no such neighbor is found, the point is deactivated. The total number of deactivated nodes over the whole period of filling is equal to six in this example. Due to the stabilizing scheme, no negative population after collision appears. Otherwise the number of negative
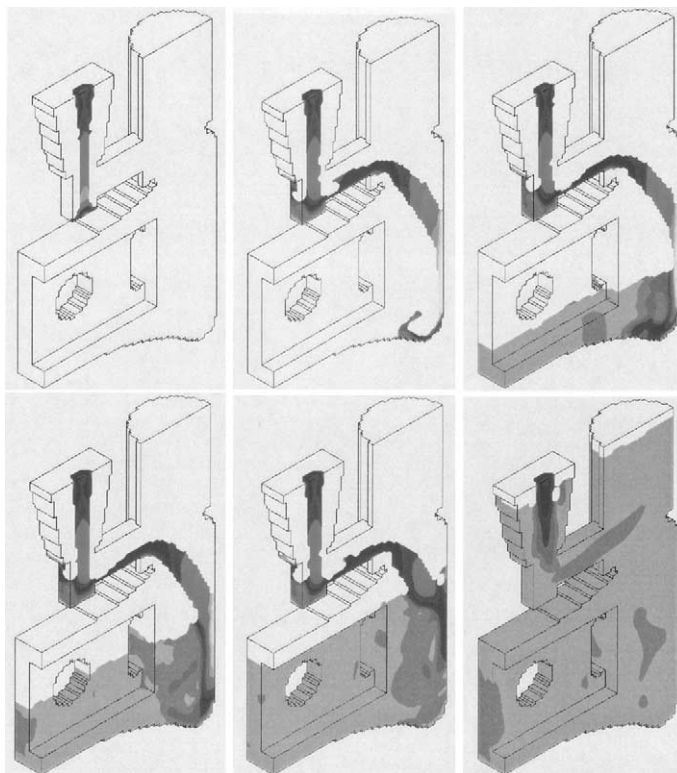
Fig. 11. Filling sequence of the velocity magnitude distribution in Hammer Box. Parameters are the same in Fig. 10. Colors: light gray where the stream reaches the right wall (257–279 cm/s), dark gray (150–171 cm/s), and gray (<86 cm/s).

populations after the reconstruction and after the collision increase drastically when τ approaches its limit value $\tau = 0.5$.

Campbell box [72]. Numerical and experimental results in this geometry (see Figs. 13 and 14) have been discussed at VII Modeling of Casting and Welding Processes Conference. We model the mold casting by using constant inlet velocity which corresponds approximately to the prescribed filling time. Simulation results at $Re = 3.2$ are shown in Fig. 13. They agree well with the polymer flow predictions [72]. Filling sequence at $Re \approx 165$ is plotted in Fig. 14. Here, the sprue develops fast along the bottom of the runner, then impacts to the nearest side of the gate and expands first to the left. Then the sprue fountains quickly to the right. Later, two vortices appear on the either side of the main filling stream. In this way, the simulations reproduce the main features of the experimental results [72]. Note that the kinematic viscosity of the aluminum (and hence Re number) is reduced here, since no turbulent modeling is used in the simulations.

Sheffield box [3]. The simulations at $Re = 24,717$, $Fr = 10.7$ in Sheffield box are displayed in Fig. 15 in case when $U = 145$ cm/s. They correspond to physical parameters of water. Flow comes from left to right and the variation in inlet velocity results in different values of maximal height of the jet column in the right gate. Our results at inlet velocities $U = 80, 95, 105, 145$ cm/s agree well with the available experimental data [3] and the numerical simulations [52]. For all inlet velocities, we use the same upwind parameters: $Pe^{\text{crit}} = 3$, $C = 1/3$. When C increases to 1, however, right water jet does not reach the top wall at $U = 145$ cm/s, indicating that the gravitation and viscous forces dominate over the convective ones (see [23]). Similar to the jet behavior in the Hammer box, this test is a good indicator of the excessive numerical diffusion.
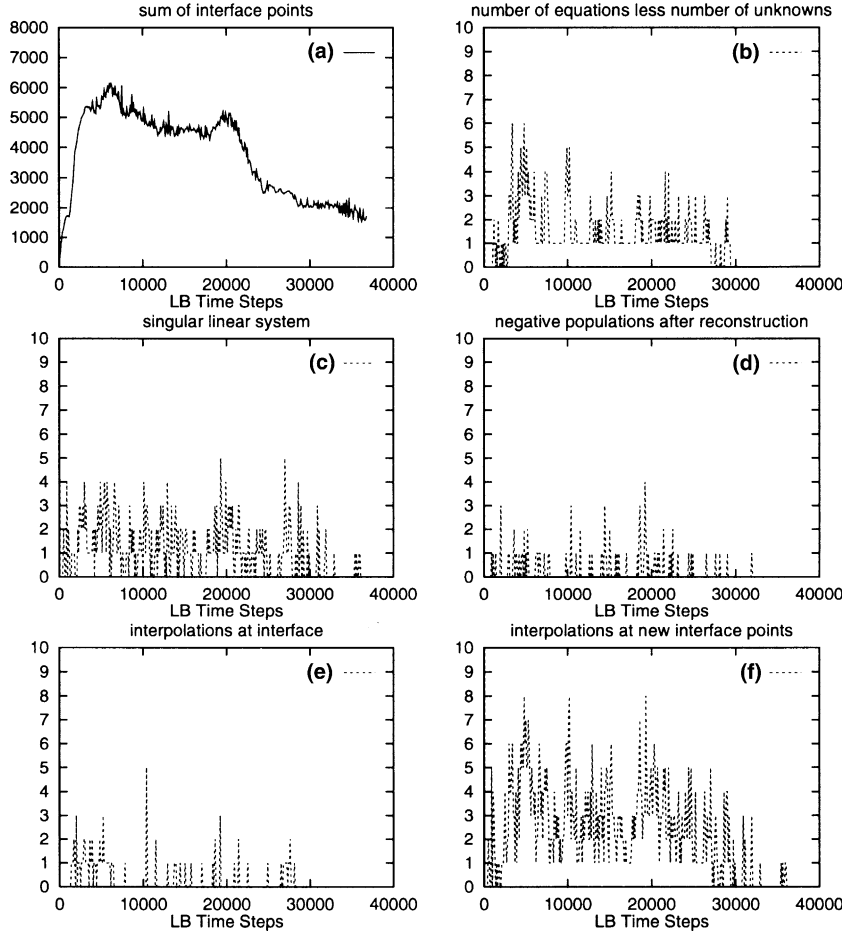
Fig. 12. Analysis of the algorithm for Hammer Box: (a) total number of interface points; (b) number of equations m is less than number of variables n_v; (c) $m \geqslant n_v$ but the linear system is singular; (d) number of cells where negative populations appear after the reconstruction; (e) number of interface cells where interpolations need; (f) number of new interface cells where interpolations need. Here, only interpolations because of (b)–(d) are accounted.

