Overview of the lattice Boltzmann method for nano- and microscale fluid dynamics in materials science and engineering

2004 Modelling Simul. Mater. Sci. Eng. 12 R13

(http://iopscience.iop.org/0965-0393/12/6/R01)

INSTITUTE OF PHYSICS PUBLISHING MODELLING AND SIMULATION IN MATERIALS SCIENCE AND ENGINEERING

Modelling Simul. Mater. Sci. Eng. **12** (2004) R13–R46 PII: S0965-0393(04)77661-5

TOPICAL REVIEW

Overview of the lattice Boltzmann method for nano- and microscale fluid dynamics in materials science and engineering

D Raabe

Max-Planck-Institut für Eisenforschung, Max-Planck-Strasse 1, 40237 Düsseldorf, Germany

E-mail: raabe@mpie.de

Received 15 March 2004, in final form 2 August 2004
Published 16 September 2004
Online at stacks.iop.org/MSMSE/12/R13
doi:10.1088/0965-0393/12/6/R01

Abstract

The article gives an overview of the lattice Boltzmann method as a powerful technique for the simulation of single and multi-phase flows in complex geometries. Owing to its excellent numerical stability and constitutive versatility it can play an essential role as a simulation tool for understanding advanced materials and processes. Unlike conventional Navier–Stokes solvers, lattice Boltzmann methods consider flows to be composed of a collection of pseudo-particles that are represented by a velocity distribution function. These fluid portions reside and interact on the nodes of a grid. System dynamics and complexity emerge by the repeated application of local rules for the motion, collision and redistribution of these coarse-grained droplets. The lattice Boltzmann method, therefore, is an ideal approach for mesoscale and scale-bridging simulations. It is capable to tackling particularly those problems which are ubiquitous characteristics of flows in the world of materials science and engineering, namely, flows under complicated geometrical boundary conditions, multi-scale flow phenomena, phase transformation in flows, complex solid–liquid interfaces, surface reactions in fluids, liquid–solid flows of colloidal suspensions and turbulence. Since the basic structure of the method is that of a synchronous automaton it is also an ideal platform for realizing combinations with related simulation techniques such as cellular automata or Potts models for crystal growth in a fluid or gas environment. This overview consists of two parts. The first one reviews the philosophy and the formal concepts behind the lattice Boltzmann approach and presents also related pseudo-particle approaches. The second one gives concrete examples in the area of computational materials science and process engineering, such as the prediction of lubrication dynamics in metal forming, dendritic crystal growth under the influence of fluid convection, simulation of metal foam processing, flow percolation in confined geometries, liquid crystal hydrodynamics and processing of polymer blends.

1. Introduction to the lattice gas and lattice Boltzmann simulation methods

1.1. Motivation for the use of discrete methods in computational fluid mechanics

The theoretical picture of fluid dynamics in the materials engineering community largely departs from the work of Navier and Stokes from the first half of the 19th century. Their differential formulation of the mechanics of incompressible flows, the so-called Navier–Stokes equation, accounts for the conservation of mass, momentum and energy, and the requirement that these quantities be conserved locally [1–3]. Tackling hydrodynamics and related problems with this equation amounts to solving coupled sets of nonlinear partial differential field equations by use of finite difference or finite element methods.

Although the Navier–Stokes framework serves as a long-established basis for predicting fluid behaviour, it has still not been possible to resolve some basic questions in the fields of modern materials science and engineering with it. This is due to the fact that the Navier–Stokes differential formulations theoretically do not apply and numerically also do not converge under conditions which are characterized by large Knudsen numbers (mean free molecule path divided by characteristic system length). Such restrictions occur when the mean free path of the fluid molecules is similar to the geometrical system constraints, such as, for instance the obstacle spacing or the roughness wavelength which may characterize mesoscopic system heterogeneity.

Prominent examples where such limitations occur are the simulation of nano- and microflows in filters, foams, micro-reactors or otherwise confined geometries; multi-component flows in the area of polymer and metal processing; tribology and wear in the area of lubricated contact mechanics and metal forming; liquid crystal processing; nanoscale process technology; lubrication in miniaturized components, liquid phase separation; joint fluid–gas flows; abrasion and sedimentation; fluid percolation in cellular structures; processing of metallic foams; as well as corrosion and solidification in non-quiescent environments to name but a few. These examples do not only challenge our basic understanding of fluid mechanics but represent at the same time key issues in modern materials science and engineering of considerable practical relevance.

Lattice gas cellular automata [4, 5] and their more mature (non-Boolean) successors, the lattice Boltzmann automaton techniques (see details in the ensuing sections), seem to be predestined to tackle some of these challenges in the domain of materials-related computational fluid dynamics in a more efficient way than the conventional Navier–Stokes approach.

The lattice Boltzmann technique belongs to a broader group of pseudo-particle methods which form a growing class of multi-scale simulation approaches to computational fluid mechanics, table 1, figure 1. Other important particle-based approaches in this class are (besides lattice gas cellular automata) the dissipative particle dynamics method [6–11] and the direct simulation Monte Carlo method [11–16] together with its hybrid mesh refinement variations [17].

These pseudo-particle approaches can be grouped into lattice-based cellular automaton approaches (lattice gas method, lattice Boltzmann method) and off-lattice approaches (dissipative particle dynamics method, direct simulation Monte Carlo method). While the reminder of this overview deals exclusively with vectorial cellular automaton models of fluid flow, the ensuing section provides a concise summary of the off-lattice methods.

1.2. Off-lattice pseudo-particle methods in computational fluid mechanics

The most important off-lattice pseudo-particle approach to computational fluid mechanics is the dissipative particle dynamics method [6–11], table 1, figure 1. This technique uses

Table 1. Overview of models in computational fluid mechanics. All approaches beyond the atomic-scale (molecular dynamics) and below the conventional continuum scale (Navier–Stokes solvers) use coarse-grained pseudo-particles which can either move on a fixed lattice (lattice-based pseudo-particle models) or continuously in space (off-lattice pseudo-particle models).

Models in computational fluid mechanics	
Molecular dynamics	
Pseudo-particle models	
Off-lattice models	Dissipative particle dynamics
	Direct simulation Monte Carlo methods
Lattice-based models	Lattice gas automata
	Lattice Boltzmann automata
Navier–Stokes solvers	

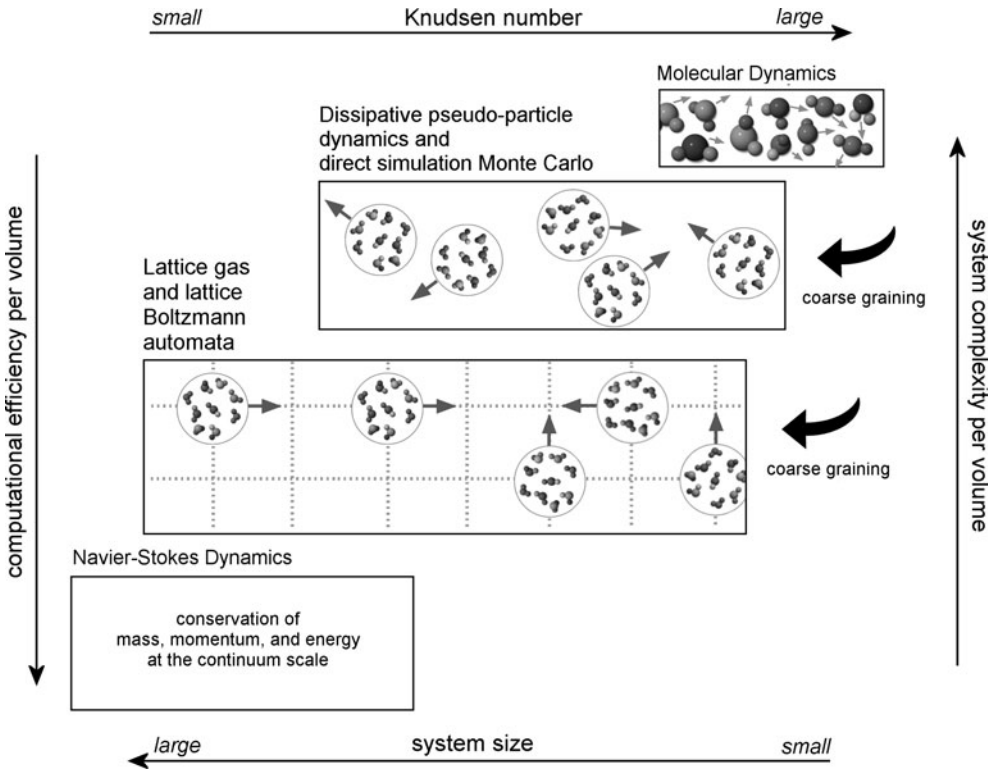

Figure 1. Various approaches to computational fluid dynamics together with their preferred range of applicability. Molecular dynamics methods integrate Newton's equations of motion for a set of molecules on the basis of an intermolecular potential. Dissipative particle dynamics and direct simulation Monte Carlo are off-lattice pseudo-particle methods in conjunction with Newtonian dynamics. Lattice gas and lattice Boltzmann methods treat flows in terms of coarse-grained fictive particles which reside on a mesh and conduct translation as well as collision steps entailing overall fluid-like behaviour. Navier–Stokes approaches solve continuum-based partial differential equations which account for the local conservation of mass, momentum and energy. These three methods have their respective strengths at different Knudsen numbers, where the Knudsen number is the ratio between the mean free molecule path and a characteristic length scale representing mesoscopic system heterogeneity (e.g. the obstacle size).

discrete fluid portions which can freely move in continuous space at discrete time increments. The method can be derived from molecular dynamics by means of coarse-graining, i.e. the pseudo-particles do not represent single atoms or molecules but rather mesoscopic droplets or clusters of atoms which carry the position and momentum of coarse-grained fluid elements. The philosophy of using such averaged particles instead of real molecules leads to a substantial gain in computational efficiency compared with conventional molecular dynamics methods, however, at the expense of a loss in microscopic detail.

The pseudo-particles interact pairwise according to a set of short-range interparticle central forces that include a repulsive conservative force a dissipative force and a random force acting symmetrically between each pair of pseudo-particles. The dissipative force acts to slow the particles down and to remove energy from them. The random force acts between all pairs of particles and is uncorrelated between different pairs. It adds energy to the system on average. Together with the dissipative force it acts as a thermostat for the system. The conservative force is derived from a pseudo potential energy similar to that in molecular dynamics.

As in conventional molecular dynamics methods the dynamical behaviour is realized by the integration of the Newtonian equations of motion. It differs from molecular dynamics in two respects. First, the conservative pairwise forces between the pseudo-particles are soft-repulsive, which makes it possible to extend the simulations to longer timescales. Second, the system thermostat for the canonical ensemble is implemented by means of the dissipative as well as the random pairwise forces such that the momentum is locally conserved. The pseudo-particle method is used to simulate hydrodynamics at mesoscopic scales in which both, hydrodynamic interactions and Brownian motion are important. At large Mach numbers and large Knudsen numbers it is superior to the cellular automaton models.

The direct simulation Monte Carlo method is also an off-lattice pseudo-particle simulation method [11–16]. The state of the system is given by the positions and velocities of a set of pseudo-particles. First, these fluid or gas portions are moved as if they did not interact. This means that their positions are updated without considering inter-particle collisions. After this translation step a fixed number of particles are randomly selected for collisions. The collision step is typically realized by placing the particles into spatial collision cells, by calculating the collision frequency in each cell, by randomly selecting collision partners within each of those cells and by the actual collisions. The probability that a pair collides only depends on their relative velocity. The actual collisions, i.e. the calculations of the post-collision velocity vectors are determined for each colliding pair by accounting for the conservation of momentum as well as the conservation of energy and by random selection of the collision angle. This splitting of the evolution between forward streaming and collisions is only accurate when the time step elapsing during one update step is a fraction of the mean collision time for a pseudo-particle. The particular strength of the direct simulation Monte Carlo method lies in the field of dilute gases.

1.3. Basic philosophy of lattice-based cellular automaton methods for fluid mechanics

The application of automaton models to the field of fluid dynamics represents a remarkable shift in modelling philosophy when compared to the continuum, molecular dynamics and pseudo-particle approaches. Lattice gas automata replace the macroscopic picture underlying the Navier–Stokes framework by discrete sets of fictive particles which carry some properties of real fluid portions, figure 2 [18, 19]. These fictive particles can be regarded as coarse-grained groups of fluid (or gas) molecules the exact Newtonian dynamics of which are not explicitly taken into account as in molecular dynamics approaches, or, to a certain extent, in the pseudo-particle methods. The fluid portions in the lattice gas move at different speeds in different

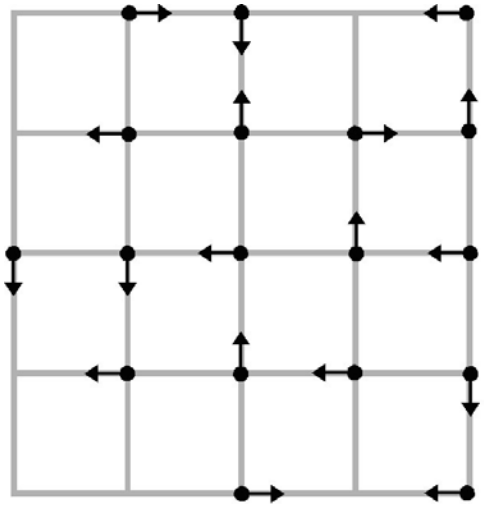

Figure 2. Pseudo-fluid particles in a lattice gas model with a quadratic grid (HPP lattice gas model of Hardy, Pomeau, de Pazzis [18], see details in the next section). All particles have the same unit mass and the same magnitude of the velocity vector (Boolean particles). Motion of the particles consists in translating them from one lattice node to their nearest neighbour in one discrete unit of time according to the direction of their unit momentum vector. The symmetry of the quadratic grid turned out not to be sufficient for the reproduction of the Navier–Stokes equation.

directions on a fixed lattice and interact by simple local rules. During each time step they move according to their current momentum vector. If two particles happen to end up on the same lattice site, they collide and change their velocities according to a set of discrete collision rules. The only restriction is that collisions have to conserve the particle number, the momentum and the energy. Using this small set of rules offers the first and coarsest way of approximating fluid dynamics in terms of lattice gas automata. An important computational advantage of this method is that any lattice node can be marked as solid, allowing for the integration of arbitrarily complex geometries that would be difficult to model with conventional continuum methods owing to convergence problems.

The basic idea of lattice gas models, like generally of cellular automata, is to mimic complex dynamical system behaviour by the repeated application of simple local translation and reaction rules. These rules simulate, in a simplified and coarse-grained mesoscopic fashion, some of the microscopical effects occurring in a real fluid. This means that lattice-gas automata take a microscopic, though not truly molecular, view of fluid mechanics by conducting fictive microdynamics on a lattice.

Solutions of the Navier–Stokes differential continuum equation can be regarded as a *top-down* approach to fluid mechanics for small Knudsen number regimes, while the pseudo-particle and lattice-based automaton methods pursue a *bottom-up* strategy valid also at larger Knudsen numbers. In the macroscopic world of the Navier–Stokes equation one directly analyses the pressure, density, viscosity and velocity of the flow. In the microscopic view taken by the pseudo-particle and lattice gas automata, such macroscopic quantities can be computed by averaging the interaction and density of the pseudo-particles locally. It must be noted though that lattice-gas automata themselves are coarse-grained methods, i.e. the fictive fluid droplets which they use as elementary constituents are averaged pseudo-particles, which do not perform individual Newtonian dynamics as in a molecular dynamics simulation, figures 1 and 3.