Filling sequence in "Motorblock" at $Re = 26,507$, $Fr = 2.36$ is shown in Fig. 16. The results are obtained with strong upwind parameters $Pe^{\text{crit.}} = 1$ and $C = 1$. The compressibile effect is much stronger here than for the Hammer and Sheffield boxes when the same inlet velocity ($U^{lb} = 0.1$) is used. It causes, in particular, a quite noticeable delay in filling time. The filling state (in percent of full stand) is plotted as a function of time in Fig. 17(a) for $U^{lb} = 0.1$ and $U^{lb} = 0.025$. Fig. 17(b) displays the error in filling state divided by factor four for $U^{lb} = 0.1$. Since both solution are close we can conclude that the error in filling time scales with M rather than with M^2. Note that even for $U^{lb} = 0.025$, the pressure in narrow channels (see inlet channels at two last pictures in Fig. 16, for instance) is still too high. The compressibility of the method is controlled by the choice of the LB characteristic velocity at the inlet: while reducing U^{lb}, pressure solution improves and correct filling time approaches. Nevertheless, an efficient strategy to maintain reasonable Mach numbers in realistic calculations needs still to be found.
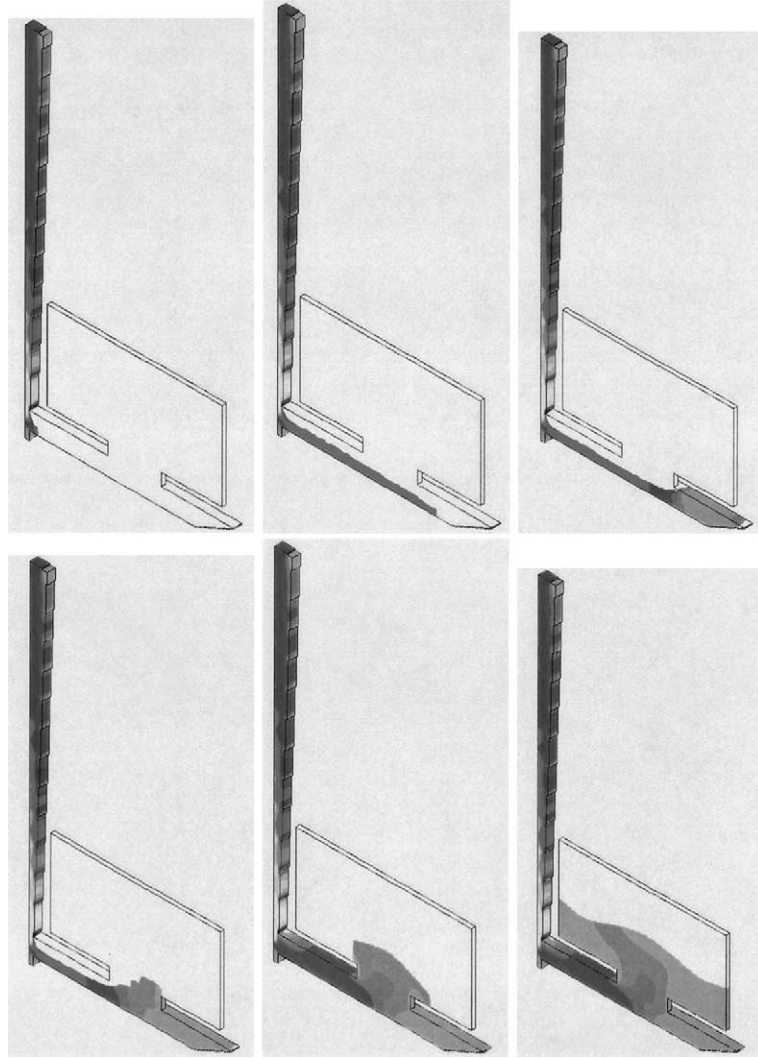
Fig. 13. Velocity magnitude in Campbell Box at $Fr = 5.56$, $Re = 3.2$ (left to right and top to bottom, $t = 0.14, 0.24, 0.28, 0.33, 0.38, 0.47$ s) Physical parameters are: $U = 88.6$ cm/s, $v^{\exp} = 4 \times 10^{-3}$ m^2/s, $T = 1.88$ s, $L^{\exp} = 1.44$ cm. Grid: 216,546 liquid cells. LB: $U^{lb} = 0.0125$, $v^{lb} = 0.047$, $\tau = 0.641$, $L^{lb} = 12$, $T^{lb} = 111,049$. Free-slip boundary conditions. Colors: light gray close to the narrowest sections (214–232 cm/s), dark gray (125–143 cm/s), and gray (<71 cm/s).

8. Conclusion

A general approach for free interface Lattice Boltzmann method has been described. This approach is based on a first-order Chapman–Enskog expansion of the population at interface nodes. Boundary conditions at curvilinear interfaces are exactly met by the coefficients of the series. Interface advection is performed with help of locally mass conserving and anti-diffusive recoloring algorithm. Since no stage of the algorithm involves geometrical interface constructions, the method is robust to any interface topology and can be regarded as a surface capturing method. In bulk, second-order LB accuracy in space is maintained. At the interface, formal second-order accuracy is kept by the expansion. At solid boundaries,
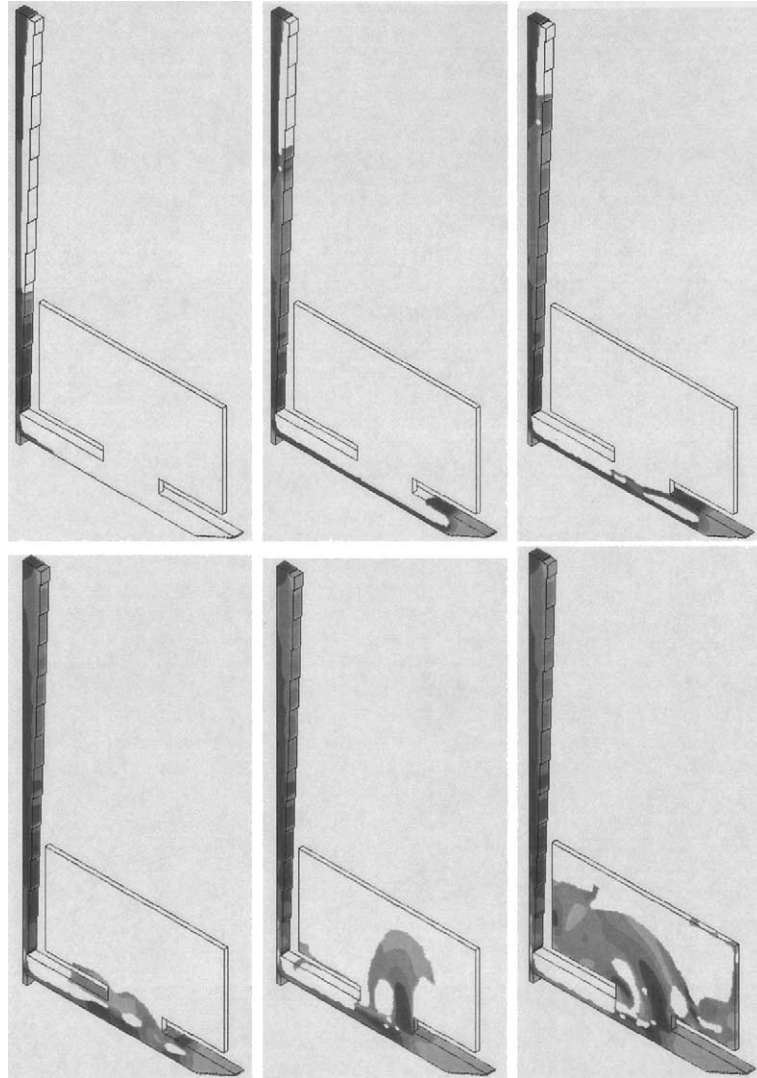
Fig. 14. Velocity magnitude in Campbell Box at $Fr = 5.965$, $Re = 165$ (left to right and top to bottom, $t = 0.09$, 0.18, 0.227, 0.32, 0.35, 0.45 s). Physical parameters are: $U = 91.75$ cm/s, $v^{\exp} = 8 \times 10^{-5}$ m^2/s, $T = 1.815$ s, $L^{\exp} = 1.44$ cm. Grid: 216,546 liquid cells. LB: $U^{lb} = 0.05$, $v^{lb} = 0.047$, $\tau = 0.641$, $L^{lb} = 12$, $T^{lb} = 27,762$. Free-slip boundary conditions. Colors: light gray close to the narrowest sections at bottom pictures (314–371 cm/s), dark gray (200–229 cm/s), and gray (<114 cm/s).