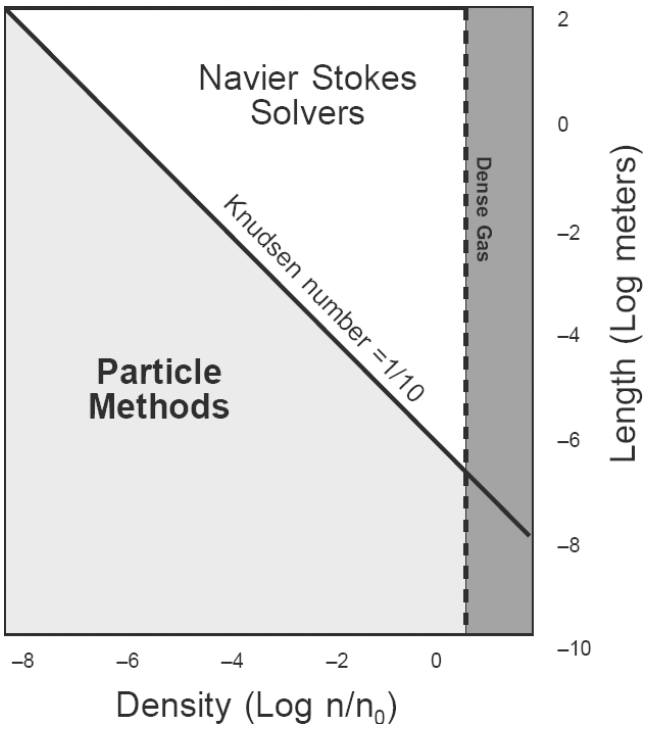

Figure 3. Validity regimes of a gas or fluid simulation method as a function of density relative to air and length scale. The figure shows that the continuum description becomes inaccurate when the characteristic length scale is within an order of magnitude of the mean free path (figure adopted from the works of Bird [20, 21] and Garcia [22].

1.4. Some important measures for flow mechanics

In the field of fluid mechanics one typically uses some elementary mesoscopic and continuum measures for the constitutive, geometrical and dynamical quantification of flows. Some of them are relevant in the context of this article, table 2.

1.5. Boolean lattice gas cellular automata (HPP and FHP models)

Lattice gas cellular automata with Boolean particle states residing on fixed nodes were originally suggested by Frisch, Hasslacher and Pomeau in 1986 (FHP lattice gas model) [19] for the reproduction of Navier–Stokes dynamics. A previous formulation for vector automata was already in 1973 suggested by Hardy, Pomeau and de Pazzis (HPP lattice gas model) [18]. However, this earlier version of a lattice gas method was based on a square grid and could, therefore, not fulfill the requirement of rotational invariance. The FHP lattice gas model published later [19] used a hexagonal two-dimensional lattice which fulfills both, conservation of particle number and rotational invariance.

All particles in a Boolean lattice gas have the same unit mass and the same magnitude of the velocity vector. The model imposes, as an exclusion principle, that no two particles may sit simultaneously on the same node if their direction is identical. For the square lattice originally suggested by the HPP model, this implies that there can be at most four particles per node. This occupation principle, originally meant to permit simple computer codes, has the consequence that the equilibrium distribution of the particles follows a Fermi–Dirac distribution. Motion of

Table 2. Some important measures for flow analysis.

Parameter	Relevance	Definition	Units
Dynamic viscosity (absolute or Newtonian viscosity)	Measure of the internal molecular resistance of a fluid to flow or shear under an applied force	$\tau = \mu_{\mathrm{dyn}} \frac{\dot{\gamma}}{\ell}$ τ: shear stress, μ_{dyn}: dynamic viscosity, $\dot{\gamma}$: shear rate of one layer relative to another, ℓ: spacing of the layers	$\left[\frac{\mathrm{Ns}}{\mathrm{m}^2}\right] = \left[\frac{\mathrm{kg}}{\mathrm{ms}}\right]$ [Poise] $= \left[\frac{\mathrm{g}}{\mathrm{ms}}\right]$
Kinematic viscosity	Viscosity divided by the density of the liquid; force free measure of the viscosity	$\mu_{\mathrm{kin}} = \frac{\mu_{\mathrm{dyn}}}{\rho}$ ρ: mass density	$\left[\frac{\mathrm{m}^2}{\mathrm{s}}\right]$ [Stoke] $= \left[\frac{\mathrm{cm}^2}{\mathrm{s}}\right]$
Knudsen number	Ratio between the mean free molecule path and a characteristic length scale representing mesoscopic system heterogeneity (e.g. obstacle size)	$K = \frac{L_1}{L_2}$ L_1: mean free path of molecule, L_2: characteristic system length	[1]
Mach number	Ratio of the speed of a particle in a medium to the speed of sound in that medium; Mach number 1 corresponds to the speed of sound	$M = \frac{c}{c_{\mathrm{s}}}$ c: speed of particle in a medium, c_{s}: speed of sound in the medium	[1]
Reynolds number	Measure of the relative strength of advective over dissipative forces quantifying the degree of turbulence in a flow	$Re \equiv \frac{F_{\mathrm{inertia}}}{F_{\mathrm{viscous}}} \approx \frac{U\,L}{\mu_{\mathrm{kin}}}$ U: characteristic macroscopic flow speed, L: characteristic length scale of flow geometry	[1]

the particles consists of moving them from one lattice node to their nearest neighbour in one discrete unit of time according to their given unit momentum vector, figure 4.

The evolution of system dynamics of the lattice gas takes place in four successive steps. The first one is the advection or propagation step. It consists of moving all particles from their nodes to their nearest neighbour nodes in the directions of their respective velocity vectors. The second one is the collision step, figure 5. It is conducted in such a way that interactions between particles arriving at the same node coming from different directions take place in the form of local instantaneous collisions. The elastic collision rules conserve both mass and momentum. This implies that particles arriving at the same node may exchange momentum if it is compatible with the imposed invariance rules. The third step is (usually) the bounce-back step. It imposes no-slip boundary conditions for those particles which hit an obstacle. The fourth step updates all sites. This is done by synchronously mapping the new particle coordinates and velocity vectors obtained from the preceding steps onto their new positions. Subsequently the time counter is increased by one unit.

Owing to the discrete treatment of the pseudo-particles and the discreteness of the collision rules Boolean lattice gas automata reveal some intrinsic flaws such as the violation of Galilean invariance[1] and the occurrence of large fluctuations. The latter disadvantage can to a certain extent be circumvented by introducing localized averaging procedures

[1] A Galilean transformation is a change to another inertial reference frame moving with constant velocity. This should not affect the properties of the flow.

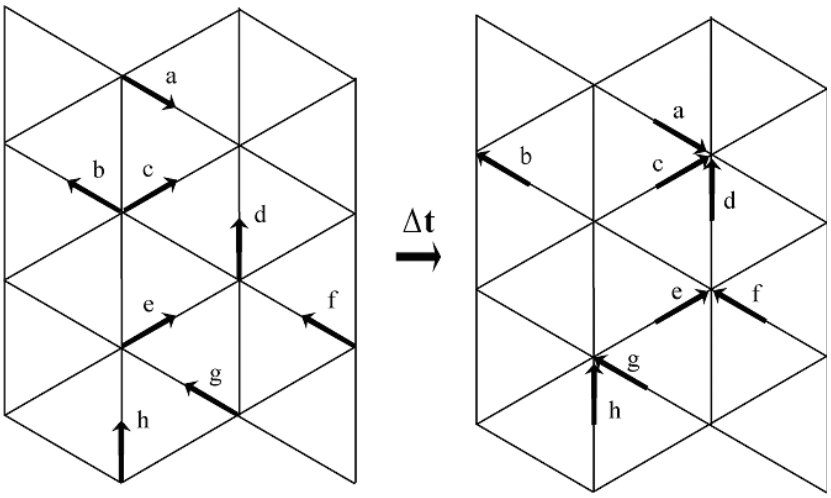

Figure 4. Positions of lattice gas pseudo-particles at two successive time steps (advection only) on a hexagonal two-dimensional lattice (FHP lattice gas model) [19].

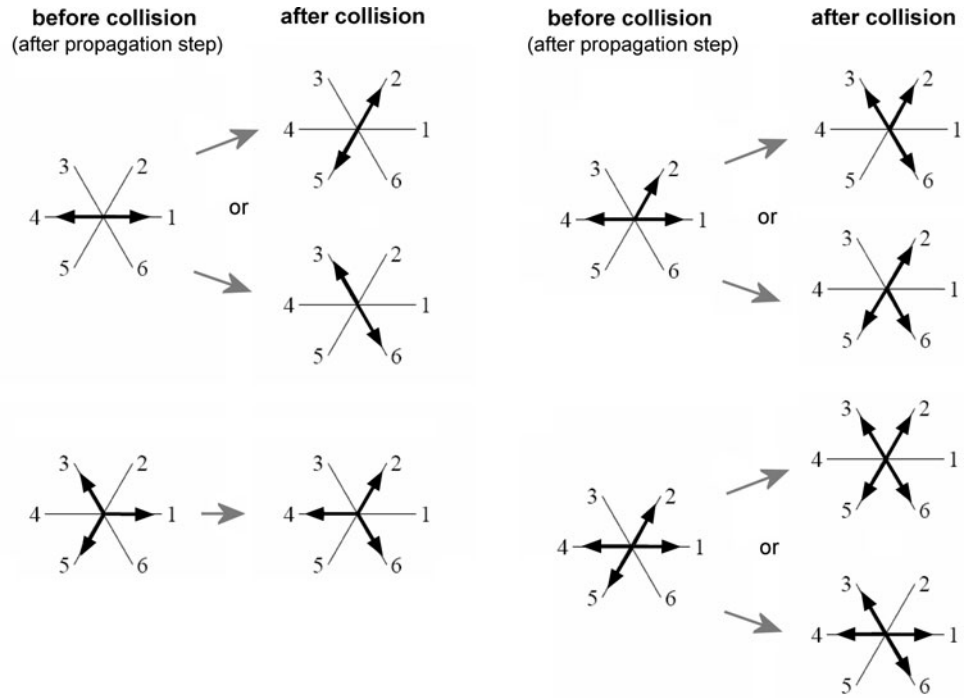

Figure 5. Collision rules of the lattice gas cellular automaton for the case of a hexagonal grid (FHP lattice gas model) [19].

where a group of neighbouring vectors is summarized into a coarse-grained net-vector, figure 6.

The main advantage of the lattice gas concept compared to classical Navier–Stokes solvers consists of its excellent numerical stability under intricate geometrical boundary

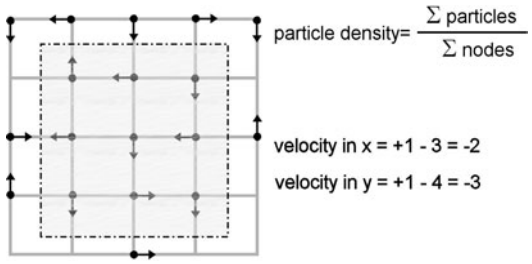

Figure 6. Schematic sketch of averaging in a lattice gas simulation. Such procedures are important in classical Boolean lattice gas simulations for reducing statistical noise.

conditions. This property qualifies them particularly for the simulation of microflow dynamics in porous microstructures and related problems arising in the field of modern materials science and engineering (see examples in part 2 of this article). Since the basic structure of the lattice gas algorithm is that of a synchronous automaton it is also an ideal platform for realizing combinations with related materials simulation methods such as solid state cellular automata or Potts models.

1.6. Introduction to the philosophy of the lattice Boltzmann approach

The lattice Boltzmann approach has evolved from the lattice gas models in order to overcome the shortcomings discussed above. It corresponds to a space-, momentum- and time-discretized version of the Boltzmann transport equation. The main rationale behind the introduction of the lattice Boltzmann automaton is to incorporate the physical nature of fluids from a more statistical standpoint into hydrodynamics solutions than in the classical lattice gas method discussed in the preceding section, table 3. According to the underlying picture of the Boltzmann transport equation the idea of the lattice Boltzmann automaton is to use sets of particle velocity distribution functions instead of single pseudo-particles and to implement the dynamics directly on those average values [23–32]. The particle velocities in the lattice Boltzmann scheme are not Boolean variables as in conventional lattice gas automata [14, 19], but real-numbered quantities as in the Boltzmann transport equation, figure 7. This means that Fermi-like statistics no longer apply. It is also important to note that in contrast to the conventional lattice gas method the lattice Boltzmann approach may use pseudo-particles with zero velocity. These are required for simulating compressible hydrodynamics by using a tunable model sound speed.

Another main difference between the original lattice gas and the lattice Boltzmann methods is the fact that the former approach quantifies the particle interactions in terms of discrete local Boolean redistribution rules (collision rules) while the latter approach conducts (non-Boolean) redistributions of the particle velocity distribution (relaxation rules, collision operator).

The main advantage of the lattice Boltzmann method compared to the original lattice gas is that small sets of neighbouring nodes in a Boltzmann lattice are capable of creating smooth flow dynamics as opposed to the lattice gas methods which entail rather coarse dynamical behaviour. This means that the Boltzmann method requires less averaging and provides increased performance.

1.7. Typical mesh types for the lattice Boltzmann method

Lattice Boltzmann models are spatially *discrete* approaches to fluid dynamics. This means that the underlying grids of such simulations must fulfill certain symmetry conditions in order to

Table 3. Overview of the lattice gas and the lattice Boltzmann model family (see detailed explanations of the lattice types D*k*Q*n* in section 1.7).