actual accuracy of local reflections is something between first and second order. Boundary method [20] can be incorporated for further improvement. Least square minimization procedure [62,63] could bring second order improvement of the normal calculations on regular grid. From the point of view of the numerical efficiency and adaptation to parallel calculations, the method is not essentially different from the ILB methods. In so far, locality of its main operations and linear increase of the computational efforts with space refining are advantageous for realistic calculations. Since no complicated discretization/advection/ solution procedure is needed, the method can be easily implemented by the LB users and novices. First-order Chapman–Enskog expansion of the populations, which contains in itself all components of the strain
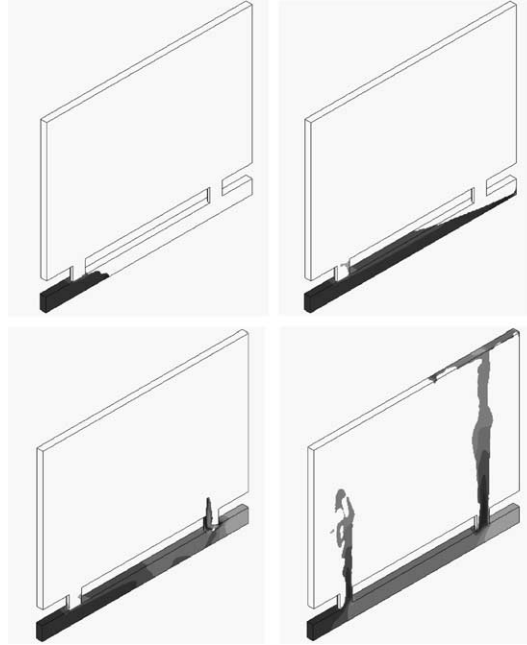
Fig. 15. Velocity magnitude in Sheffield box at $Fr = 10.7$, $Re = 24,717$ (left to right and top to bottom, $t = 0.06$, 0.17, 0.23, 0.4 s). Physical parameters are: $U = 145$ cm/s, $v^{\exp} = 1.17 \times 10^{-6}$ m²/s, $T = 2.31$ s, $L^{\exp} = 2$ cm. Grid: 1,270,420 liquid cells. LB: $U^{lb} = 0.1$, $v^{lb} = 8.1 \times 10^{-5}$, $\tau = 0.500243$, $L^{lb} = 20$, $T^{lb} = 33,432$. Free-slip boundary conditions. Colors: dark gray (250–286 cm/s), and gray (<143 cm/s).

tensor, allows local and simple incorporation of viscoelastic effects into the model. In particular, we combine the LB filling algorithm with the regularized Bingham model [2]. First results [24] are found in good agreement with the theoretical and the numerical predictions.

Besides other straightforward extensions of the method, for instance to other LB velocity models or to two (or more) fluids, several problems remain. The first one is related to intrinsic compressibility of the method. Robust applications of the method in complex geometries require to develop adaptive strategy for dynamic change of the LB parameters and introduce variable space resolution into the model. The second difficulty is observed in filling simulations at very small Reynolds numbers, e.g. processing of metal alloys. We conjecture that the reason lies in inaccuracy of first-order Chapman–Enskog approximation and/or boundary conditions in the limit of high LB viscosities ($\tau > 1$). Although the problem can be avoided by restriction of the LB viscosities to their reliable interval, the corresponding reduction in LB velocities slows the method. Finally, effective and accurate design of LB upwind schemes needs further investigation. Despite these difficulties, the method seems very promising for real-life simulations in injected molding provided that the conditions on its accuracy are met by a proper choice of the numerical parameters.

Two-phase methods are required in situations where free interface hypotheses are not valid and/or the description of the light phase is desirable (e.g. entrainment of air). We believe that the main ideas of the free interface method can be used to analyze and, maybe, to design two-phase LB methods. In particular, existing two-phase LB models implicitly impose interface boundary conditions. Their analysis can be performed based on the coefficients of the series solutions of the obtained distribution functions coming from both sides of the interface. Computing of these coefficients in the interface coordinate system can be done in analogy to reconstruction step in the free interface algorithm.
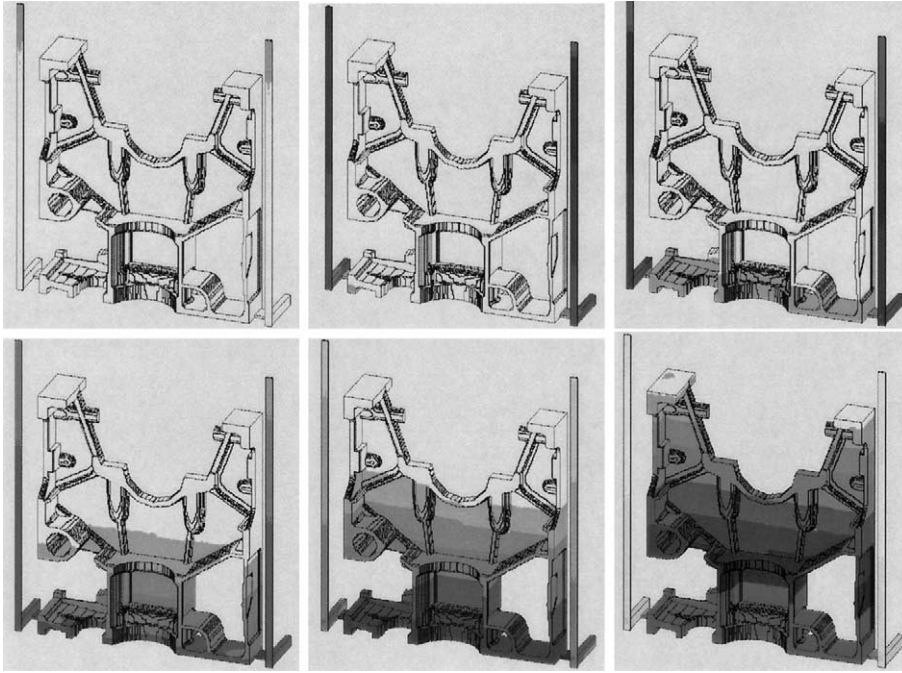
Fig. 16. Pressure distribution is shown at Motorblock at $Re = 26,507$, $Fr = 2.36$ (left to right and top to bottom, $t = 0.3$, 2.5, 7.52, 15, 20, 25 s; accordingly, 1.25%, 10%, 30%, 60%, 80%, 100%.) Physical parameters are: $U = 83.23$ cm/s, $v^{exp} = 9.42 \times 10^{-6}$ m²/s, $T = 30.08$ s, $L^{exp} = 3$ cm, $\rho^{exp} = 7$ g³/cm³. Grid: 625,817 liquid cells. LB: $U^{lb} = 0.025$, $v^{lb} = 6.0 \times 10^{-6}$, $\tau = 0.500017$, $L^{lb} = 6$, $T^{lb} = 200,261$. Free-slip boundary conditions. Colors: light gray (2181–2500 mbar), dark gray (1765–1863 mbar), and gray (<1438 mbar).
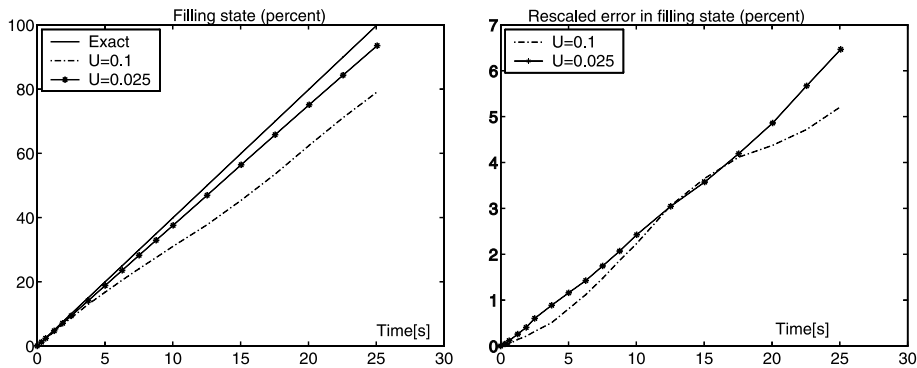

Fig. 17. Comparison of the effective filling of the box with the exact filling in case $U^{lb} = 0.025$ and $U^{lb} = 0.1$. Left: filling state (percents). Right: absolute error is divided by factor four in case $U^{lb} = 0.1$.

Acknowledgements

Appendix A. Generalized lattice Boltzmann equation

In the first two sections, we present basis vectors and eigenvalues for $D2Q9$ and $D3Q15$ models. This is followed by common remarks and particular solutions for free eigenvalues in Section A.3. Details to implementation of collision step are discussed in Section A.4. First-order term $\epsilon N_i^{(1)}$ is obtained in Section A.5 using notations of current paper.

A.1. Model D2Q9

Let nine velocities of the $D2Q9$ model be ordered as following: $(0,0)$, $(1,0)$, $(0,1)$, $(-1,0)$, $(0,-1)$, $(1,1)$, $(-1,1)$, $(-1,-1)$, $(1,-1)$. Orthonormal basis vectors for $D2Q9$ can be chosen in the form (cf. relation (12))

$$\mathbf{e}_1 = \frac{1}{3}\{C_i^0\}, \quad \mathbf{e}_2 = \frac{1}{\sqrt{6}}\{C_{ix}\}, \quad \mathbf{e}_3 = \frac{1}{\sqrt{6}}\{C_{iy}\},$$

$$\mathbf{e}_4 = 6\left\{t_p^{\star} C_{ix} C_{iy}\right\}, \quad \mathbf{e}_5 = 3\left\{t_p^{\star}\left(C_{ix}^2 - \frac{1}{2}c_i^2\right)\right\},$$

$$\mathbf{e}_6 = \sqrt{3}\left\{t_p^{\star}(C_{ix}^3 - 3C_{ix}C_{iy}^2)\right\}, \quad \mathbf{e}_7 = \sqrt{3}\left\{t_p^{\star}(C_{iy}^3 - 3C_{iy}C_{ix}^2)\right\}, \tag{A.1}$$

$$\mathbf{e}_8 = \frac{\mathbf{E}^{\mathrm{im}}}{\|\mathbf{E}^{\mathrm{im}}\|}, \quad \mathbf{E}^{\mathrm{im}} = \left\{t_p^{\star}\frac{c_i^2}{D} - r_p^{\star}\right\}, \quad \|\mathbf{E}^{\mathrm{im}}\| = \frac{1}{6}\sqrt{(41 - 138c_s^2 + 117c_s^4)},$$

$$\mathbf{e}_9 = \frac{\mathbf{T}_{\mathrm{p}}}{\|\mathbf{T}_{\mathrm{p}}\|}, \quad T_0 = 4(1 - 3c_s^2), \quad T_1 = 13 - 21c_s^2, \quad T_2 = 24c_s^2 - 14,$$

$$\|\mathbf{T}_{\mathrm{p}}\| = 6\sqrt{(41 - 138c_s^2 + 117c_s^4)}, \tag{A.2}$$

where vector $\mathbf{T}_{\mathrm{p}}$ has constant value T_p for each p-class. Basis vectors (A.2) are similar to those in [25,26,38,50], except the two last vectors. Let us refer here to two alternative, c_s^2-independent, basis vectors as $\mathbf{E}$ and $\mathbf{H}$:

$$E_i = \{3C_i^2 - 4\}, \quad H_i = \{9C_{ix}^2 C_{iy}^2 - 6C_i^2 + 4\},$$

$$E_i = -\frac{2}{\|\mathbf{E}^{\mathrm{im}}\|}(-2 + 3c_s^2)\mathbf{e}_8 + \frac{36}{\|\mathbf{T}_{\mathrm{p}}\|}(-5 + 9c_s^2)\mathbf{e}_9, \tag{A.3}$$

$$H_i = \frac{1}{\|\mathbf{E}^{\mathrm{im}}\|}(-5 + 9c_s^2)\mathbf{e}_8 + \frac{72}{\|\mathbf{T}_{\mathrm{p}}\|}(-2 + 3c_s^2)\mathbf{e}_9.$$

The eigenvalues associated with basis vectors (A.2) are

$$\{0, 0, 0, \lambda_\psi^{\alpha\beta}, \lambda_\psi^{\alpha\alpha}, \lambda_2, \lambda_2, \lambda_e, \lambda_l\}. \tag{A.4}$$

When $\lambda_e = \lambda_l$, the system of basis vectors (A.2) is equivalent to those which use the vectors (A.2) with the corresponding equal eigenvalues. When $\lambda_\psi^{\alpha\beta} = \lambda_\psi^{\alpha\alpha}$, the projection of first-order population expansion (10), (12) in standard coordinate system on the basis (A.2) yields the coefficients of the decomposition as (see first terms for $(N_i^{\mathrm{eq.}}, \mathbf{e}_k)$ and last term for $(N_i^{(1)}, \mathbf{e}_k)$, respectively):

$$(\mathbf{N}, \mathbf{e}_1) = \frac{1}{3}\rho, \quad (\mathbf{N}, \mathbf{e}_2) = \frac{J_x}{\sqrt{6}}, \quad (\mathbf{N}, \mathbf{e}_3) = \frac{J_y}{\sqrt{6}},$$

$$(\mathbf{N}, \mathbf{e}_4) = \frac{j_x j_y}{2\rho} + \frac{1}{6}\frac{1}{\lambda_\psi}\left(\frac{\partial j_x}{\partial y} + \frac{\partial j_y}{\partial x}\right),$$

$$(\mathbf{N}, \mathbf{e}_5) = \frac{1}{2\rho}\left(j_x^2 - j_y^2\right) + \frac{1}{3\lambda_\psi}\left(\frac{\partial j_x}{\partial x} - \frac{\partial j_y}{\partial y}\right),$$

$$(\mathbf{N}, \mathbf{e}_6) = \frac{\sqrt{3}J_x}{6}, \quad (\mathbf{N}, \mathbf{e}_7) = \frac{\sqrt{3}J_y}{6}, \tag{A.5}$$

$$(\mathbf{N}, \mathbf{e}_8) = \alpha_8\rho + \beta_8\frac{(j_x^2 + j_y^2)}{\rho} + \frac{1}{\|\mathbf{E}^{\mathrm{im}}\|}\frac{1}{\lambda_e}\left(\frac{\partial j_x}{\partial x} + \frac{\partial j_y}{\partial y}\right),$$

$$\alpha_8 = -\frac{1}{12\|\mathbf{E}^{\mathrm{im}}\|}(39c_s^4 - 43c_s^2 + 12), \quad \beta_8 = -\frac{1}{12\|\mathbf{E}^{\mathrm{im}}\|}(15c_s^2 - 9),$$

$$(\mathbf{N}, \mathbf{e}_9) = \alpha_9\rho + \beta_9\frac{(j_x^2 + j_y^2)}{\rho},$$

$$\alpha_9 = \frac{1}{\|\mathbf{T}_{\mathrm{p}}\|}(4 - 6c_s^2), \beta_9 = \frac{3}{\|\mathbf{T}_{\mathrm{p}}\|}(3c_s^2 - 1). \tag{A.6}$$

When $c_s^2 = 1/3$, the data are: $\|\mathbf{E}^{\mathrm{im}}\| = \sqrt{2}/3$, $\|\mathbf{T}_{\mathrm{p}}\| = 12\sqrt{2}$, $E_i = 3\sqrt{2}(\mathbf{e}_8 - \mathbf{e}_9)$, $H_i = -3\sqrt{2}(\mathbf{e}_8 + \mathbf{e}_9)$, $\alpha_8 = -\sqrt{2}/4$, $\beta_8 = \sqrt{2}/2$, $\alpha_9 = \sqrt{2}/12$, $\beta_9 = 0$.

A.2. D3Q15 model

Let 15 velocities of the $D3Q15$ model be ordered as following: $(0,0,0)$, $(1,0,0)$, $(0,1,0)$, $(-1,0,0)$, $(0,-1,0)$, $(0,0,1)$, $(0,0,-1)$, $(1,1,1)$, $(-1,1,1)$, $(-1,-1,1)$, $(1,-1,1)$, $(1,1,-1)$, $(-1,1,-1)$, $(-1,-1,-1)$, $(1,-1,-1)$. Orthonormal vectors are chosen in the form (cf. relation (12)), written in standard coordinate system

$$\mathbf{e}_1 = \frac{1}{\sqrt{15}}\{C_i^0\}, \quad \mathbf{e}_2 = \frac{1}{\sqrt{10}}\{C_{ix}\}, \quad \mathbf{e}_3 = \frac{1}{\sqrt{10}}\{C_{iy}\}, \quad \mathbf{e}_4 = \frac{1}{\sqrt{10}}\{C_{iz}\},$$

$$\mathbf{e}_5 = 6\sqrt{2}\{t_p^\star C_{ix}C_{iy}\}, \quad \mathbf{e}_6 = 6\sqrt{2}\{t_p^\star C_{iy}C_{iz}\}, \quad \mathbf{e}_7 = 6\sqrt{2}\{t_p^\star C_{ix}C_{iz}\},$$

$$\mathbf{e}_8 = \frac{9}{2\sqrt{3}}\left\{t_p^\star\left(C_{ix}^2 - \frac{1}{D}c_i^2\right)\right\}, \quad \mathbf{e}_9 = \frac{3}{2}\{t_p^\star(C_{iy}^2 - C_{iz}^2)\},$$

$$\mathbf{e}_{10} = \frac{1}{\sqrt{8}}\{t_p^\star C_{ix}C_{iy}C_{iz}\}, \quad \mathbf{e}_{11} = \frac{3}{\sqrt{10}}\{t_p^\star(2C_{ix}^3 - 3C_{ix}(C_{iy}^2 + C_{iz}^2))\},$$