The lattice gas and lattice Boltzmann automaton family	
Lattice gas automata	*HPP-model (according to Hardy, Pomeau, and Pazzis).* The original form of the lattice gas automaton with Boolean pseudo-fluid particles residing on a discrete two-dimensional quadratic grid (Hardy *et al* [18])
	FHP-model (according to Frisch, Hasslacher and Pomeau). Two-dimensional lattice gas, hexagonal grid. FHP-I: 6 neighbour nodes; FHP-II: 6 neighbour nodes and one rest particle FHP-III: 6 neighbour nodes, one rest particle and complete collision rules (Frisch *et al* [19], d'Humières and co-workers [35–37])
	FCHC-model (face-centred-hypercubic). FHP-type three-dimensional lattice gas model, four-dimensional Bravais lattice with 24 neighbour nodes projected on a three-dimensional spatial lattice (d'Humières and co-workers [35–37])
	Two-colour FCHC-model (multi-phase model on the basis of FHP-III). FHP model with a two-dimensional hexagonal lattice, two types of (coloured) fluid phases (red, blue), phase separation occurs by the introduction of a local flux and a colour gradient vector (Rothman and Keller [33])
Lattice Boltzmann automata	*LB-model (single-phase lattice Boltzmann model).* Real-numbered particle velocity distribution functions, typical lattice types: D2Q9, D3Q15 and D3Q19, collision matrix (Frisch *et al* [19], McNamara and Zanetti [26])
	LBGK-model (lattice Boltzmann with Bhatnagar–Gross–Krook relaxation). The lattice Boltzmann model, assumption of an equilibrium velocity distribution, collision matrix replaced by a single-step collision relaxation towards equilibrium (Bhatnagar *et al* [34], Higuera and Jimenez [27], Qian *et al* [38])
	Multi-phase LBGK-model (multi-phase model on the basis of LBGK). LBGK models for multi-phase applications using single-step relaxation and gradient terms, pseudo-potentials, or free-energy functionals for phase separation (Gunstensen *et al* [39, 40], Grunau *et al* [41], Shan and Chen [42, 43], Shan and Doolen [44, 45])

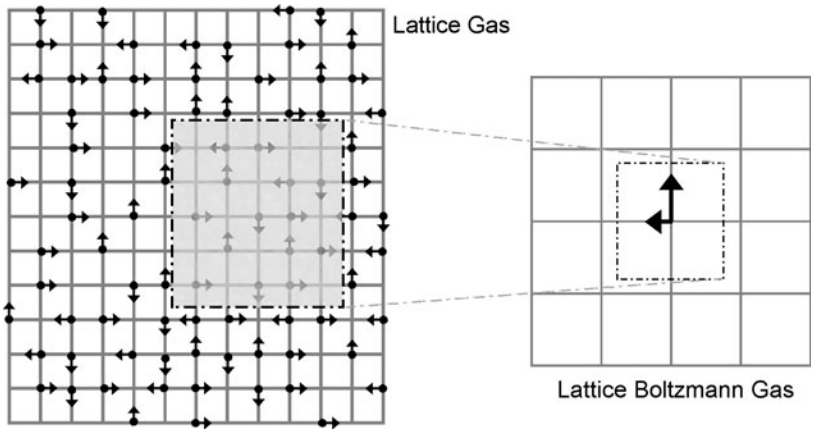

Figure 7. Schematic example demonstrating the idea of a coarse-graining procedure which renders a Boolean lattice gas into a lattice *Boltzmann* gas. The arrows residing at the nodes on the two grids represent discrete lattice particles (lattice gas, left-hand side) and portions of a local vector distribution function, respectively (lattice Boltzmann method, right-hand side). The figure shows that the lattice gas method works with discrete velocity vectors and discrete fluid portions. The lattice Boltzmann gas works also with discrete velocity vector directions but it uses real-numbered fluid portions.

Table 4. Overview of the weight factors w_i for the most important lattice types.

Lattice model	Zero position	Simple cubic vectors [100]	Diagonal vectors [110]	Cubic vectors [111]
D2Q9	4/9	1/9	1/36	do not exist
D3Q15	2/9	1/9	do not exist	1/72
D3Q19	1/3	1/18	1/36	do not exist

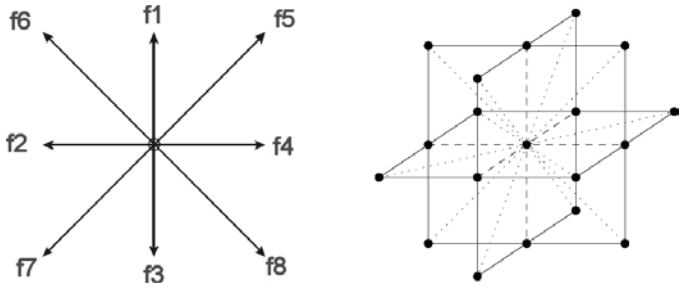

Figure 8. Velocity vectors for a D2Q9- (left) and a D3Q19-lattice geometry (right).

recover hydrodynamic behaviour with full rotational symmetry of space. This requires that the invariance measures which form from the respective sets of underlying lattice vectors up to fourth order are isotropic, i.e.

$$\begin{aligned}
&\sum_{i=0}^{n} w_i = 1, \qquad \sum_{i=0}^{n} w_i e_{i\alpha} = 0, \\
&\sum_{i=0}^{n} w_i e_{i\alpha} e_{i\beta} = \Pi^{(2)} \delta_{\alpha\beta}, \qquad \sum_{i=0}^{n} w_i e_{i\alpha} e_{i\beta} e_{i\gamma} = 0, \\
&\sum_{i=0}^{n} w_i e_{i\alpha} e_{i\beta} e_{i\gamma} e_{i\theta} = \Pi^{(4)} (\delta_{\alpha\beta}\delta_{\gamma\theta} + \delta_{\alpha\gamma}\delta_{\beta\theta} + \delta_{\alpha\theta}\delta_{\beta\gamma}),
\end{aligned} \tag{1}$$

where w_i represent weight factors which must be properly chosen for each grid type in order to correct the lattice with respect to isotropy, $\vec{e}_i$ are the lattice vectors with the Greek indices $\alpha, \beta, \gamma, \theta$ for the spatial directions and hyper-directions, and $\Pi^{(2)}$ and $\Pi^{(4)}$ are lattice constants which are related to the lattice sound speed c_s. The index i is the counter for the lattice vectors.

The most frequent mesh types for lattice Boltzmann simulations are the D1Q3-, the D2Q9-, the D3Q15- and the D3Q19-lattice, table 4, figure 8. The terminology DkQn refers to the number k of dimensional sublattices (equivalent to the number of independent speeds) and to the discrete number n of spatial translation vectors $\vec{e}_i$ constituting the vector basis of the distribution function. In three dimensions, isotropy generally requires a multi-speed lattice. Like for all lattice gas automata the units of the corresponding set of velocity vectors, $\vec{c}_i$, are calculated by the corresponding lattice vectors, $\vec{e}_i$, divided by the time step (time proceeds synchronously for all nodes in discrete steps, Δt, as in all automata). The number of occurring speeds, therefore, corresponds to the number of sublattice vector types. It is important to note that $\vec{c}_0$ is a zero vector.

The D1Q3-lattice has 1 sublattice and 3 discrete velocity vectors (identity, left and right). The D2Q9-lattice has 2 sublattices and 9 discrete velocity vectors (identity, north, west, south,

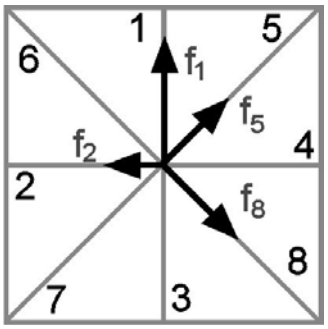

Figure 9. Schematic figure showing some non-zero vectors of the particle velocity distribution function at a node. The two-dimensional lattice has 9 velocity vectors (8 neighbours and a zero velocity). Zero velocity vectors are required for simulating compressible flows.

east, northwest, southwest, southeast and northeast). The basis vectors of the D2Q9-lattice are

$$\vec{e}_i = \vec{c}_i \Delta t = (\vec{e}_1, \vec{e}_2, \vec{e}_3, \vec{e}_4, \vec{e}_5, \vec{e}_6, \vec{e}_7, \vec{e}_8, \vec{e}_9) = \begin{pmatrix} 0 & 1 & 0 & -1 & 0 & 1 & -1 & 1 & -1 \\ 0 & 0 & 1 & 0 & -1 & 1 & 1 & -1 & -1 \end{pmatrix}. \tag{2}$$

The D3Q15-lattice has 3 sublattices and 15 discrete velocity vectors (identity, 6 towards face centres and 8 towards vertices of a cube). The basis vectors of the D3Q15-lattice are

$$\vec{e}_i = \vec{c}_i \Delta t = (\vec{e}_1, \vec{e}_2, \vec{e}_3, \vec{e}_4, \vec{e}_5, \vec{e}_6, \vec{e}_7, \vec{e}_8, \vec{e}_9, \vec{e}_{10}, \vec{e}_{11}, \vec{e}_{12}, \vec{e}_{13}, \vec{e}_{14}, \vec{e}_{15})$$

$$= \begin{pmatrix} 0 & 1 & 0 & 0 & -1 & 0 & 0 & 1 & -1 & 1 & 1 & -1 & 1 & -1 & -1 \\ 0 & 0 & 1 & 0 & 0 & -1 & 0 & 1 & 1 & -1 & 1 & -1 & -1 & 1 & -1 \\ 0 & 0 & 0 & 1 & 0 & 0 & -1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & 1 \end{pmatrix}. \tag{3}$$

The D3Q19-lattice has 3 sublattices and 19 discrete velocity vectors (identity, 6 velocities to the face centres and 12 towards edge centres of a cube).

$$\vec{e}_i = \vec{c}_i \Delta t = (\vec{e}_1, \vec{e}_2, \vec{e}_3, \vec{e}_4, \vec{e}_5, \vec{e}_6, \vec{e}_7, \vec{e}_8, \vec{e}_9, \vec{e}_{10}, \vec{e}_{11}, \vec{e}_{12}, \vec{e}_{13}, \vec{e}_{14}, \vec{e}_{15}, \vec{e}_{16}, \vec{e}_{17}, \vec{e}_{18}, \vec{e}_{19})$$

$$= \begin{pmatrix} 0 & 1 & 0 & 0 & -1 & 0 & 0 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & -1 & 0 & 1 & 1 & -1 & -1 & 0 & 0 & 0 & 0 & 1 & -1 & 1 & -1 \\ 0 & 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \end{pmatrix}. \tag{4}$$

While the D3Q15-lattice model requires less computation and less memory than the D3Q19-lattice model, it suffers more from finite size effects and is less accurate.

1.8. Formal description of the lattice Boltzmann method for single-phase flow

The lattice Boltzmann gas uses as a central quantity a particle velocity distribution function, $f_i(\vec{x}, t)$ [24, 26], which quantifies the (real-numbered) probability to observe a pseudo-fluid particle with discrete velocity $\vec{c}_i$ at lattice node $\vec{x}$ at time t. The particle velocity distribution function is defined for particles moving synchronously along the nodes of a discrete regular spatial lattice. The subscripts $i = 0, \ldots, m$ of the velocity vectors indicate their discrete lattice direction on the chosen grid, figure 9. The occurring velocity vectors depend on the number of sublattices and on the coordination sphere as outlined in the preceding section. The fluid particles can collide with each other as they move under applied forces.

In the lattice Boltzmann approach the temporal evolution of the particle velocity distribution function satisfies a lattice Boltzmann equation of the type

$$f_i^{\text{new}}(\vec{x} + \vec{c}_i \Delta t, t + \Delta t) - f_i^{\text{old}}(\vec{x}, t) = \Delta t \Omega_i \qquad (i = 0, \ldots, n), \tag{5}$$

where Δt is the lattice time step. The index i stands for the n base vectors of the underlying lattice type. The left hand term, $f_i^{\text{new}}(\vec{x} + \vec{c}_i \Delta t, t + \Delta t) - f_i^{\text{old}}(\vec{x}, t)$, is the advection term which represents free propagation of the particle packets along the lattice links. The term $f_i^{\text{new}}(\vec{x} + \vec{c}_i \Delta t, t + \Delta t)$ is the new distribution function after advection and redistribution. When considering an additional external source of momentum (e.g. body forces such as occurring in pressure gradients or gravitational fields), F_i, one obtains

$$f_i^{\text{new}}(\vec{x} + \vec{c}_i \Delta t, t + \Delta t) - f_i^{\text{old}}(\vec{x}, t) = \Delta t \Omega_i + \Delta t F_i \qquad (i = 0, \ldots, n). \tag{6}$$

Normalization provides

$$f_i \in [0, 1]. \tag{7}$$

The symbol Ω_i represents the collision operator. In the first variant of the lattice Boltzmann model suggested by McNamara and Zanetti [26], collisions where formulated as direct transcriptions of the lattice gas approach, i.e. complete fluid particles were exchanged between the different lattice vectors without altering their mass content. By using this method the particles' mean free path and thus the fluid viscosity were still fixed. Releasing that constraint and allowing the exchange of matter between the fluid packets leads to variable viscosity. In this case the collision operator is a vector. The collision term may be linearized by assuming that there is always a local equilibrium particle distribution, $f_i^{\text{eq}}(\vec{x}, t)$, which depends only on the locally conserved mass and momentum density. A first-order approximation for the collision operator yields

$$\Omega_i^*(f_i^{\text{old}}(\vec{x}, t)) = \Omega_i^*(f_i^{\text{eq}}(\vec{x}, t)) + \Omega_i(f_i^{\text{old}}(\vec{x}, t) - f_i^{\text{eq}}(\vec{x}, t)). \tag{8}$$

In order to ensure the local conservation of mass and momentum the equilibrium distributions must at each node satisfy

$$\sum_{i=0}^{n} f_i^{\text{eq}}(\vec{x}, t) = \sum_{i=0}^{n} f_i^{\text{old}}(\vec{x}, t) \tag{9}$$

and

$$\sum_{i=0}^{n} f_i^{\text{eq}}(\vec{x}, t)\vec{c}_i = \sum_{i=0}^{n} f_i^{\text{old}}(\vec{x}, t)\vec{c}_i, \tag{10}$$

where i stands for the lattice vectors. Since Ω_i now only acts on the departure from equilibrium, the first term in the first-order approximation for the collision operator, namely $\Omega_i^*(f_i^{\text{eq}}(\vec{x}, t))$, vanishes. A convenient formulation for the remainder, used by most current versions of the lattice Boltzmann automaton, has the form of a single-step relaxation as suggested by the Bhatnagher–Gross–Krook approximation [32], namely,

$$\Omega_i = -\frac{1}{\tau}(f_i^{\text{old}}(\vec{x}, t) - f_i^{\text{eq}}(\vec{x}, t)). \tag{11}$$

In this expression the relaxation time, τ, is a parameter which quantifies the rate of change towards local equilibrium for incompressible isothermal materials. The Bhatnagher–Gross–Krook relaxation yields maximal local randomization. All particle distributions relax at the same rate, $\omega = 1/\tau$, towards their corresponding equilibrium value. As was first pointed out by Qian *et al* [38] the relaxation rate must obey $0 < \omega < 2$ for the method to be stable and for the particle density and viscosity to be positive. The condition where $0 < \omega < 1$ is

called sub-relaxation regime while $1 < \omega < 2$ is referred to as over-relaxation regime. The use of a Bhatnagher–Gross–Krook single-step relaxation scheme has replaced the use of a discrete collision matrix which had to be formulated for collisions in earlier lattice gas models. The lattice Boltzmann equation in the single-step relaxation approximation corresponds to the discrete form of the classical Chapman–Enskog first-order Taylor expansion of the Boltzmann equation.

For non-isothermal flows or fluids with variable density the relaxation time may deviate from a constant according to

$$\tau^* = \frac{1}{2} + \frac{1}{\xi(\vec{x}, t, T)} \left(\tau - \frac{1}{2} \right), \tag{12}$$

where T is the temperature and $\xi(\vec{x}, t, T)$ is the local particle density which can be calculated as the local sum over the particle velocity distribution according to

$$\xi(\vec{x}, t, T) = \sum_{i=0}^{n} f_i(\vec{x}, t, T). \tag{13}$$

The relaxation time is a parameter which characterizes the constitutive behaviour of the fluent material at a microscopic level. It is connected with the macroscopic kinematic viscosity of the simulated fluid according to

$$\nu = c_s^2 \Delta t \left(\tau^* - \tfrac{1}{2} \right), \tag{14}$$

which reduces to

$$\nu = \frac{2\tau - 1}{6} \tag{15}$$

for incompressible isothermal flows where $c_s = 1/\sqrt{3}$ is the lattice sound speed.

In the lattice Boltzmann Bhatnagher–Gross–Krook method (LBGK) the particle distribution after propagation is relaxed towards the local equilibrium particle distribution function. The equilibrium distribution, $f_i^{\mathrm{eq}}(\vec{x}, t)$, depends, in the Bhatnagher–Gross–Krook approximation, only on locally conserved quantities such as mass density and momentum density. It is carefully chosen so that Galilean invariance and the isothermal Navier–Stokes equation in the incompressible fluid limit are recovered.