$$\mathbf{e}_{12} = \frac{3}{\sqrt{10}}\{t_p^\star(2C_{iy}^3 - 3C_{iy}(C_{ix}^2 + C_{iz}^2))\}, \tag{A.7}$$

$$\mathbf{e}_{13} = \frac{3}{\sqrt{10}}\{t_p^\star(2C_{iz}^3 - 3C_{iz}(C_{ix}^2 + C_{iy}^2))\},$$

$$\mathbf{e}_{14} = \frac{\mathbf{E}^{\mathrm{im}}}{\|\mathbf{E}^{\mathrm{im}}\|}, \quad \mathbf{E}^{\mathrm{im}} = \left\{t_p^\star\frac{c_i^2}{D} - r_p^\star\right\}, \quad \|\mathbf{E}^{\mathrm{im}}\| = \frac{1}{36}\sqrt{(1410 - 6660c_s^2 + 7938c_s^4)},$$

$$\mathbf{e}_{15} = \frac{\mathbf{T}_{\mathrm{p}}}{\|\mathbf{T}_{\mathrm{p}}\|}, \quad T_0 = 2(5 - 21c_s^2), \quad T_1 = 25 - 57c_s^2, \quad T_2 = -20 + 48c_s^2,$$

$$\|\mathbf{T}_{\mathrm{p}}\| = \sqrt{30(235 - 1110c_s^2 + 1323c_s^4)}. \tag{A.8}$$

Here again, the constant multiple before the lattice vector corresponds to inverse of the norm of this vector. Basis vectors (A.6) are similar to those in [41], except the two last vectors. Let us refer here to two alternative, c_s^2-independent, basis vectors as $\mathbf{E}$ and $\mathbf{H}$:

$$E_i = \{C_i^2 - 2\}, \quad H_i = \frac{1}{2}\{15C_i^4 - 55C_i^2 + 32\},$$

$$E_i = \frac{1}{3\|\mathbf{E}^{im}\|}(-5 + 9c_s^2)\mathbf{e}_{14} - \frac{30}{\|\mathbf{T}_p\|}(-11 + 27c_s^2)\mathbf{e}_{15},$$

$$H_i = \frac{5}{3\|\mathbf{E}^{im}\|}(-11 + 27c_s^2)\mathbf{e}_{14} + \frac{120}{\|\mathbf{T}_p\|}(-5 + 9c_s^2)\mathbf{e}_{15}. \tag{A.9}$$

Basis vectors (A.6) are associated with the following eigenvalues:

$$\{0, 0, 0, 0, \lambda_\psi^{\alpha\beta}, \lambda_\psi^{\alpha\beta}, \lambda_\psi^{\alpha\beta}, \lambda_\psi^{\alpha\alpha}, \lambda_\psi^{\alpha\alpha}, \lambda_{xyz}, \lambda_2, \lambda_2, \lambda_2, \lambda_e, \lambda_l\}. \tag{A.10}$$

Same remarks as for (A.2) are valid here. In case $\lambda_\psi^{\alpha\beta} = \lambda_\psi^{\alpha\alpha}$, the coefficients of the decomposition on the basis (A.6) are related with the macroscopic quantities as

$$(\mathbf{N}, \mathbf{e}_1) = \frac{\rho}{3\sqrt{5}}, \quad (\mathbf{N}, \mathbf{e}_2) = \frac{J_x}{\sqrt{10}}, \quad (\mathbf{N}, \mathbf{e}_3) = \frac{J_y}{\sqrt{10}}, \quad (\mathbf{N}, \mathbf{e}_4) = \frac{J_z}{\sqrt{10}},$$

$$(\mathbf{N}, \mathbf{e}_5) = \frac{j_x j_y}{2\sqrt{2}\rho} + \frac{1}{6\sqrt{2}}\frac{1}{\lambda_\psi}\left(\frac{\partial j_x}{\partial y} + \frac{\partial j_y}{\partial x}\right),$$

$$(\mathbf{N}, \mathbf{e}_6) = \frac{j_y j_z}{2\sqrt{2}\rho} + \frac{1}{6\sqrt{2}}\frac{1}{\lambda_\psi}\left(\frac{\partial j_y}{\partial z} + \frac{\partial j_z}{\partial y}\right),$$

$$(\mathbf{N}, \mathbf{e}_7) = \frac{j_x j_z}{2\sqrt{2}\rho} + \frac{1}{6\sqrt{2}}\frac{1}{\lambda_\psi}\left(\frac{\partial j_x}{\partial z} + \frac{\partial j_z}{\partial x}\right),$$

$$(\mathbf{N}, \mathbf{e}_8) = \frac{(2j_x^2 - (j_y^2 + j_z^2))}{2\sqrt{3}\rho} + \frac{\sqrt{3}}{9\lambda_\psi}\left(2\frac{\partial j_x}{\partial x} - \left(\frac{\partial j_y}{\partial y} + \frac{\partial j_z}{\partial z}\right)\right), \tag{A.11}$$

$$(\mathbf{N}, \mathbf{e}_9) = \frac{(j_y^2 - j_z^2)}{2\rho} + \frac{1}{3}\frac{1}{\lambda_\psi}\left(\frac{\partial j_y}{\partial y} - \frac{\partial j_z}{\partial z}\right),$$

$$(\mathbf{N}, \mathbf{e}_{10}) = 0, \quad (\mathbf{N}, \mathbf{e}_{11}) = \frac{7J_x}{6\sqrt{10}}, \quad (\mathbf{N}, \mathbf{e}_{12}) = \frac{7J_y}{6\sqrt{10}}, \quad (\mathbf{N}, \mathbf{e}_{13}) = \frac{7J_z}{6\sqrt{10}},$$

$$(\mathbf{N}, \mathbf{e}_{14}) = \alpha_{14}\rho + \beta_{14}\frac{(j_x^2 + j_y^2 + j_z^2)}{\rho} + \frac{1}{\|\mathbf{E}^{im}\|}\frac{1}{\lambda_e}\left(\frac{\partial j_x}{\partial x} + \frac{\partial j_y}{\partial y} + \frac{\partial j_z}{\partial z}\right),$$

$$\alpha_{14} = -\frac{1}{72\|\mathbf{E}^{im}\|}(441c_s^4 + 72 - 353c_s^2), \quad \beta_{14} = -\frac{1}{72\|\mathbf{E}^{im}\|}(57c_s^2 - 25),$$

$$(\mathbf{N}, \mathbf{e}_{15}) = \alpha_{15}\rho + \beta_{15}\frac{(j_x^2 + j_y^2 + j_z^2)}{\rho},$$

$$\alpha_{15} = \frac{2}{\|\mathbf{T}_p\|}(5 - 11c_s^2), \quad \beta_{15} = \frac{10}{\|\mathbf{T}_p\|}(3c_s^2 - 1). \tag{A.12}$$

Similar as above, vector $\mathbf{N} - \mathbf{N}^{eq.}$ has no projection on first four vectors, corresponding to mass and momentum and the corresponding terms can be omitted in summation in Eq. (5). When $c_s^2 = 1/3$, the data are: $\|\mathbf{E}^{im}\| = \sqrt{2}/6$, $\|\mathbf{T}_p\| = 6\sqrt{10}$, $E_i = -2\sqrt{2}\mathbf{e}_{14} + \sqrt{10}\mathbf{e}_{15}$, $H_i = -10\sqrt{2}\mathbf{e}_{14} - 4\sqrt{10}\mathbf{e}_{15}$, $\alpha_{14} = -5\sqrt{2}/36$, $\beta_{14} = \sqrt{2}/4$, $\alpha_{15} = 2\sqrt{10}/45$, $\beta_{15} = 0$.

A.3. Remarks on choice of the free eigenvalues

The basis vectors above are mainly chosen among the polynomial vectors coming into Chapman–Enskog expansion. In case of more general model considered in [25,26,38], the eigenvalues $\lambda_\psi^{\alpha\beta}$ and $\lambda_\psi^{\alpha\alpha}$ can differ under condition that the equilibrium function is modified in order to recover the correct stress-tensor term in the derived Navier–Stokes equations. Similar generalization [21] is done for $D3Q15$ model. In this paper, we assume $\lambda_\psi^{\alpha\beta}$ and $\lambda_\psi^{\alpha\alpha}$ to be equal and denote them as λ_ψ. Eigenvalues λ_e and λ_ψ enter as the coefficients into first-order expansion (cf. (12)) and, therefore, determine the transport coefficients (15a) and (15b). Other eigenvalues are free.