Macroscopic parameters are determined by the integration of the distribution functions. These integrands are referred to as moments. Important in that context are those parameters which are relevant with respect to local conservation laws, namely, the local particle density, $\xi(\vec{x}, t)$, the local mass density, $\rho(\vec{x}, t)$, and the local velocity vector, $\vec{u}(\vec{x}, t)$, which relates to the momentum density vector, $\rho(\vec{x}, t)\vec{u}(\vec{x}, t)$, and the local kinetic energy density, $\vartheta(\vec{x}, t)$. They can be calculated as moments of the particle distribution according to

$$\begin{aligned}
&\xi(\vec{x}, t) = \sum_{i=0}^{n} f_i(\vec{x}, t), \\
&\rho(\vec{x}, t) = m \sum_{i=0}^{n} f_i(\vec{x}, t) = m\xi(\vec{x}, t), \\
&\vec{u}(\vec{x}, t) = \frac{1}{\xi(\vec{x}, t)} \sum_{i=0}^{n} f_i(\vec{x}, t)\vec{c}_i = \frac{\sum_{i=0}^{n} f_i(\vec{x}, t)\vec{c}_i}{\sum_{i=0}^{n} f_i(\vec{x}, t)}, \\
&\rho(\vec{x}, t)\vec{u}(\vec{x}, t) = m \sum_{i=0}^{n} f_i(\vec{x}, t)\vec{c}_i, \\
&\vartheta(\vec{x}, t) = \frac{1}{2} \sum_{i=0}^{n} f_i(\vec{x}, t)|\vec{c}_i - \vec{u}(\vec{x}, t)|^2 = \xi(\vec{x}, t)\frac{D}{2}RT = \frac{\rho(\vec{x}, t)}{m}\frac{D}{2}RT,
\end{aligned} \tag{16}$$

where m is the mass of a lattice particle, D is the dimension of the momentum space of the discrete lattice velocities and R is the gas constant. The last equation defines the local temperature.

The moments must be conserved during the collision phase, i.e. the velocity moments of the collision term must vanish at each node according to

$$\sum_{i=0}^{n} \Omega_i = 0 \qquad \text{and} \qquad \sum_{i=0}^{n} \Omega_i \vec{c}_i = 0. \tag{17}$$

The momentum density tensor, $M^{\text{eq}}_{\alpha\beta}$, can be calculated as the second moment of the distribution function according to the equation

$$M^{\text{eq}}_{\alpha\beta} = m \sum_{i=0}^{n} c_{i\alpha} c_{i\beta} f_i^{\text{eq}}(\vec{x}, t), \tag{18}$$

where α and β are lateral cartesians of the i different velocity vectors $\vec{c}_i$. The strain rate tensor can be approximated as the first-order symmetric part of the velocity gradient tensor according to the equation

$$\dot{\varepsilon}_{\alpha\beta}(\vec{x}, t) = \frac{1}{2}\left(\frac{\partial u_\alpha(\vec{x}, t)}{\partial x_\beta(\vec{x}, t)} + \frac{\partial u_\beta(\vec{x}, t)}{\partial x_\alpha(\vec{x}, t)}\right). \tag{19}$$

Rotation rates can be approximated as the first-order skew-symmetric part of the velocity gradient tensor

$$\dot{\Phi}_{\alpha\beta}(\vec{x}, t) = \frac{1}{2}\left(\frac{\partial u_\alpha(\vec{x}, t)}{\partial x_\beta(\vec{x}, t)} - \frac{\partial u_\beta(\vec{x}, t)}{\partial x_\alpha(\vec{x}, t)}\right). \tag{20}$$

The equilibrium distribution for an incompressible isothermal fluid, $f_i^{\text{eq}}(\vec{x}, t)$, which approximates the Maxwell–Boltzmann equilibrium distribution up to a second-order Taylor series, can be written as

$$f_i^{\text{eq}}(\vec{x}, t) = w_i \xi(\vec{x}, t)[c_1 + c_2(\vec{c}_i \vec{u}) + c_3(\vec{c}_i \vec{u})^2 + c_4\,(\vec{u}\vec{u})], \tag{21}$$

where c_1, c_2, c_3, c_4 are lattice constants which depend on the lattice type and the lattice sound speed, c_s, as $c_1 = 1$, $c_2 = 1/c_s^2$, $c_3 = 1/(2c_s^4)$, and $c_4 = -1/c_s^2$. The symbols w_i represent the weight factors.

The strength of the lattice Boltzmann method is its computational simplicity and its locality. The latter aspect is an advantage which predestines the method for parallelization. The algorithm requires information about the distribution function only at nearby points in space. It allows one to treat flows under complex boundary conditions in simple terms by accounting for reflections and bounces at appropriate spatial locations flagged as solid obstacles. The use of an averaged quantity, $f_i(\vec{x}, t)$, as a central state variable avoids statistical noise so that finite size effects practically do not occur. As in all automata, the set of allowed velocities in the lattice Boltzmann models is constrained by conservation of mass and momentum, and by the requirement of rotational symmetry (isotropy). However, these restrictions turn out to be much less severe than in the lattice gas cellular automaton models. This means that lattice effects practically do not occur in lattice Boltzmann models. By using a small velocity Chapman–Enskog expansion one can show that the lattice Boltzmann formulation as outlined in this chapter reproduces the Navier–Stokes equation for incompressible flows in the limit of small Knudsen and low Mach numbers (below 0.15), figures 1 and 3.

1.9. The lattice Boltzmann method for multi-phase flows

1.9.1. Introduction. Multi-phase flow phenomena are characterized by movable and deformable phase boundaries at which the properties of the flow may discontinuously change. One assumes further that the flows do not evaporate. An essential feature of immiscible multi-phase flows (as for multi-phase solids) is the occurrence of a Laplacian surface tension for each of the phases which guides the system towards the reduction of interface energy.

Three types of lattice Boltzmann approaches have been suggested for the simulation of complex reaction-free multi-phase flows, namely, the chromodynamic (colour), the pseudo-potential and the free-energy models. The following sections will give a brief introduction to these different approaches. An excellent discussion of the different approaches to lattice-based multi-phase flows is given in [24].

1.9.2. Chromodynamic or colour models of multi-phase lattice Boltzmann flows. The first lattice-based model for immiscible two-phase flow was proposed by Rothman and Keller [33]. It was formulated as a lattice gas approach. The authors used as a starting point the single-phase FHP model with hexagonal lattice and introduced two types of (coloured) fluid phases, termed *red* and *blue* (hence the term *chromodynamic or colour models*). Phase separation was, in their approach, introduced by a local flux and colour gradient term. The work of the colour flux against the field minimum was chosen to encourage the preferential grouping of identical phases. Owing to its descend from the Boolean lattice gas method, the original form of the Rothman–Keller model suffered from the deficiencies associated with lattice artefacts and noise, discussed above.

A later version of a lattice-based two-phase chromodynamic flow model was the two-phase lattice Boltzmann model of Gunstensen and co-workers [39, 40]. This method was inspired by the original Rothman–Keller lattice gas scheme, but it was based on the lattice Boltzmann method of McNamara and Zanetti [26] in conjunction with the linearized collision operator proposed by Higuera and Jimenez [27]. Although the unphysical properties like the lack of Galilean invariance and statistical noise inherent to the lattice gas were overcome, the pressure was still velocity dependent in this approach. In addition, the linearized collision operator was not computationally efficient and the model could not handle two fluids with different densities and viscosities. Grunau *et al* [41] developed the model further by introducing a single-time relaxation approximation with a proper particle equilibrium distribution function. These modifications eliminated the problems of the formulation of Gunstensen and co-workers [39, 40]. The following presentation follows, therefore, essentially the approach of Grunau *et al* [41].

Multi-phase lattice models use at least two separate phases. Each phase is characterized in terms of an individual particle distribution function and individual *equilibrium* particle distribution function. This means that the overall particle occupation state at each node is described by a set of particle velocity distribution functions, each of which follows in its dynamic evolution a lattice Boltzmann equation, with an individual collision operator for each phase. It is important to note that these individual collision operators do not describe the interactions between dissimilar pseudo-particles. In the simplest case of a two-component flow, the phases and the associated particle populations are traditionally labelled by colours [33, 39–41]. The two separate particle velocity distribution functions are $f_i^{\mathrm{red}}(\vec{x}, t)$ and $f_i^{\mathrm{blue}}(\vec{x}, t)$. Generalization to multi-phase flows leads to $f_i^{\phi}(\vec{x}, t)$, where the index ϕ refers to the k different fluid phases (running from $\phi = 1$ to k). The index i refers to the ith velocity vector, $\vec{x}$ to the lattice node position, and t to time.

Assuming only a red and a blue phase these particle distributions are evolved by a set of modified lattice Boltzmann equations of the following form

$$\begin{aligned} f_i^{\text{new,red}}(\vec{x}+\vec{c}_i\Delta t, t+\Delta t) - f_i^{\text{old,red}}(\vec{x},t) &= \Delta t\Omega_i^{\text{red}} + S_i^{\text{red}}, \\ f_i^{\text{new,blue}}(\vec{x}+\vec{c}_i\Delta t, t+\Delta t) - f_i^{\text{old,blue}}(\vec{x},t) &= \Delta t\Omega_i^{\text{blue}} + S_i^{\text{blue}}, \end{aligned} \tag{22}$$

where Ω_i^{red} and Ω_i^{blue} are the individual single-phase collision operators for the two phases. They have the conventional form of the single-step Bhatnagher–Gross–Krook relaxation [32] according to

$$\begin{aligned} \Omega_i^{\text{red}} &= -\frac{1}{\tau^{\text{red}}}[f_i^{\text{red}}(\vec{x},t) - f_i^{\text{red,eq}}(\vec{x},t)], \\ \Omega_i^{\text{blue}} &= -\frac{1}{\tau^{\text{blue}}}[f_i^{\text{blue}}(\vec{x},t) - f_i^{\text{blue,eq}}(\vec{x},t)] \end{aligned} \tag{23}$$

with the characteristic relaxation times τ^{red} and τ^{blue}, and the equilibrium distributions $f_i^{\text{red,eq}}(\vec{x},t)$ and $f_i^{\text{blue,eq}}(\vec{x},t)$. One should note that the viscosity of each fluid phase can be individually selected by choosing the desired relaxation times for that phase, since the corresponding operators account for collisions with particles of the same type only. The particle velocity equilibrium distribution for each individual phase, $f_i^{\text{red,eq}}(\vec{x},t)$, $f_i^{\text{blue,eq}}(\vec{x},t)$, depends (in all multi-phase flow lattice models) on the local macroscopic variables pertaining to that phase, i.e. $\xi^\phi(\vec{x},t)$, $\rho^\phi(\vec{x},t)$ and $\vec{u}^\phi(\vec{x},t)$. The equilibrium distributions can, hence, be written as

$$f_i^{\phi,\text{eq}}(\vec{x},t) = w_i\xi^\phi(\vec{x},t)[c_1 + c_2(\vec{c}_i\vec{u}^\phi) + c_3(\vec{c}_i\vec{u}^\phi)^2 + c_4\,(\vec{u}^\phi\,\vec{u}^\phi)], \tag{24}$$

where the index $\phi = 1,\ldots,k$ refers to the k different fluid components and c_1, c_2, c_3, c_4 are the Taylor expansion coefficients $c_1 = 1$, $c_2 = 1/c_{\text{s}}^2$, $c_3 = 1/2c_{\text{s}}^4$ and $c_4 = -1/c_{\text{s}}^2$. The relevant moments of the individual flows and of the total flow are

$$\begin{aligned} \xi^\phi(\vec{x},t) &= \sum_{i=0}^{n} f_i^\phi(\vec{x},t) = \sum_{i=0}^{n} f_i^{\phi,\text{eq}}(\vec{x},t), \\ \rho^\phi(\vec{x},t) &= m^\phi \sum_{i=0}^{n} f_i^\phi(\vec{x},t) = m^\phi \sum_{i=0}^{n} f_i^{\phi,\text{eq}}(\vec{x},t) = m^\phi\xi^\phi(\vec{x},t), \\ \rho(\vec{x},t) &= \sum_{\phi=1}^{k} \rho^\phi(\vec{x},t), \\ \vec{u}^\phi(\vec{x},t) &= \frac{1}{\xi^\phi(\vec{x},t)} \sum_{i=0}^{n} f_i^\phi(\vec{x},t)\vec{c}_i = \frac{\sum_{i=0}^{n} f_i^\phi(\vec{x},t)\vec{c}_i}{\sum_{i=0}^{n} f_i^\phi(\vec{x},t)}, \\ \rho(\vec{x},t)\vec{u}(\vec{x},t) &= \sum_{\phi=1}^{k} m^\phi \sum_{i=0}^{n} f_i^\phi(\vec{x},t)\vec{c}_i = \sum_{\phi=1}^{k} m^\phi \sum_{i=0}^{n} f_i^{\phi,\text{eq}}(\vec{x},t)\vec{c}_i, \end{aligned} \tag{25}$$

where m^ϕ is the mass of the constituent particles pertaining to the ϕth flow and $\rho(\vec{x},t)\vec{u}(\vec{x},t)$ is the total local momentum vector of the multi-phase flow.

The source terms S_i^{red} and S_i^{blue} in the lattice Boltzmann equation represent the interaction between the two phases, i.e. they must be designed to capture the phase separation and coarsening dynamics of the flow. In the chromodynamic model the source terms are defined in such a way that they influence the configuration of neighbouring sites enabling the pressure tensor to become anisotropic near the fluid interface.

The main ingredients to the formulation of these inter-phase interaction operators are, in the *colour* approach [33, 39–41], the chromodynamic or colour current vector $\vec{K}$

$$\vec{K}(\vec{x},t) = \sum_{i=0}^{n} [f_i^{\text{blue}}(\vec{x},t) - f_i^{\text{red}}(\vec{x},t)]\vec{e}_i = \vec{K}^{\text{blue}}(\vec{x},t) - \vec{K}^{\text{red}}(\vec{x},t) \tag{26}$$

and the chromodynamic or colour gradient vector $\vec{G}$

$$\vec{G}(\vec{x},t) = \sum_{i=0}^{n} [\rho^{\text{blue}}(\vec{x}+\vec{e}_i,t) - \rho^{\text{red}}(\vec{x}+\vec{e}_i,t)]\vec{e}_i. \tag{27}$$

According to the lattice Boltzmann models of Gunstensen and co-workers [39, 40] and Grunau *et al* [41] the two source terms, S_i^{red} and S_i^{blue}, can be written as

$$S_i^{\text{red}} = A^{\text{red}} \frac{\vec{G}_i^2(\vec{x},t) - \zeta \vec{G}_i^2}{|\vec{G}|}, \qquad S_i^{\text{blue}} = A^{\text{blue}} \frac{\vec{G}_i^2(\vec{x},t) - \zeta \vec{G}_i^2}{|\vec{G}|}. \tag{28}$$

In this heuristic quadratic approach $|\vec{G}|$ is the magnitude of the chromodynamic gradient vector, $\vec{G}_i = \vec{G}\,\vec{e}_i$ is the projection of the gradient vector along the lattice node direction $\vec{e}_i$, ζ is a constant proportional to the square of the lattice speed of sound, and A^{red} and A^{blue} are two adjustable parameters which control the surface tension for the two (or more) phases. This formulation shows that the interaction rules redirect the momentum of the components according to the gradient of a colour field which is defined by the spatial distribution of the phases. One should also note that the colour gradient vanishes in each single-phase region of the incompressible flow. Therefore, the source terms only contribute to interfaces and mixing regions.

The moments for the two flow phases amount to

$$\begin{aligned}
&\xi^{\text{red}}(\vec{x},t) = \sum_{i=0}^{n} f_i^{\text{red}}(\vec{x},t), \qquad \xi^{\text{blue}}(\vec{x},t) = \sum_{i=0}^{n} f_i^{\text{blue}}(\vec{x},t),\\
&\xi(\vec{x},t) = \xi^{\text{red}}(\vec{x},t) + \xi^{\text{blue}}(\vec{x},t),\\
&\rho^{\text{red}}(\vec{x},t) = m^{\text{red}}\xi^{\text{red}}(\vec{x},t), \qquad \rho^{\text{blue}}(\vec{x},t) = m^{\text{blue}}\xi^{\text{blue}}(\vec{x},t),\\
&\rho(\vec{x},t) = \rho^{\text{red}}(\vec{x},t) + \rho^{\text{blue}}(\vec{x},t),
\end{aligned} \tag{29}$$

where $\xi^{\text{red}}(\vec{x},t)$ and $\xi^{\text{blue}}(\vec{x},t)$ are the particle densities of the two flows, and $\xi(\vec{x},t)$ is the total particle density at lattice point $\vec{x}$ and time t. The quantities $\rho^{\text{red}}(\vec{x},t)$, $\rho^{\text{blue}}(\vec{x},t)$, and $\rho(\vec{x},t)$ are the corresponding mass densities. The local velocities are

$$\begin{aligned}
&\vec{u}^{\text{red}}(\vec{x},t) = \frac{1}{\rho^{\text{red}}(\vec{x},t)} \sum_{i=0}^{n} f_i^{\text{red}}(\vec{x},t)\vec{c}_i, \qquad \vec{u}^{\text{blue}}(\vec{x},t) = \frac{1}{\rho^{\text{blue}}(\vec{x},t)} \sum_{i=0}^{n} f_i^{\text{blue}}(\vec{x},t)\vec{c}_i,\\
&\vec{u}(\vec{x},t) = \frac{1}{\rho(\vec{x},t)} \sum_{i=0}^{n} (f_i^{\text{red}}(\vec{x},t) + f_i^{\text{blue}}(\vec{x},t))\vec{c}_i,
\end{aligned} \tag{30}$$

where $\vec{u}^{\text{red}}(\vec{x},t)$, $\vec{u}^{\text{blue}}(\vec{x},t)$ and $\vec{u}(\vec{x},t)$ are the corresponding local velocities at lattice point $\vec{x}$ and time t.

1.9.3. Pseudo-potential models of multi-phase lattice Boltzmann flows. An alternative to the chromodynamic approach of Rothman and Keller [33], Gunstensen and co-workers [39, 40] and Grunau *et al* [41] for the lattice-based simulation of multi-phase flows was suggested

by Shan and Chen [42, 43] and Shan and Doolen [44, 45]. Their formulation uses a pseudo-potential model and introduces a non-local interaction force between dissimilar flow particles. This potential is essential for the description of non-ideal fluids since it controls the form of the resulting equation of state of the fluid as well as the kinetics of phase separation. The model of Shan and Chen can be readily formulated for an arbitrary number of phases consisting of particles with different molecular masses.

The evolution of the particle populations follow, for the k different fluid components at each lattice node, a form of the lattice Boltzmann equation with different relaxation and interparticle interaction properties, as outlined above for the chromodynamic model, according to

$$\begin{aligned} f_i^{\phi=1,\mathrm{new}}(\vec{x}+\vec{c}_i\Delta t, t+\Delta t) - f_i^{\phi=1,\mathrm{old}}(\vec{x},t) &= -\frac{\Delta t}{\tau^{\phi=1}}(f_i^{\phi=1,\mathrm{old}}(\vec{x},t) - f_i^{\phi=1,\mathrm{eq}}(\vec{x},t)) \\ &\quad + S_i^{\phi=1}, \ldots f_i^{\phi=k,\mathrm{new}}(\vec{x}+\vec{c}_i\,\Delta t, t+\Delta t) - f_i^{\phi=k,\mathrm{old}}(\vec{x},t) \\ &= -\frac{\Delta t}{\tau^{\phi=k}}(f_i^{\phi=k,\mathrm{old}}(\vec{x},t) - f_i^{\phi=k,\mathrm{eq}}(\vec{x},t)) + S_i^{\phi=k}, \end{aligned} \tag{31}$$

where $\tau^1, \tau^2, \ldots, \tau^k$ are the relaxation parameters for the $1, 2, \ldots, k$ individual flows (i.e. they do not account for interactions among dissimilar particle types). The index i stands for the n base vectors of the respective lattice type. The index ϕ refers to the different fluid phases (running from $\phi = 1$ to $\phi = k$). The equilibrium distributions for the individual phases are in the model of Shan and co-workers [42–45] formulated in the same way as outlined above for the chromodynamic model, equation (24).

The source terms $S_i^{\phi=1} \cdots S_i^{\phi=k}$ describe the interaction between the phases using a pseudo-potential formulation. They are in the model of Shan and co-workers [42–45] usually formulated in the following way

$$S_i^{\phi} = \vec{F}^{\phi}\vec{e}_i, \tag{32}$$

where S_i^{ϕ} is the interaction source term for phase ϕ in the direction of the lattice vector $\vec{e}_i$ and $\vec{F}^{\phi}$ is the total effective interparticle force vector acting on the ϕth component associated with the pseudo-potential of the pairwise interaction between different particle types. Interactions between *identical* particle types are considered by the single-phase one-step relaxation terms, as in all Bhatnagher–Gross–Krook versions of the lattice Boltzmann model.

The interaction force between particles of component ϕ at site $\vec{x}$ and of component ϕ' at site $\vec{x}'$ is assumed to be proportional to their respective effective mass. The interaction is approximated in the form of an effective free-energy potential, $\psi^{\phi}(\rho^{\phi}(\vec{x}))$, which is in the Shan–Chen model written for phase ϕ at position $\vec{x}$ as a function of the local particle mass density. It takes the following switch-like empirical form

$$\psi^{\phi}(\vec{x}) = \psi^{\phi}(\rho^{\phi}(\vec{x})) = \rho_0^{\phi}\left(1 - \exp\left(-\frac{\rho^{\phi}(\vec{x})}{\rho_0^{\phi}}\right)\right), \tag{33}$$

which marks a sharp transition between the light and the dense phase. In this expression ρ_0^{ϕ} is a tunable constant in the form of a reference density which defines the transition between the light and the dense phase.

One should remark that the spacing between particles of component ϕ at site $\vec{x}$ and of component ϕ' at site $\vec{x}'$ takes in the Shan–Chen model only pairwise nearest-neighbour interactions into account, i.e. $|\vec{x} - \vec{x}'| = |\vec{e}_i|$. The total interaction force on component ϕ at site $\vec{x}$ can be written as

$$\begin{aligned} \vec{F}^{\phi}(\vec{x}) &= -\sum_{\phi'=1}^{k}\sum_{\vec{x}'} V_{\phi\phi'}(\vec{x},\vec{x}')(\vec{x}' - \vec{x}) = -\sum_{\phi'=1}^{k}\sum_{i=0}^{n} V_{\phi\phi'}(\vec{x},\vec{x}+\vec{e}_i)\vec{e}_i, \\ V_{\phi\phi'}(\vec{x},\vec{x}') &= G_{\phi\phi'}(\vec{x},\vec{x}')\psi^{\phi}(\vec{x})\psi^{\phi'}(\vec{x}'), \end{aligned} \tag{34}$$

where $V_{\phi\phi'}(\vec{x}, \vec{x}')$ is an interaction pseudo-potential between different phases. The summation symbol over $\vec{x}'$ accounts for nearest neighbour nodes. The symbol $G_{\phi\phi'}(\vec{x}, \vec{x}')$ is the strength of the interaction. It assumes the form of a Green's function matrix which satisfies the symmetry relationship $G_{\phi\phi'}(\vec{x}, \vec{x}') = G_{\phi\phi'}(\vec{x}', \vec{x})$. The expression for the pairwise phase interaction shows that the pseudo-potential force acting on the component phase ϕ at site $\vec{x}$ is simply a neighbour sum of the forces between the fluid particles belonging to phase ϕ at site $\vec{x}$ and the fluid particles belonging to phase ϕ' at the neighbouring sites $\vec{x}'$. If only homogeneous isotropic interactions between the nearest neighbours are considered, the Green's function $G_{\phi\phi'}(\vec{x}, \vec{x}')$ assumes the form of a simple symmetric lattice matrix with constant elements, i.e.

$$G_{\phi\phi'}(\vec{x}, \vec{x}') = \begin{cases} 0 & \text{if } |\vec{x} - \vec{x}'| > |\vec{e}_i|, \\ g_{\phi\phi'} & \text{if } |\vec{x} - \vec{x}'| = |\vec{e}_i|, \end{cases} \tag{35}$$

where $|\vec{e}_i|$ is the magnitude of the lattice parameter and $g_{\phi\phi'}$ is the amplitude factor of the strength of the interaction potential between components ϕ and ϕ'.

It is an important feature of the Green's function method that phase separation starts spontaneously once the interaction strength exceeds a critical threshold value. This means that, inversely, this critical interaction value acts like a phase transformation temperature [24, 30–32, 42–45]. Therefore, it can be used to calibrate the system with respect to the thermodynamic properties of the interacting flows under consideration of lattice type and initial density. The value of the amplitude factor $g_{\phi\phi'}$ can be related to the surface tension. The moments of these k flows can be calculated, as outlined in the preceding subsection.

Alternative formulations of the pseudo-potential method for multi-phase lattice flows use a related approach, where the lattice Boltzmann equation is not equipped with a separate source term. In these approaches the actual pair interaction between the immiscible phases enters through a modified form of the equilibrium distribution functions rather than explicitly through a separate source term. One can show, though, that these lattice formulations are equivalent.

1.9.4. Free-energy models of multi-phase lattice Boltzmann flows. The multi-phase lattice Boltzmann models outlined in the preceding two subsections are based on phenomenological sharp-interface approaches to interface energy and dynamics. Although, particularly, the model formulations suggested by Shan and Chen [42, 43] and Shan and Doolen [44, 45] are well suited for describing spontaneous phase separation including Laplace-type capillary effects in isothermal multi-component flows, an important improvement was suggested by Swift and co-workers [46–48] in terms of the free-energy approach.

The basic approach of the free-energy model is that the equilibrium distribution can be defined consistently, based on thermodynamics, using, for instance, a classical form of a diffuse interface approach. Consequently, the conservation of the total energy, including surface, kinetic and internal energy terms, can be properly satisfied. The van der Waals or, respectively, the Cahn–Hilliard formulation of quasilocal thermodynamics for a two-component fluid in equilibrium at a fixed temperature has a free-energy functional form which is assumed to depend on density and density gradients according to

$$\psi(\rho) = \int [\varphi(T, \rho) + W(\nabla\rho)] \, \mathrm{d}V, \tag{36}$$

where the term $\varphi(T, \rho)$ in the integral is the bulk free-energy density and the second term, $W(\nabla\rho)$, is the free-energy contribution from density gradients and is related to the surface tension. An important aspect of this approach is that the free-energy functional can be written in the form of a Landau potential which includes high-order gradient terms that act as a penalty

contribution with respect to interface curvature. When using a quadratic interface penalty approximation the non-local system pressure P is related to the free-energy density functional according to

$$P = \rho \frac{\mathrm{d}\psi(T,\rho)}{\mathrm{d}\rho} - \psi(T,\rho) = P_0 - k\rho\nabla^2\rho^2 - \frac{1}{2}k|\nabla\rho|^2. \tag{37}$$

The full Ginzburg–Landau-type pressure expression, which includes also off-diagonal terms (see derivations in [24, 46]), enters finally a modified form of the equilibrium particle velocity distribution function which accounts also for some weak non-local terms [46–48].

1.10. Thermal fluctuations and movable interfaces—lattice Boltzmann simulations of colloidal particle-fluid suspensions

Lattice Boltzmann automata are well-suited for the simulation of colloidal suspensions owing to their conceptual potential to tackle intricate boundary conditions and to incorporate fluctuation forces [24, 25]. In order to simulate particles suspended in fluids the lattice Boltzmann method must incorporate discretized solid particles that can move across the nodes of the stationery lattice as well as an approximate treatment of the interaction of those particles with the fluid.

The latter aspect can be treated in the framework of the fluctuation–dissipation approach. Thermal fluctuations on a mesoscopic scale can be introduced into the lattice Boltzmann framework by means of stochastic Brownian *hits*. These thermal *hits* can be included in the form of small random pulses each exerting an additional force term which may shift the positions of the suspended particles to any of the neighbouring nodes in a probabilistic fashion. While each individual force pulse may push the particle in a arbitrary direction, the overall directional and amplitude distribution of the pulses must reproduce a Gaussian form. The variance of the distribution is adjusted in such a way as to define the temperature of the system by means of the fluctuation–dissipation theorem [24, 25, 49, 51]. Similar approaches are well known from solutions of Langevin-type continuum-field differential equations which require the incorporation of stochastic terms that mimic small thermal hits on continuum objects.

The second open question in this context is the treatment of the solid obstacles suspended in the fluid. According to the work of Ladd and co-worker [25, 49] a solid boundary can be mapped onto the lattice and a corresponding set of boundary nodes, $\vec{r}_\mathrm{b}$, can be defined in the middle of links, whose interior points represent a suspended particle. A no-slip boundary condition on the moving particle requires the fluid velocity to have the same speed at the boundary nodes as the particle velocity $\vec{u}_\mathrm{b}$ which has a translational portion $\vec{U}$ and a rotational portion $\vec{L}$. Assuming that the centre position of the particle is $\vec{R}$, then

$$\vec{u}_\mathrm{b} = \vec{U} + \vec{L} \times (\vec{r}_\mathrm{b} - \vec{R}). \tag{38}$$

The distribution function f_i is then defined for grid points inside and outside the suspended particle. To account for the momentum change when $\vec{u}_\mathrm{b}$ is not zero, Ladd proposed to add a term to the distribution function for both sides of the boundary nodes:

$$f_i^*(\vec{x}) = f_i(\vec{x}) \pm B(\vec{e}_i \cdot \vec{u}_\mathrm{b}), \tag{39}$$

where B is a coefficient which depends on the detailed lattice structure and which is proportional to the mass density of the fluid. The + sign applies to boundary nodes at which the particle is moving toward the fluid and the − sign for moving away from the fluid.

1.11. The lattice Boltzmann method for reactive flows

Reactive flows are ubiquitous in materials science and engineering. Prominent examples occur in the fields of corrosion and tribology. For describing such reactive flows consisting

Table 5. Overview of some groups which make simple trial versions of lattice Boltzmann source codes or executables available.