Magic solution for free eigenvalues relates the eigenvalues associated with the odd order polynomial eigenvectors $(\lambda_\mathscr{O} = \{\lambda_2, \lambda_{xyz}\})$ to those associated with the even order polynomials $(\lambda_\mathscr{E} = \{\lambda_\psi, \lambda_e, \lambda_l\})$ through "magic" condition

$$\lambda_\mathscr{O}(\lambda_\mathscr{E}) = -8\frac{\lambda_\mathscr{E} + 2}{\lambda_\mathscr{E} + 8}. \tag{A.13}$$

Its properties are discussed in Section 4.1. When the nonlinear term is present at equilibrium (10), the solution (A.13) is not more exact for Poiseuille flow until "free" projection $\alpha\mathbf{H}$ is introduced into the equilibrium function

$$\mathbf{N}^{\text{eq.}} \to \mathbf{N}^{\text{eq.}} + \alpha\mathbf{H}. \tag{A.14}$$

Here, α is some constant and $\mathbf{H}$ is given by relation (A.3) for the $D2Q9$ model and by relation (A.9) for the $D3Q15$ model. Important here that equilibrium projection on $\mathbf{H}$ does not influence the derived Navier–Stokes equations. The coefficient α can be used to annihilate the contribution of a non-linear term in $\epsilon N_i^{(1)}$. In particular, when

$$\alpha = \frac{1}{12}(j_x^2 + j_y^2) \quad \text{for } D2Q9, \tag{A.15a}$$

$$\alpha = -\frac{1}{24}(j_x^2 + j_y^2 + j_z^2) \quad \text{for } D3Q15, \tag{A.15b}$$

such a term vanishes when j has only one non-zero component, e.g. Poiseuille flow. In terms of equilibrium weights, solution (A.15a) means that the projection of equilibrium in a form (10) is doubled in 2D: $(\mathbf{N}^{\text{eq.}} + \alpha\mathbf{H}, \mathbf{H}) = 2(\mathbf{N}^{\text{eq.}}, \mathbf{H})$. This coincides with the solution obtained with another approach by D.d'Humières [40]. In 3D, we have $(\mathbf{N}^{\text{eq.}} + \alpha\mathbf{H}, \mathbf{H}) = 4(\mathbf{N}^{\text{eq.}}, \mathbf{H})$. Numerical computations confirm that when $\alpha\mathbf{H}$ is added to equilibrium, momentum definition (3b) is used, and magic solution (A.13) is employed for free eigenvalues, Poiseuille profile in a channel of given width is obtained exactly, likely as in case of linear equilibrium function.

First-order solution for free eigenvalues puts all eigenvalues except λ_ψ equal to -1

$$\lambda_2 = \lambda_e = \lambda_{xyz} = \lambda_l = -1. \tag{A.16}$$

In this case, only the projection on second-order polynomial basis vectors associated with λ_ψ does not vanish after collision. This becomes especially transparent when the collision is written in the equivalent form:

$$\widetilde{N}_i(\mathbf{r}, t) = N_i^{\text{eq.}}(\mathbf{r}, t) + \sum_{k=0}^{b_m}(1 + \lambda_k)(\mathbf{N} - \mathbf{N}^{\text{eq.}}, \mathbf{e}_k)e_{ki}. \tag{A.17}$$

While using Chapman–Enskog expansion at reconstruction step, we neglect $O(\epsilon^2)$ and $O(M^2)$ terms associated with the eigenvalues λ_e and free eigenvalues. Assuming that first-order collision could dump the oscillations in these terms, we often use it for calculations in this paper.

A.4. Implementation of collision

Equation (5) is formulated in terms of the normalized basis vectors in order to simplify the notations. It is much more numerically efficient to represent basis vectors as the vectors with integer components, say $\mathbf{e}_k^{\text{int}}$. This enables us to compute easily all equal linear combinations which come into projection and into the decomposition: $\phi_k = 1/\|\mathbf{e}_k^{\text{int}}\|^2 \times (\mathbf{N}, \mathbf{e}_k^{\text{int}})$ and $\sum_k \phi_k \mathbf{e}_k^{\text{int}}$, accordingly. Moreover, computing the generalized collision (5) does not require the evaluation of equilibrium function in a form (10). Following idea [40], one can represent it in a form of equilibrium projection. The collision reads then

$$\tilde{\mathbf{N}}(\boldsymbol{r}, t) = \mathbf{N}(\boldsymbol{r}, t) + \sum_{k=0}^{b_m} \lambda_k \{\phi_k - \phi_k^{\text{eq.}}\} \mathbf{e}_k^{\text{int}}, \quad \phi_k^{\text{eq.}} = (\mathbf{N}^{\text{eq.}}, \mathbf{e}_k^{\text{int}}). \tag{A.18}$$

Since $\phi_k^{\text{eq.}}$ can be computed analytically (see (A.4) and (A.10)), the computational efforts reduce drastically (at least at factor two) and become quite comparable with the BGK collision where the equilibrium (10) should be computed. Nevertheless, when the equilibrium is computed for some other purpose as well, first-order collision is relatively fast. A particular fastest choice $\lambda_k \equiv -1$ is employed in [81]. In case of convergence to stationary state, however, a proper choice of the eigenvalues reduces drastically the number of time steps without loss of the accuracy.

A.5. First-order expansion

We precise here how we obtain first-order correction to equilibrium $\epsilon N_i^{(1)}$ in the form (12) in the standard coordinate system. Following [14], Chapman–Enskog expansion (9) $\partial_t = \epsilon \partial_{t_1} + \epsilon^2 \partial_{t_2}, \partial_x = \epsilon \partial_{\tilde{x}}$ leads to ϵ^1-accurate macroscopic relations:

$$\partial_{t_1} \rho + \nabla' \cdot j_\alpha = 0, \tag{A.19a}$$

$$\partial_{t_1} j_\alpha + \partial_{\tilde{\beta}} P_{\alpha\beta} = 0, \quad P_{\alpha\beta} = c_s^2 \rho \delta_{\alpha\beta} + \rho u_\alpha u_\beta. \tag{A.19b}$$

Correction $N_i^{(1)}$ satisfies first-order Taylor development of Eq. (5):

$$\partial_{t_1} N_i^{\text{eq.}} + C_{i\alpha} \partial_{\tilde{\alpha}} N_i^{\text{eq.}} = \sum_{j=0}^{b_m} A_{ij} N_j^{(1)}, \quad i \in \{0, \dots, b_m\}. \tag{A.20}$$

Substitution of the relation (10) into (A.20) yields with help of the relations (A.19a), (A.19b) and when $O(u^2)$ and $O(\epsilon^2)$ terms are neglected:

$$\partial_{t_1} N_i^{\text{eq.}} + C_{i\alpha} \partial_{\tilde{\alpha}} N_i^{\text{eq.}} = C_{i\alpha} t_p^\star \overbrace{[\partial_{t_1} j_\alpha + c_s^2 \partial_{\tilde{\alpha}} \rho]}^{=0} + r_p^\star \partial_{t_1} \rho + \frac{\partial j_\alpha}{\partial \beta'} t_p^\star C_{i\alpha} C_{i\beta}$$

$$= \nabla' \cdot \boldsymbol{j} \left(t_p^\star \frac{c_i^2}{D} - r_p^\star \right) + \frac{\partial j_\alpha}{\partial \beta'} t_p^\star \left(C_{i\alpha} C_{i\beta} - \frac{c_i^2}{D} \delta_{\alpha\beta} \right). \tag{A.21}$$

By using the definitions in (12), relation (A.20) becomes

$$\epsilon \sum_{j=0}^{b_m} A_{ij} N_j^{(1)} = \frac{\partial j_\alpha}{\partial \beta} Q_{i\alpha\beta} + \nabla \cdot \boldsymbol{j} E_i^{\text{im}}. \tag{A.22}$$

Replacing $\{\alpha, \beta\}$ by $\{\alpha', \beta'\}$, the derivation in any other coordinate system follows exactly the same lines. Relation (A.22) takes then more general form

$$\epsilon \sum_{j=0}^{b_m} \mathsf{A}_{ij} N_j^{(1)} = \frac{\partial j_\alpha}{\partial \beta'} Q_{i\alpha'\beta'} + \nabla \cdot \boldsymbol{j} E_i^{\mathrm{im}}. \tag{A.23}$$

Since the vectors $\{Q_{i\alpha'\beta'}\}$ are fully decomposed on the second-order polynomial basis vectors associated with the eigenvalue λ_ψ and vector $\mathbf{E}^{\mathrm{im}}$ is chosen to be a basis vector associated with the eigenvalue λ_e, relation (A.23) takes a form (12).

References

[1] C. Adler, D. d'Humières, D.H. Rothman, Surface tension and interface fluctuations in immiscible lattice gases, J. Phys. A France 4 (1994) 29–46.

[2] A.N. Alexandrou, E. Duc, V. Entov, Inertial, viscous and yield stress effects in Bingham fluid filling of a 2-D cavity, J. Non-Newtonian Fluid Mech. 96 (3) (2001) 383.

[3] M. Bakhudarov, H. You, J. Ortega, J. Beech, S.B. Chin, D.H. Kirkwood, Experimental validation and development of FLOW-3D for casting problems, in: Modeling of Casting Welding and Advanced Solidification Processes-VI, Proceedings of the VIth International Conference 1993, Palm Beach, FL, TMS, AIME, 1993, p. 441.

[4] A.N. Brooks, T.J.R. Hughes, Streamline upwind/Petrov–Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier–Stokes equations, Comp. Methods Appl. Mech. Eng. 32 (1982) 199.

[5] J.M. Buick, C.A. Greated, Gravity in a lattice Boltzmann model, Phys. Rev. E 61 (2000) 5307.

[6] C. Cercignani, Mathematical Methods in Kinetic Theory, Macmillan, New York, 1969.

[7] H. Chen, C. Teixeira, K. Molvig, Realization of fluid boundary conditions via discrete Boltzmann dynamics, Int. J. Mod. Phys. 9 (1998) 1281.

[8] S. Chen, Z. Wang, X. Shan, Gary D. Doolen, Lattice Boltzmann computational fluid dynamics in three dimensions, J. Stat. Phys. 68 (1992) 378.

[9] S. Chen, G.D. Doolen, Lattice Boltzmann method for fluid flows, J. Fluid Mech. 30 (1998) 329.

[10] I. Christie, D.F. Griffiths, A.R. Mitchell, O.C. Zienkiewicz, Finite element methods for second order differential equations with significant first derivatives, Int. J. Numer. Methods Eng. 10 (1976) 1389.

[11] High pressure die casting simulation using Smoothed Particle Hydrodynamics, J. Cast Metals Res. 12 (2000) 335.

[12] R. Cornubert, D. d'Humières, D. Levermore, A Knudsen layer theory, Physica D 47 (1991) 241.

[13] O. Filippova, D. Hänel, Boundary-fitting and local grid refinement for lattice-BGK models, Int. J. Mod. Phys. 9 (1998) 1271.

[14] U. Frisch, D. d'Humiéres, B. Hasslacher, P. Lallemand, Y. Pomeau, J.-P. Rivet, Lattice gas hydrodynamics in two and three dimensions, Complex Systems 1 (1987) 649.

[15] D.E. Fyfe, E.S. Oran, M.J. Fritts, Surface tension and viscosity with Lagrangian hydrodynamics on a triangular mesh, J. Comput. Phys. 76 (1988) 394.

[16] O.S. Galaktionov, P.D. Anderson, G.W.M. Peters, F.N. Van de Vosse, An adaptive front tracking technique for three-dimensional transient flows, Int. J. Numer. Methods Fluids 32 (2000) 201.

[17] D.M. Gao, A three-dimensional hybrid finite element-volume tracking model for mold filling in casting processes, Int. J. Numer. Methods Fluids 29 (1999) 877.

[18] I. Ginzbourg, P.M. Adler, Boundary flow condition analysis for three-dimensional lattice Boltzmann model, J. Phys. II France 4 (1994) 191.

[19] I. Ginzbourg, P.M. Adler, Boundary conditions at a plane liquid–liquid interface in FCHC lattice Boltzmann model (unpublished, 1994);
I. Ginzbourg, P.M. Adler, Surface tension models with different viscosities, Transport Porous Med. 20 (1995) 37.

[20] I. Ginzbourg, D. d'Humières, Local second-order boundary method for lattice Boltzmann models, J. Stat. Phys. 84 (5/6) (1996) 927;
I. Ginzbourg, D. d'Humières, Local second-order boundary method for Lattice Boltzmann models. Part II. Application to complex geometries (unpublished, 1996).

[21] I. Ginzburg, General second order Chapman–Enskog expansion for Lattice Boltzmann models (unpublished, 1999).

[22] I. Ginzburg, G. Wittum, Two-Phase flows on interface refined grids modeled with VOF, staggered finite volumes, and spline interpolants, J. Comput. Phys. 166 (2001) 302.

[23] I. Ginzburg, Introduction of upwind and free boundary into lattice Boltzmann method, in: T. Sonar, I. Thomas (Eds.), Discrete Modelling and Discrete Algorithms in Continuum Mechanics, Logos-Verlag, Berlin, 2001, pp. 97–109.