University of Braunschweig, Germany (http://www.cab.bau.tu-bs.de/institut/mitarbeiter/drittmittel/freudiger/freudiger.htm)
University of Erlangen, Germany (http://www.lstm.uni-erlangen.de/lbm2001)
Alfred-Wegener-Institut, Germany (http://www.awi-bremerhaven.de/Modelling)
University of Tokyo, Japan (http://www.gaea.k.u-tokyo.ac.jp/∼niimura/niimura-QF)
University of Geneva, Switzerland (http://cui.unige.ch/spc/Cosmase)

of a number of miscible species in the framework of the lattice Boltzmann approach requires to introduce a set of distribution functions, f_i^ϕ, matching the various components ϕ. The corresponding lattice Boltzmann equations amount to

$$f_i^{\phi,\mathrm{new}}(\vec{x}+\vec{c}_i\Delta t, t+\Delta t) - f_i^{\phi,\mathrm{old}}(\vec{x},t) = -\frac{\Delta t}{\tau^\phi}(f_i^{\phi,\mathrm{old}}(\vec{x},t) - f_i^{\phi,\mathrm{eq}}(\vec{x},t)) + R_i^\phi, \tag{40}$$

where R_i^ϕ is the reactive term which must have the property to reproduce the correct rates of the density changes, $\dot{\rho}^\phi$, and of the energy changes, $\dot{q}^\phi$, for each of the reaction partners, i.e.

$$m^\phi \sum_{i=0}^{n} R_i^\phi = \dot{\rho}^\phi, \qquad m^\phi \sum_{i=0}^{n} R_i^\phi \vec{e}_i = 0, \qquad m^\phi \sum_{i=0}^{n} R_i^\phi \frac{\vec{e}_i}{2} = \dot{q}^\phi, \tag{41}$$

where m^ϕ is the mass of a fluid particle belonging to species ϕ. Additional boundary conditions are due to the conservation of mass, i.e.

$$\sum_{\phi=1}^{k} m^\phi \sum_{i=0}^{n} R_i^\phi = 0, \qquad \sum_{\phi=1}^{k} m^\phi \sum_{i=0}^{n} R_i^\phi \frac{\vec{e}_i}{2} = \dot{q}, \tag{42}$$

where $\dot{q}$ amounts to the total heat exchange in the reactive flow. The rates $\dot{\rho}^\phi$ and $\dot{q}^\phi$ are usually sensitive (exponential) functions of the temperature. While the basic constraints for R_i^ϕ are given above in terms of the conservation equations, the detailed coefficients of the reaction term approximation depend on the specific problem addressed [24].

1.12. Implementation, boundary conditions and initial value conditions for lattice Boltzmann simulations

1.12.1. Numerical aspects, implementation and parallelization. The main steps in a lattice Boltzmann algorithm are the definition of the boundary conditions, the initialization of start values for density and momentum, the calculation of the local equilibrium distribution with these given values, the propagation of the particle portions to the next neighbour (except for the distribution of the rest particles), collision and the calculation of the new density and momentum distribution. After this step the time increment is increased by one unit and the algorithm starts again with the calculation of the equilibrium distribution, table 5.

The lattice Boltzmann method is basically resource intensive when it comes to larger three-dimensional arrays. This means that running simulations on systems in excess of 100^3 nodes is not practical because of the lack of memory resources and long processing times. However, one should underline that the method has very low memory use and high processing speed when counted per lattice site, particularly when it comes to complicated boundary conditions. This makes it an ideal and efficient method for materials-related applications which are often characterized by rough interfaces and flow percolation problems.

Because of these limitations set by conventional single processor architectures and owing to the fact that the lattice Boltzmann method generally requires only near-field neighbour

information (like most cellular automata), the algorithm is a good candidate for parallel implementations. Parallelization is typically realized by multiple instruction multiple data (MIMD) systems which run in single program multiple data (SPMD) mode. This means that the data set is divided into spatially contiguous blocks along one axis. Multiple copies of the same program are then executed simultaneously on different processors belonging to the parallel computer, each operating on its own block of data (SPMD concept). Each copy of the program runs as an independent process, and typically each process runs on its own processor. At the end of each iteration, data for the lines (two-dimensional) or planes (three-dimensional) that lie on the boundaries between blocks are passed between the appropriate processes. This means that almost all parts of an algorithm must be carried out fully parallel in order to obtain maximum acceleration upon parallelization. The exchange of data between the processors must be provided within the code by using communication library routines such as the message passing interface (MPI) library or the parallel virtual machine (PVM) library.

1.12.2. Boundary conditions. The first step in the set-up of the boundary conditions for a lattice Boltzmann simulation consists in the definition of the character of each lattice node. Fluid nodes are those grid points on which the flow collision operator is fully applied. All other grid points are referred to as solid nodes. The relevant ones among them are the boundary nodes. These are the ones where flows impinge on at least one solid node which may belong to a movable particle or to the system wall. The node type can be identified by a Boolean marker.

Collisions of fluid particles with solid objects at the boundary nodes can be grouped into three types of obstacle situations [4, 24, 32]. These are collisions with static solid objects, e.g. static wall elements, collisions with moving walls, e.g. to shear the system, and collisions with moving particles, e.g. such as occurring for freely suspended colloids (see separate section). In each of these cases the additional possibility of wall reactivity can be taken into account by separate rules.

Such contact situations are in lattice Boltzmann simulations usually implemented by applying so-called no-slip, or stick, boundary conditions in the case when a solid obstacle imposes friction, figure 10. This is achieved by implementing a bounce-back algorithm on the links [4, 24, 32]: during propagation, the component of the distribution function that would propagate into the solid node is bounced back and ends up back at the fluid node, but pointing in the opposite direction. This means that incoming particle portions are reflected back towards the nodes they came from. This rule produces stick boundary conditions at roughly one-half the distance along the link vector joining the solid and fluid nodes, ensuring that the velocity of the fluid in contact with the solid equals the velocity of the latter. In the case when the zero-velocity plane must be located exactly inside the boundary layer, i.e. on the corresponding boundary layer nodes rather than being shifted from the location of the boundary nodes half-way into the fluid, one can use suited interpolation algorithms [24, 30–32]. An alternative to the introduction of a nodal bounce-back interpolation rule is to place the boundary nodes midway between solid and fluid nodes. Frictional slip or the limiting case of free-slip boundary conditions may be appropriate for smooth boundaries with small or negligible friction exerted on the flow.

The surface forces resulting from particle bounce-back are calculated from the momentum transfer at each boundary node and summed to give the force and torque on each obstacle object. In contrast to finite-difference and finite-element methods, where local surface normals are required to integrate the stresses over the obstacle surface, the bounce-back rule eliminates these complications by directly summing the surface forces.

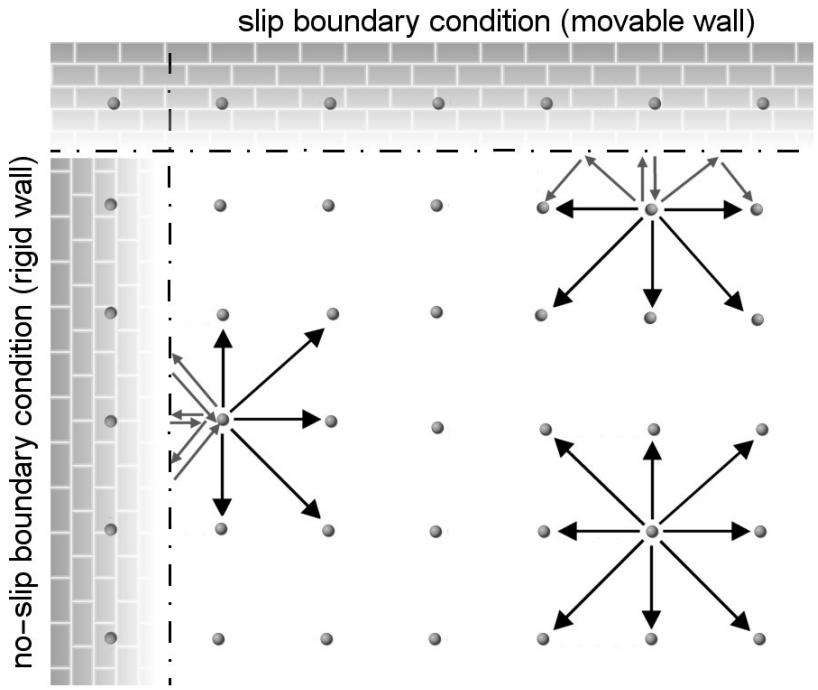

Figure 10. Schematic figure showing the implementation of the bounce-back algorithm both, for no-slip (rigid wall) as well as for slip boundary conditions (movable or deformable wall).

1.12.3. Initial conditions. Initial conditions can be defined by starting from an equilibrium distribution. This means that the flow density is equal to a constant everywhere on the grid, since $\rho(\vec{x}, t) = m \sum_{i=0}^{n} f_i^{\mathrm{eq}}(\vec{x}, t)$, and the speed is equal to 0 at each node in the system before the first translation and collision operations. The initiation of flows can than be induced by imposing constant velocity boundary conditions at the fluid inlet for instance in conjunction with periodic boundary conditions. Such settings can typically approximate the experimental practice of constant flow rates. Periodic boundary conditions are particularly useful for modelling bulk systems because they tend to minimize finite-size edge effects. Another important initial standard condition is the assumption of constant pressure.

1.13. Conventional cellular automata and the lattice Boltzmann method

The basic structure of the lattice Boltzmann method resembles that of a conventional cellular automaton algorithm, which has been successfully used particularly for the simulation of growth, recrystallization and coarsening phenomena in metals [50, 51]. The classical cellular automata typically used in materials science and engineering differ from both lattice gas and lattice Boltzmann methods, since they do not use fluid flow vectors or momentum vector distribution functions but scalar (e.g. energy) or tensorial (e.g. orientation) parameters as internal state variables.

Otherwise they follow a scheme common to all automata [52–55], that is, they are discrete in time and space and use Boolean or real-valued state variables to describe the constitutive behaviour of the materials at a microscopic level. They may be defined on different regular or non-regular two-dimensional or three-dimensional lattices considering the first, second or third neighbour shells for the calculation of the state change of a node. The system complexity

emerges from the repeated and synchronous application of certain cellular automaton rules equally to all nodes of the lattice. These local rules can for many cellular automaton and lattice Boltzmann models in materials science be derived through finite-difference formulations of the underlying differential equations that govern the system dynamics at a microscopic level. Important fields where materials-oriented cellular automata have been successfully used for microstructure predictions are primary static recrystallization and recovery [56–65] and solidification [66–70]. The automaton properties of the lattice Boltzmann method makes it an ideal platform for combinations with related materials simulation methods such as cellular automata for instance for the case of crystal growth [50, 51]. An overview on the relationship between the lattice Boltzmann method and conventional cellular automaton is given in [71].

2. Some applications of the lattice Boltzmann method in materials science and engineering

2.1. Introduction

The second part of the article provides an introduction to applications of the lattice Boltzmann method in the fields of materials science and engineering. Although the lattice Boltzmann method is considerably gaining momentum in the fields of general computational fluid mechanics, kinetic theory, chemical process engineering and soil mechanics, the materials engineering community has not yet fully exploited this approach.

Important topics which are of interest in the context of materials science and engineering are flow dynamic issues associated with tribology and friction during metal forming, fluid dynamics during melting, casting, semi-solid processing of metals and polymers including multi-component flows, hydrodynamic effects during liquid–liquid and liquid–solid phase transformations, flows in microporous microstructures such as those occurring during processing and infiltration of metallic foams or related composite pre-forms, colloidal flows, liquid crystal flows, lubricated contact mechanics, microdevice engineering, abrasion and crystal growth kinetics in conjunction with fluid flow.

All these examples have three points in common. First, they mark challenging topics in current materials science, engineering, and processing. Second, it is difficult to yield numerical convergence when simulating such situations with the aid of classical Navier–Stokes solvers, owing to the intricate boundary conditions and constitutive behaviour inherent to such flows. Third, these problems are typically too large in terms of their respective spatial dimensions and characteristic timescales, so that off-lattice pseudo-particle or molecular dynamics approaches cannot be used. This means that most of the materials-related problems mentioned above are excellent candidates for the application of the lattice Boltzmann method.

An important aspect that must be considered, though, before the use of a lattice-based fluid dynamics simulation method to an engineering problem is its validity regime with respect to the situation encountered, as already discussed in greater detail in the first part of this article. The two important criteria in this context are the ratio of the mean free particle path relative to the characteristic system length (e.g. obstacle spacing) as expressed by the Knudsen number and the occurring characteristic macroscopic flow speed regimes as quantified by the Mach number. As a rule of thumb the lattice Boltzmann scheme is particularly well suited for small Mach numbers (below 0.15) and small Knudsen numbers (below 0.2), figures 1 and 3.

Some of the engineering topics mentioned above will be discussed in the following. It must be noted, though, that the intention of this part of the work does not to present in-depth treatment of the various topics, but rather, to present some representative examples which document the huge potential of the lattice Boltzmann simulation technique in the field of advanced materials

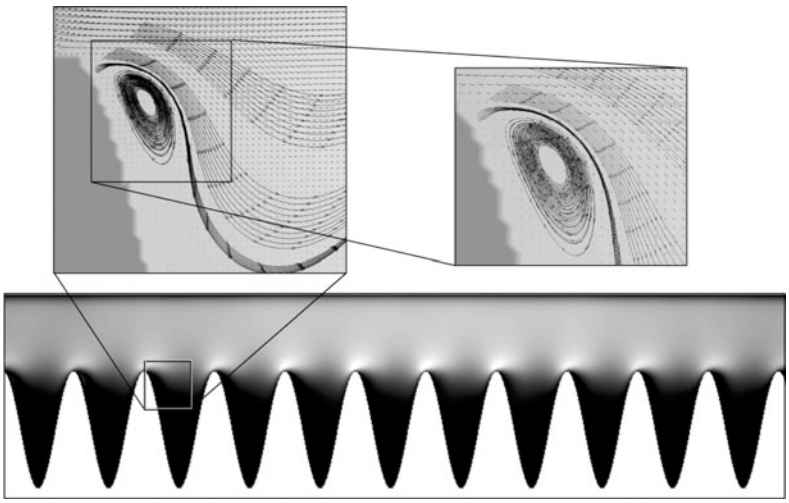

Figure 11. Application of the lattice Boltzmann method for the prediction of turbulence as a function of surface roughness at deformable metallic surfaces [50, 71]. The white area indicates the rough sheet surface. The greyscale pattern marks the fluid pressure where light tones corresponds to high and dark tones to low pressure. The arrows in the upper figures track some fluid portions visualizing turbulence.

science and engineering. The intention, therefore, is to stimulate the reader's interest in this method with respect to current and new problems in materials research. Further details which are beyond the limits of this article must, therefore, be obtained from the original references provided in each subsection.

2.2. Lubrication dynamics in metal forming

The precision which is nowadays required in the area of metal forming and tool design requires detailed knowledge of the underlying contact mechanics between workpiece, lubricant and tool. An essential example is the domain of large-scale automotive sheet forming where the overall shape accuracy after forming, including elastic springback, must be of the order of some hundred microns. Another example is the field of microdeformation processing such as used when forming metallic parts in the millimetre and centimetre range. Predicting the processing of such products is even more intricate when it comes to the treatment of contact micromechanics at a quantitative level. Related issues occur in the fields of sheet and foil rolling or for flows and corrosion in narrow tubes with rough surfaces.

A main aspect in the context of sheet forming—at least as far as fluid mechanics is concerned—is the importance of the surface roughness and the resulting (Prandtl-type) boundary layer flow dynamics in the vicinity of such a rough interface. Of particular interest are scaling effects in boundary layers which arise from changes in the surface topography of metals such as occurring during plastic forming. Scaling is important because metallic surfaces become rougher during deformation while the fluid properties may remain unchanged at least within certain bounds (temperature changes due to dissipated heat as well as abrasion are neglected at this point). An important observation is the transition from laminar to turbulent flow as a function of the roughness of the deformed metal surface.