[24] I. Ginzburg, K. Steiner, A free surface lattice-Boltzmann method for modelling the filling of expanding cavities by Bingham Fluids, Philos. Trans. R. Soc. Lond. A 360 (2002) 453.
[25] L. Giraud, D. d'Humières, P. Lallemand, A lattice Boltzmann model for Jeffreys viscoelastic fluid, Europhys. Lett. 42 (1998) 625.
[26] L. Giraud, Fluides visco-élastiques par la méthode de Boltzmann sur réseau, PhD, Université Paris VI, 1997.
[27] D. Wolf-Gladrow, Lattice Gas Cellular Automata and Lattice Boltzmann Models: An Introduction, Lecture Notes in Mathematics, vol. 1725, Springer, Berlin, 2000, ISBN 3-540-66973-6.
[28] A.K. Gunstensen, D.H. Rothmann, S. Zaleski, G. Zanetti, Lattice Boltzmann model of immiscible fluids, Phys. Rev. A 43 (8) (1991) 4320;
A.K. Gunstensen, D.H. Rothmann, Europhys. Lett. 18 (1992) 157;
, J. Geophys. Res. 98 (1993) 6431.
[29] Injection Molding Handbook: The Complete Molding Operation: Technology, Performance, Economics, Van Nostrand Reinhold, New York, 1986.
[30] Xiaoyi He, Yisu Zou, Li-Shi Luo, Micah Dembo, Analytic solutions of simple flows and analysis of nonslip boundary conditions for the lattice Boltzmann BGK model, J. Stat. Phys. 87 (1997) 913.
[31] X. He, Li-Shi Luo, A priori derivation of the lattice Boltzmann equation, Phys. Rev. E 55 (1997) R6333.
[32] X. He, S. Chen, Raoyang Zhang, A lattice Boltzmann scheme for incompressible multiphase flow and its application in simulation of Rayleigh–Taylor instability, J. Comput. Phys. 152 (1999) 642.
[33] F.J. Higuera, J. Jimenez, Boltzmann approach to lattice gas simulations, Europhys. Lett. 9 (1989) 663.
[34] C.W. Hirt, B.D. Nicholls, Volume of fluid (VOF) method for the dynamics of free boundaries, J. Comput. Phys. 39 (1981) 201.
[35] C.W. Hirt, A Flow-3D study of the importance of fluid momentum in mold filling, Num. Sim. of Casting Solidification in Automotive Applications. The Minerals Metals and Material, Society (1991).
[36] C.W. Hirt, R.P. Harper, Flow modeling for casting analysis, Technical Note FSI-88-TN7, Flow Science Int., Los Alamos, NM, 1988.
[37] S. Hou, J. Sterling, S. Chen, G.D. Doolen, A lattice Boltzmann subgrid model for high Reynolds number flows, Fields Institute Commun. 6 (1996) 151.
[38] D. d'Humières, Generalized lattice-Boltzmann equations, AIAA rarefied gas dynamics: theory and simulations, Prog. Astronaut. Aeronaut. 59 (1992) 450.
[39] I. Ginzburg, D. d'Humières, Multi-reflection boundary conditions for lattice Boltzmann models, Institut für Techno- und Wirtschaftsmathematik, Berichte des Fraunhofer ITWM, Nr. 38, Kaiserslautern, www.itwm.fhg.de, 2002.
[40] D.d'Humières, private communications.
[41] D. d'Humières, I. Ginzburg, M. Krafczyk, P. Lallemand, L.-S. Luo, Multiple-relaxation-time lattice Boltzmann models in three dimensions, Philos. Trans. R. Soc. Lond. A 360 (2002) 437.
[42] J.H. Jeong, D.Y. Yang, Finite element analysis of transient fluid flow with free surface using VOF (VOLUME-OF-FLUID) method and adaptive grid, Int. J. Numer. Methods Fluids 26 (1998) 1127.
[43] I.V. Karlin, A. Ferrante, H.C. Öttinger, Perfect entropy functions of the lattice Boltzmann method, Europhys. Lett. 47 (2) (1999) 182.
[44] J. Kuhnert, S. Tiwari, private communications.
[45] P. Klein, Scalable parallel software by chromatic ordered communications patterns, Institut für Techno- und Wirtschaftsmathematik, Annual Report, Kaiserslautern, 1999.
[46] D. Kehrwald, Numerical analysis of immiscible lattice BGK, PhD Thesis, Kaiserslautern, 2002.
[47] D.B. Kothe, Perspective on Eulerian finite volume methods for incompressible interfacial flows. Available from http://lune.mst.lanl.gov/Telluride/Text/publications.html.
[48] D. Kothe, D. Juric, K. Lam, B. Lally, Numerical receipes for mold filling simulation, The Minerals, Metals and Materials Society, 17 (1998). [In: Modeling of Casting Welding and Advanced Solidification Processes-VIII, edited by B.G. Thomas, C. Beckermann].
[49] A.J.C. Ladd, Numerical simulations of particulate suspensions via a discretized Boltzmann equation. Part 2. Numerical results, J. Fluid Mech. 271 (1994) 311.
[50] P. Lallemand, Li-Shi Luo, Theory of the lattice Boltzmann method: dispersion, dissipation, isotropy, Galilean invariance, and stability, Phys. Rev. E 61 (2000) 6546.
[51] J. Li, Calcul d'Interface Affine par Morceaux, C. R. Acad. Sci. Paris IIb 320 (1995) 391.
[52] M.D. Lipinski, Mold filling simulation for casting processes, Ph.D. Thesis, Aachen, 1996.
[53] N. Lock, M. Jaeger, M. Medale, R. Occelli, Local mesh adaptation technique for front tracking problems, Int. J. Numer. Methods Fluids 28 (1998) 719.
[54] Li-Shi Luo, Theory of the lattice Boltzmann method: lattice Boltzmann models for non-ideal gases, Phys. Rev. E 62 (2000) 4982.
[55] F. Mashayek, N. Ashgriz, A hybrid finite-element-volume-of-fluid method for simulating free surface flows and interfaces, Int. J. Numer. Methods Fluids 20 (1995) 1363.
[56] J.C. Maxwell, Philos. Trans. R. Soc. I, Appendix(1879), in Scientific Papers of J.C. Maxwell, Dover Publications, 1965.

[57] J.J. Monaghan, Simulating free surface flows with SPH, J. Comput. Phys. 110 (1994) 399.
[58] Renwei Mei, Li-Shi Luo, Wei Shyy, An accurate curved boundary treatment in the lattice Boltzmann method, J. Comput. Phys. 155 (1999) 307.
[59] S.J. Mosso, B.K. Swartz, D.B. Kothe, S.P. Clancy, Recent enhancements of volume tracking algorithm for irregular grids (Los Alamos National Laboratory, Los Alamos, NM, LA_UR_96_277, 1996), presented at the Parallel CFD Conference, Capri, Italy, March 20–23, 1996.
[60] S.J. Mosso, B.K. Swartz, D.B. Kothe, R.C. Ferrell, A parallel, volume-tracking algorithm for unstructured meshes, Los Alamos National Laboratory, Los Alamos, NM, LA_UR_96_2420, 1996.
[61] X. Nie, Y. Qian, G.D. Doolen, S. Chen, Lattice Boltzmann simulation of the two-dimensional Rayleigh–Taylor instability, Phys. Rev. E 58 (5) (1998) 6861–6864.
[62] J.E. Pilliod Jr., E.G. Puckett, Second order accurate volume-of-fluid algorithms for tracking material interfaces, Technical Report, Lawrence Berkeley National Laboratory, No. LBNL-40744.
[63] E.G. Puckett, A.S. Almgren, J.B. Bell, D.L. Marcus, W.J. Rider, A high-order projection method for tracking fluid interfaces in variable density incompressible flows, J. Comput. Phys. 79 (1988) 12.
[64] G.R. Price, G.T. Reader, R.D. Rowe, J.D. Bugg, A Piecewise Parabolic Interface Calculations for volume Tracking, Proceedings of the Sixth Annual Conference of the Computational Fluid/Dynamics Society of Canada, University of Victoria, Victoria, British/Columbia, 1998.
[65] W.H. Press, S.A. Teukolsky, W.T. Wetterling, B.P. Flannnery, Numerical Recipes in C., Cambridge University Press, Cambridge, 1992.
[66] Y.H. Qian, D. d'Humières, P. Lallemand, Lattice BGK models for Navier–Stokes equation, Europhys. Lett. 17 (1992) 479.
[67] W.J. Rider, D.B. Kothe, Reconstructing volume tracking, J. Comput. Phys. 141 (1998) 112.
[68] M. Rudman, Volume tracking methods for interfacial flow calculations, Int. J. Numer. Methods Fluids 24 (1997) 671.
[69] M. Rudman, Volume tracking methods for incompressible multifluid flows with large density variations, Int. J. Numer. Methods Fluids 28 (1998) 357.
[70] D.H. Rothman, S. Zaleski, Lattice-gas models of phase separation: interfaces, phase transitions, Rev. Mod. Phys. 66 (1994) 1417.
[71] D.H. Rothmann, S. Zaleski, Lattice Gas Dynamics Automata – Simple Model for Complex Hydrodynamics, Cambridge University Press, Cambridge, 1997, ISBN: 0-521-55201-X.
[72] B. Sirrel, M. Holiday, J. Campbell, The Benchmark Test 1995, in: Modeling of Casting Welding and Advanced Solidification Processes-VII, Proceedings of the VIIth International Conference, TMS, AIME, 1995, p. 915.
[73] T. Sato, S.M. Richardson, Numerical simulation method for viscoelastic flow with free surfaces-fringe element generation method, Int. J. Numer. Methods Fluids 26 (1994) 555.
[74] R. Scardovelli, S. Zaleski, Direct numerical simulation of free-surface and interfacial flow, Annu. Rev. Fluid Mech. 31 (1999) 567.
[75] M. Schelkle, M. Rieber, A. Frohn, Comparison of lattice Boltzmann and Navier–Stokes simulations of three-dimensional free surface flows, FED-VOL. 236, Proceedings of the ASME, Book No. H01072-1996.
[76] J.D. Sterling, S. Chen, Stability analysis of Lattice Boltzmann methods, J. Comput. Phys. 123 (1996) 196.
[77] S. Succi, R. Benzi, F. Higuera, The lattice Boltzmann equation: a new tool for computational fluid dynamics, Physica D 47 (1991) 219.
[78] M. Sussmann, P. Smereka, Axisymmetric free boundary problems, J. Fluid Mech. 341 (1997) 269.
[79] B. Swartz, The second order sharpening of blurred smooth borders, Math. Comput. 52 (1989) 675.
[80] S.H. Unverdi, G. Tryggvason, A front-tracking method for viscous, incompressible, multi-fluid flows, J. Comput. Phys. 100 (1992) 25.
[81] R. Verberg, A.J.C. Ladd, Simulation of low-Reynolds-number flow via a time-independent lattice-Boltzmann method, Phys. Rev. E 60 (1999) 3366.
[82] R.A. Worthing, J. Mozer, G. Seeley, Stability of lattice Boltzmann methods in hydrodynamic regimes, Phys. Rev. E 56 (1997) 2243.
[83] D.L. Youngs, Time-dependent multi-material flow with large fluid distortion, in: K.W. Morton, M.L. Norman (Eds.), Numerical Methods for Fluid Dynamics, 1986.