Fluid dynamics for such a situation can be simulated by the use of a lattice Boltzmann automaton. For the example given in figure 11 the simulation strategy was designed to study the transition from laminar to turbulent flow as a function of the increasing roughness of a

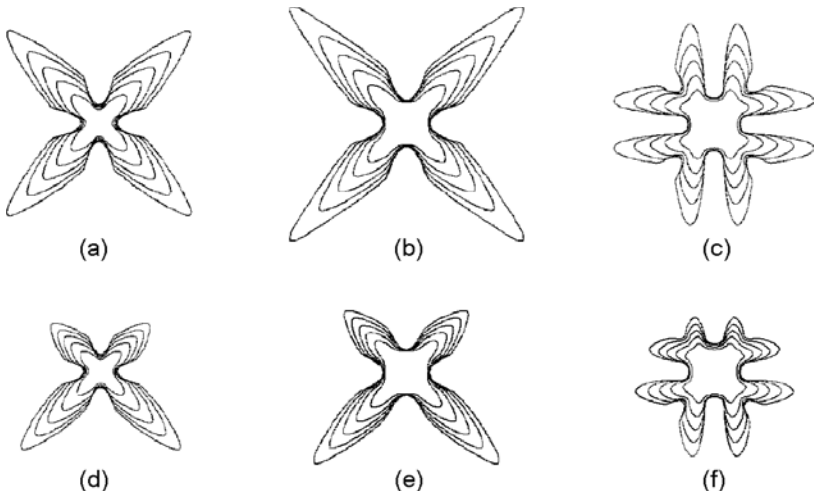

Figure 12. Simulation of Miller and Succi [74] of dendritic growth in a fluid environment. Figures (*a*), (*b*) and (*c*) show the evolution of crystal shapes for different seeds and tilting angles in the case of diffusive transport only. Figures (*d*), (*e*) and (*f*) show the evolution of the same crystal shapes for different seeds and tilting angles if buoyancy convection is present.

surface and of the viscosity of the fluid. The transition is characterized by the formation of turbulences in the vicinity of the tips of the roughness peaks. The rough metallic surface is modelled as a sinusoidal wall. One important parameter in the study is the variation of the period and amplitude of the sinusoidal surface. The flow is modelled by using a standard lattice Boltzmann automaton with single-step relaxation, figure 11.

2.3. Dendritic crystal growth under the influence of fluid convection

The group of Miller and co-workers [72–75] has recently designed a three-dimensional parallel lattice Boltzmann code for the simulation of liquid–solid phase transformations, in particular, for predicting dendritic growth in a flow environment, figure 12. The basic challenge of such an approach consists of providing a numerical tool for calculating growth kinetics together with fluid flow on a mesoscopic scale in one integrated simulation approach. Related pioneering work about the conjunction of forced flows with crystal growth was published by Tönhardt and Amberg [76] as well as by Beckermann *et al* [77]. The aim of such simulation studies is to study the influence of fluid convection on the crystal growth kinetics and on the resulting microstructures of the crystals.

The engineering perspective of such approaches is at hand. For instance, the growth of large single crystals with high quality for electronic and optical purposes is of huge industrial importance. Single crystals with certain lattice defects such as twins, small angle boundaries or related dislocation arrays resulting from growth do no longer have the same functional properties as a perfect single crystal. It is likely that a strong relationship exists between the flow dynamics on the one hand and the elementary atomic-scale and mesoscale solidification mechanisms on the other. It is, hence, of substantial importance to combine these two aspects in one theoretical framework, i.e. the development of joint simulation approaches may help to predict the optimal conditions for crystal growth experiments with respect to achieving crystals with good properties.

The approach of Miller and co-workers [72–75] is based on a joint phase-field lattice Boltzmann automaton concept. The two phases, liquid and solid, are, in this approach,

distinguished by a phase-field structure variable, similar as in the Ginzburg–Landau or Allen–Cahn models. In contrast to conventional continuum solid-state phase field models, where the time evolution of the phase field is computed by the integration of a Ginzburg–Landau-type differential equation, Miller *et al* describe the phase transition by a reaction model as originally suggested by de Fabritiis *et al* [78]. This model describes the phase transformation in terms of transition rates across the interface from one phase into the other and vice versa. The transition rates are calculated by using frequency factors from the inverse timescale for solidification and melting, respectively, together with switch functions which control the onset of melting and solidification around the critical temperature.

2.4. Simulation of metal foam processing

Metal foams are a novel class of energy absorbing structural materials with a considerable perspective for applications in the field of light-weight materials engineering. Their widespread commercial use, though, is still impeded by the unsatisfactory reproduction of material homogeneity. The microstructure and pore distribution of foams are to a large extent determined by the process parameters and by the details of the production strategy. Therefore, it is sensible to accompany further initiatives for microstructure and property optimization of metallic foams with systematic process simulations. These should be designed to identify process windows for optimum foam homogeneity and reproducibility of the cellular microstructure.

The simulation of the evolution and decay of metallic foams produced by powder metallurgical routes is a very demanding target for applications of the lattice Boltzmann method, since such processes are characterized by intricate boundary conditions, phase transformation, melt dynamics, gas dynamics and gas–melt interaction. Also, gravity occurs in such processes as a relevant body force.

The group of Singer [79] has recently published such an investigation on the formation of metal foams by using a lattice Boltzmann method. They had chosen a formulation with free surface boundary conditions which allowed them to incorporate the gas–liquid interface that is typical of the cellular foam microstructure evolving during processing. The study aimed in particular at the clarification of the relationship between the processing parameters and the resulting microstructures placing attention on pore nucleation, pore growth, pore coalescence and solidification. The simulations were used to better understand corresponding experiments which were conducted with an aluminium alloy which was mixed and subsequently processed with the TiH_2 as an agent providing the gas by a powder metallurgical processing route. The study provided basic insight into the influence of viscosity, surface tension, body forces and mould form on the kinetic and structural evolution of the foam microstructure.

Related fields where the lattice Boltzmann method has reached the necessary maturity as a simulation tool for optimizing production processes is the modelling of flows in complex and time-dependent geometries, as they are encountered in the context of composite materials that are manufactured by infiltrating fibre or powder pre-forms, figure 13.

2.5. Hydrodynamics of liquid crystalline polymers

Liquid crystalline polymers are in a state of matter in which liquid-like order exists at least in one direction of space and in which some degree of structural anisotropy is present. Typically such materials consist of rod- or plate-like molecular constituents which can align to a certain extent. One differentiates between two types of liquid crystalline polymers. Nematic ones

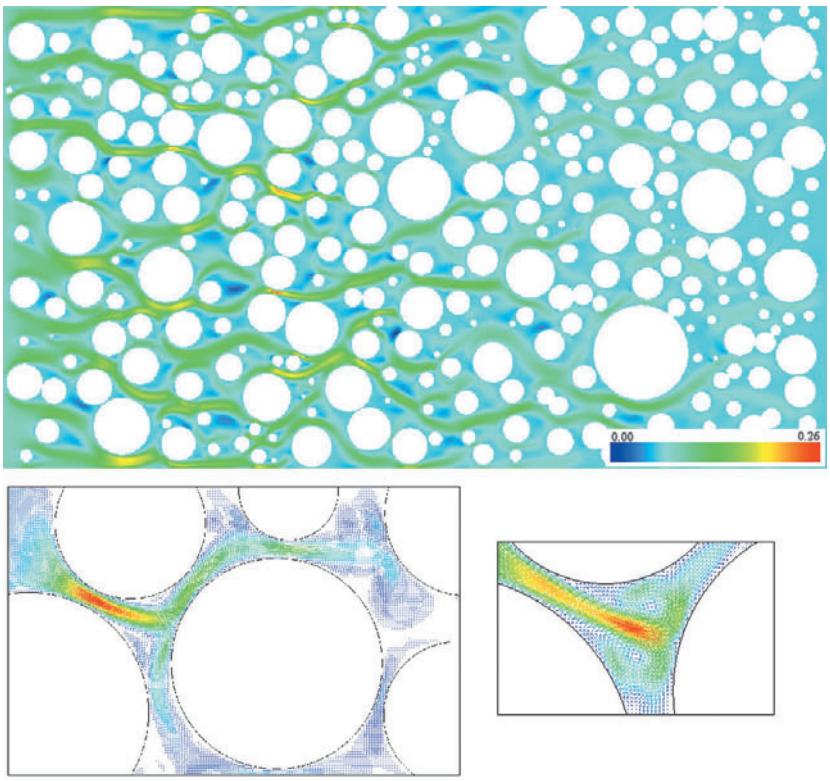

Figure 13. Simulation of flow infiltration through highly idealized porous microstructures. The upper figure shows the pressure distribution and the two lower figures show details of the flow vectors.

are composed of molecules with their long axes aligned along a specific direction while their centres of mass are distributed randomly in space. The smectic liquid crystal state has a higher degree of order than the nematic one due to existence of quasi-long range order in the positions of the centres of gravity of the molecules in one or two dimensions. This means that the nematic state is characterized by orientational order while the smectic one reveals both translational and orientational order. Since the state of structural order in these liquid crystals is between the traditional solid and liquid phases they are sometimes synonymously referred to as mesogenic materials. To quantify just how much order is present in a material, an order parameter can be defined which quantifies the angular deviation between the director and the long axis of each molecule. For a perfect crystal, the order parameter is one. Typical values for the order parameter of a liquid crystal range between 0.3 and 0.9, with the exact value a function of temperature, as a result of kinetic molecular motion. The alignment of the liquid crystal molecules entails tensorial anisotropy of the properties.

Studying the relationship between the structure of liquid crystalline materials and the underlying hydrodynamics is essential for understanding the properties of these materials. Other than conventional Newtonian flows, liquid crystals reveal a strong coupling between their microscopic structure and the velocity fields imposed by the flow. For instance, shear flow can induce non-equilibrium phase transition from the isotropic (fluid) to the nematic state or lead to phenomena such as shear-thinning and thickening.

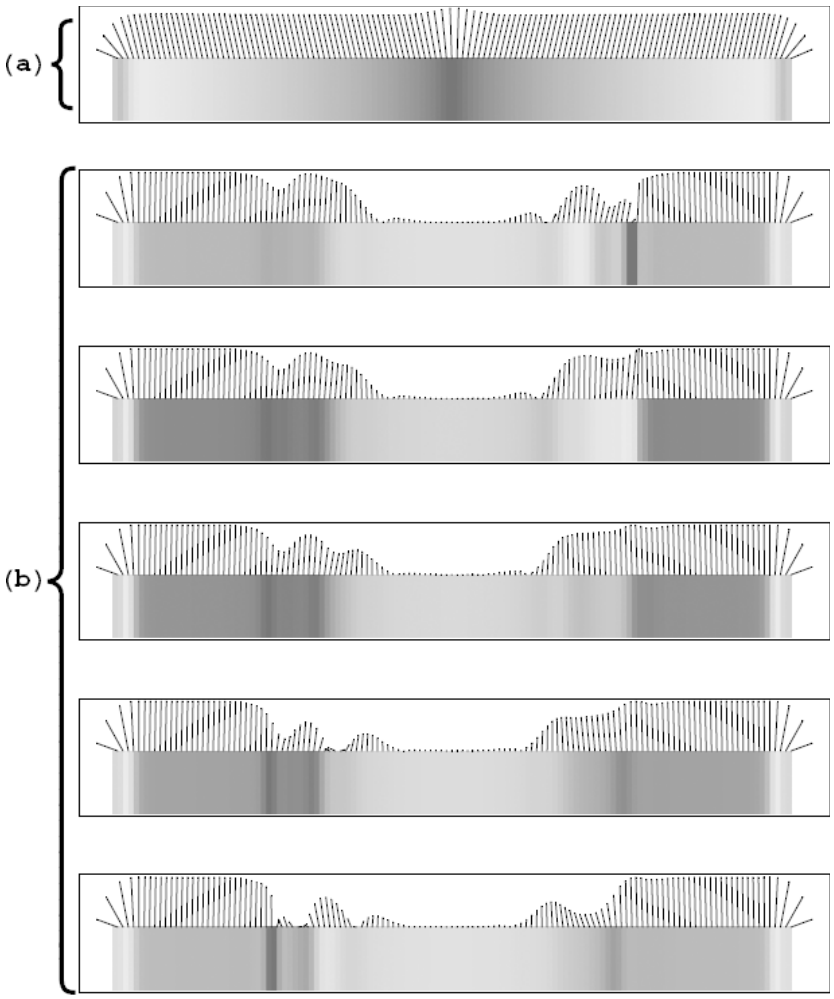

Figure 14. Lattice Boltzmann simulations of liquid crystal hydrodynamics. The coupling between the tensor order parameter and the flow is treated consistently, allowing investigation of a wide range of non-Newtonian flow behaviour [82]. The figure shows two different states in Poiseuille flow, where the lines represent the director orientation of the nematic phase projected down onto the x–y plane, and the shading represents the amplitude of the order parameter. Flow is from top to bottom, and the walls are at the left and right. At the walls, the director is aligned perpendicular to the boundary. (*a*) A stable configuration at low flow. (*b*) Snapshots of an oscillating configuration where the central region is in the 'log-rolling state' (director perpendicular to the plane) and the boundary region consists of a transition from a configuration in the shear plane to a 'tumbling' and 'kayaking' region (director rotating in and out of the plane) interfacing to the central log-rolling state.

In order to better understand the mechanics of such flows it is pertinent to use simulation methods which can take into account the different length and timescales that are relevant for the constitutive behaviour of such materials.

Edwards and co-workers [80, 81] and the group of Denniston and Yeomans [82–84] have suggested to use a multi-phase lattice Boltzmann free-energy method in conjunction with the Bhatnagher–Gross–Krook single-step approximation for solving the hydrodynamic equations of motion for nematic liquid crystals, figure 14. The main modification of their formalism when compared to conventional multi-phase lattice Boltzmann schemes is the introduction of

an additional symmetric traceless tensor-valued structural distribution function $\boldsymbol{S}$.

$$\begin{aligned} f_i^{\text{new}}(\vec{x}+\vec{c}_i\Delta t, t+\Delta t) - f_i^{\text{old}}(\vec{x}, t) &= -\frac{\Delta t}{\tau^f}(f_i^{\text{old}}(\vec{x}, t) - f_i^{\text{eq}}(\vec{x}, t)), \\ \boldsymbol{S}_i^{\text{new}}(\vec{x}+\vec{c}_i\Delta t, t+\Delta t) - \boldsymbol{S}_i^{\text{old}}(\vec{x}, t) &= -\frac{\Delta t}{\tau^S}(\boldsymbol{S}_i^{\text{old}}(\vec{x}, t) - \boldsymbol{S}_i^{\text{eq}}(\vec{x}, t)), \end{aligned} \tag{43}$$

where $\boldsymbol{S}_i^{\text{eq}}$ is the corresponding equilibrium distribution. The values of this tensor density variable are related to a tensorial order parameter which can describe the crystalline anisotropy of those volume portions that assume the nematic or smectic state. Backflow, the hydrodynamics of topological defects, and the possibility of transitions between the liquid crystalline and isotropic phases appear naturally within the formalism. The method can also be used to study the velocity dependence of the critical temperature in the nematic-isotropic transition.

2.6. Flow percolation in confined geometries

Figure 15 shows a two-dimensional example (taking a perspective into the plane in which the fluid flows) where a Boltzmann-based lattice simulation has been applied to the situation of a confined lubrication flow. The roughness data were experimentally obtained from a plastically deformed steel surface. The experimental roughness analysis conducted on the surface of the sample allowed us to separate the free volume portions from the regions where the sample was in closed contact with the tool. The percolative flow in the remaining confined interface represents a typical example of flow in a porous environment. The joint experiments and fluid dynamics simulations aim at a better understanding of tribology and contact mechanics during steel sheet rolling. Parameters to be varied in the experiments and simulations are the viscosity of the lubricants, the surface roughness of the tool and sheet materials, and the deformation rate.

The upper left-hand side of figure 15 shows the pressure distribution in the lubricant when compressed and redistributed in the experimentally determined obstacle (contact) field. The data show an in-plane view into the contact layer. The other two figures show magnifications of flow details.

2.7. Processing of polymer blends—breakup and coalescence of fluid droplets

Another important area for applying the lattice Boltzmann method to materials engineering is the field of polymer processing, particularly the mixing of immiscible polymers. Because most chemically different polymers are relatively immiscible, fluid blending of such materials is an ubiquitous challenge in the field of industrial polymer processing.

The two most important classes of engineering polymers produced by blending are rubber-toughened plastics and stiffened elastomers. Such composites are characterized by synergetic mechanical properties which arise from the two immiscible polymer compounds in them. Blending processes for polymers are typically designed to produce fine spherical inclusions for an increase in impact resistance or fibre-type dispersoids for enhanced unidirectional strength. Typical products made of such materials are structural macroscopic parts. Examples are blends of nylon and rubber and rubber-toughening of brittle glassy polymers where the rubber inclusions can stop propagating cracks through the brittle material and dissipate energy.

For these reasons associated with mechanical properties and microstructure homogeneity the aim of the blending process is to produce very fine dispersions of the order of submicron-size droplets. Optimization of the process window for achieving these goals requires an improved understanding of the dynamics and mechanisms associated with polymer droplet breakup during mixing of such immiscible polymers. Important subtasks in this context are

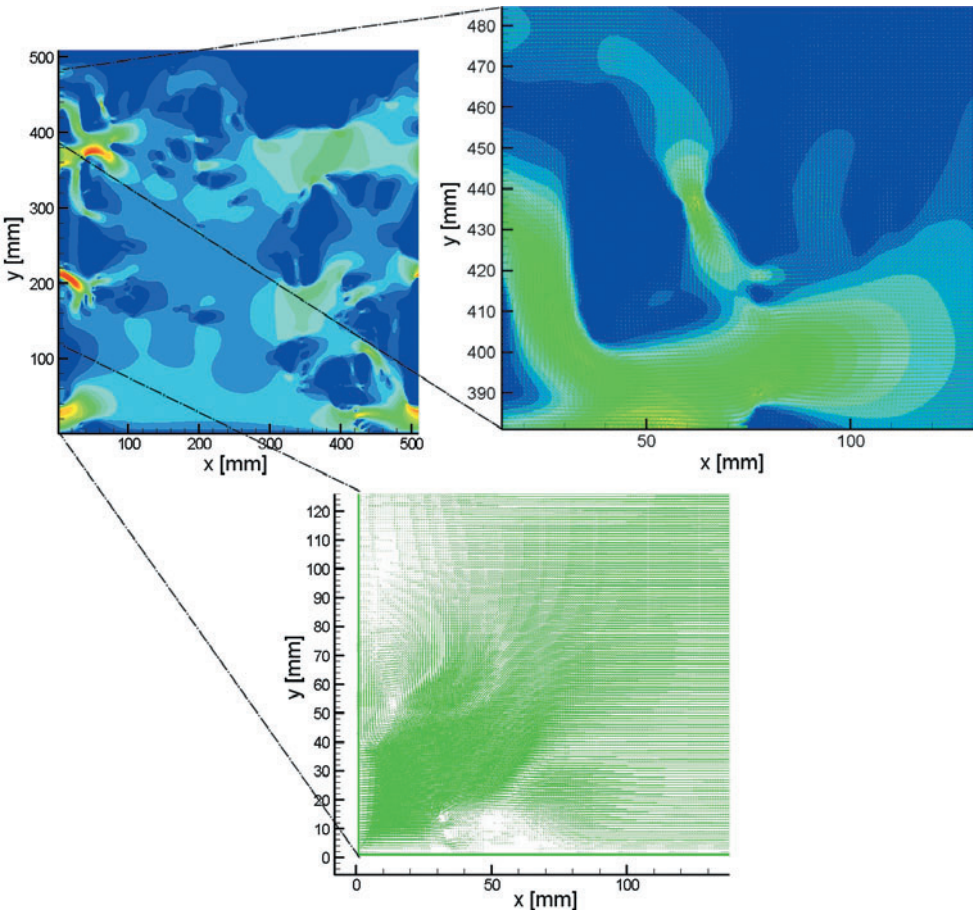

Figure 15. A two-dimensional example of the application of a Boltzmann lattice gas simulation to measured the roughness data of plastically deformed steel. The upper left figure shows the pressure distribution in the lubricant when compressed and redistributed in the experimentally determined obstacle (contact) field. The data show an in-plane view into the contact layer. The other two figures show magnifications of flow details [71].

the simulation of the microscopic breakup processes of droplets in shear flow and added block copolymers, the prediction of the droplet size distribution in heavily sheared multi-phase flows, the coalescence of polymer droplets in blends during processing, as well as the detailed analysis of such microscopic processes under realistic industrial boundary conditions in terms of geometry, processing rates, pressure and shear rates. The lattice Boltzmann method represents an excellent approach to the numerical analysis of such multi-phase polymer processing operations, particularly when aiming at the simulation of the breakup of droplets under shear. The approach is very effective in systems which involve low Reynolds numbers, different phases and complex geometrical boundary conditions [85, 86].

3. Conclusions

The study gave an overview of the lattice Boltzmann simulation method as an advanced tool for predictions in the field of materials science and engineering. The first part presented the basic

structure of the lattice Boltzmann simulation method while the second part gave examples from materials science and engineering. The main conclusions are:

- The lattice Boltzmann simulation method is superior to conventional Navier–Stokes solvers in the fields of materials science and engineering owing to its excellent numerical stability and constitutive versatility.
- The lattice Boltzmann approach is an ideal platform for scale-bridging simulations since the constitutive data, which finally characterize the flow at a macroscopic level, are incorporated at a microscopic scale. This allows one to describe the behaviour of flows (relaxation, collisions with another fluid or solid phase) at an elementary and physically based level.
- The basic structure of the lattice Boltzmann method resembles that of a cellular automaton where the main constitutive internal state variable is a real-valued momentum vector distribution function. This makes it an ideal platform for combinations with related simulation methods such as cellular automata or Potts-type models for crystal growth.

References

[1] Prandtl L, Tietjens O G and Hartjog J 1934 *Applied Hydro and Aeromechanics* (London: McGraw-Hill)
[2] Lamb H 1945 *Hydrodynamics* (New York: Dover)
[3] Landau L D and Lifshitz E M 1959 *Fluid Mechanics* (Oxford: Pergamon)
[4] Rothman D H and Zaleski S 1997 *Lattice Gas Cellular Automata* (Cambridge: Cambridge University Press)
[5] Rivet J P and Boom J P 2001 *Lattice Gas Hydrodynamics* (*Cambridge Nonlinear Science Series 11*) (Cambridge: Cambridge University Press)
[6] Hoogerbrugge P J and Koelman J M V A 1992 *Europhys. Lett.* **19** 155
[7] Koelman J M V A and Hoogerbrugge P J 1993 *Europhys. Lett.* **21** 369
[8] Espanol P and Warren P 1995 *Europhys. Lett.* **30** 191
[9] Schlijper A G, Hoogerbrugge P J and Manke C W 1995 *J. Rheol.* **39** 567
[10] Kong Y, Manke C W, Madden W G and Schlijper A G 1997 *J. Chem. Phys.* **107** 1
[11] Groot R D and Warren P 1997 *J. Chem. Phys.* **107** 4423
[12] Bird G A 1994 *Molecular Gas Dynamics and the Direct Simulation of Gas Flows* (Oxford: Clarendon)
[13] Oran E S, Oh C K and Cybyk B Z 1998 *Annu. Rev. Fluid Mech.* **30** 403
[14] Alexander F, Garcia A and Alder B 1994 *Phys. Fluids* **6** 3854
[15] Alexander F and Garcia A 1997 *Comput. Phys.* **11** 588
[16] Muntz E P 1989 *Ann. Rev. Fluid Mech.* **21** 387
[17] Garcia A, Bell J, Crutchfield Y and Alder B 1999 *J. Comp. Phys.* **154** 134
[18] Hardy J, Pomeau Y and de Pazzis O 1973 *J. Math. Phys.* **14** 1746
[19] Frisch U, Hasslacher B and Pomeau Y 1986 *Phys. Rev. Lett.* **56** 1505
[20] Bird G A 1976 *Molecular Gas Dynamics* (Oxford: Clarendon)
[21] Bird G A 1994 *Molecular Gas Dynamics and the Direct Simulation of Gas Flows* (Oxford: Clarendon)
[22] Garcia A L 1994 *Numerical Methods for Physics* (Englewood Cliffs, NJ: Prentice-Hall) chapter 10
[23] Chapman S and Cowling T G 1970 *The Mathematical Theory of Non-Uniform Gases* (Cambridge: Cambridge University Press)
[24] Succi S 2001 *The Lattice Boltzmann Equation: for Fluid Dynamics and Beyond* (*Series Numerical Mathematics and Scientific Computation*) (Oxford: Oxford University Press)
[25] Ladd A J C and Verberg R 2001 *J. Stat. Phys.* **104** 1191
[26] McNamara G and Zanetti G 1988 *Phys. Rev. Lett.* **61** 2332
[27] Higuera F J and Jimenez J 1989 *Europhys. Lett.* **9** 663
[28] Benzi R, Succi S and Vergassola M 1992 *Phys. Rep.* **3** 145
[29] Chen S, Shan X, Wang Z and Doolen G D 1992 *J. Stat. Phys.* **68** 379
[30] Chen S and Doolen G D 1998 *Ann. Rev. Fluid Mech.* **30** 329
[31] Chopard B and Droz M 1998 *Cellular Automata Modeling of Physical Systems* (Cambridge: Cambridge University Press)
[32] Wolf-Gladrow D A 2000 *Lattice-Gas Cellular Automata and Lattice Boltzmann Models* (*Lecture Notes in Mathematics, No. 1725*) (Berlin: Springer)
[33] Rothman D H and Keller J M 1988 *J. Stat. Phys.* **52** 1119

[34] Bhatnagar P, Gross E P and Krook M K 1954 *Phys. Rev.* **94** 511
[35] d'Humières D, Lallemand P and Frisch U 1986 *Europhys. Lett.* **2** 291
[36] Frisch U, d'Humières D, Hasslacher B, Lallemand P, Pomeau Y and Rivet J-P 1987 *Complex Sys.* **1** 649
[37] d'Humières D, Lallemand P and Searby G 1987 *Complex Syst.* **1** 633
[38] Qian Y, d'Humières D and Lallemand P 1992 *Europhys. Lett.* **17** 479
[39] Gunstensen A K, Rothman D H, Zaleski S and Zanetti G 1991 *Phys. Rev.* A **43** 4320
[40] Gunstensen A K and Rothman D H 1992 *Europhys. Lett.* **18** 157
[41] Grunau D, Chen S and Eggert K 1993 *Phys. Fluids* A **5** 2557
[42] Shan X and Chen H 1993 *Phys. Rev.* E **47** 1815
[43] Shan X and Chen H 1994 *Phys. Rev.* E **49** 2941
[44] Shan X and Doolen G D 1995 *J. Stat. Phys.* **49** 2941
[45] Shan X and Doolen G D 1996 *Phys. Rev.* E **54** 3614
[46] Swift M R, Osborn W R and Yeomans J M 1995 *Phys. Rev. Lett.* **75** 830
[47] Orlandini E, Swift M R and Yeomans J M 1995 *Europhys. Lett.* **32** 463
[48] Swift M R, Orlandini E, Osborn W R and Yeomans J M 1996 *Phys. Rev.* E **54** 504
[49] Ladd A J C 1994 *J. Fluid Mech.* **271** 285
Ladd A J C 1994 *J. Fluid Mech.* **271** 311
[50] Raabe D 2002 *Ann. Rev. Mater. Res.* **32** 53
[51] Raabe D 1998 *Computational Materials Science* (Weinheim: Wiley-VCH)
[52] von Neumann J 1963 The general and logical theory of automata *Papers of John von Neumann on Computing and Computer Theory* (*Charles Babbage Institute Reprint Series for the History of Computing* vol 12) ed W Aspray and A Burks (Cambridge, MA: MIT Press, 1987)
[53] Wolfram S 1986 *Theory and Applications of Cellular Automata* (*Advanced Series on Complex Systems, selected papers 1983–1986* vol 1) (Singapore: World Scientific)
[54] Minsky M 1967 *Computation: Finite and Infinite Machines* (Englewood Cliffs, NJ: Prentice-Hall)
[55] Conway J H 1971 *Regular Algebra and Finite Machines* (London: Chapman and Hall)
[56] Hesselbarth H W and Göbel I R 1991 *Acta Metall.* **39** 2135
[57] Pezzee C E and Dunand D C 1994 *Acta Metall.* **42** 1509
[58] Sheldon R K and Dunand D C 1996 *Acta Mater.* **44** 4571
[59] Davies C H J 1995 *Scripta Metall.* **33** 1139
[60] Marx V, Raabe D, Engler O and Gottstein G 1997 *Textures Microstruct.* **28** 211
[61] Marx V, Reher F R and Gottstein G 1998 *Acta Mater.* **47** 1219
[62] Davies C H J 1997 *Scripta Mater.* **36** 35
[63] Raabe D 1999 *Phil. Mag.* A **79** 2339
[64] Raabe D and Becker R 2000 *Modelling Simul. Mater. Sci. Eng.* **8** 445
[65] Raabe D 2000 *Comp. Mater. Sci.* **19** 13
[66] Cortie M B 1993 *Metall. Trans.* B **24** 1045
[67] Brown S G R, Williams T and Spittle J A 1994 *Acta Metall.* **42** 2893
[68] Gandin C A and Rappaz M 1997 *Acta Metall.* **45** 2187
[69] Gandin C A 2001 *Adv. Eng. Mater.* **3** 303
[70] Brown S G R, Clarke G P and Brooks A J 1995 *Mater. Sci. Technol.* **11** 370
[71] Raabe D, Roters F, Barlat F and Chen L-Q (ed) 2004 *Continuum Scale Simulation of Engineering Materials—Fundamentals—Microstructures—Process Applications* (Weinheim: Wiley-VCH)
[72] Miller W and Schröder W 2001 *J. Cryst. Growth* **230** 1
[73] Miller W 2001 *J. Cryst. Growth* **230** 263
[74] Miller W and Succi S 2002 *J. Stat. Phys.* **107** 173
[75] Miller W, Succi S and Mansutti D 2001 *Phys. Rev. Lett.* **86** 3578
[76] Tönhardt R and Amberg G 1998 *J. Cryst. Growth* **194** 406
[77] Beckermann C, Diepers H-J, Steinbach I, Karma A and Tong X 1999 *J. Comput. Phys.* **154** 468
[78] de Fabritiis G, Mancini A, Mansutti D and Succi S 1998 *Int. J. Mod. Phys.* C **9** 1405
[79] Körner C, Thies M and Singer R F 2002 *Adv. Eng. Mater.* **4** 765
[80] Beris A N and Edwards B J 1994 *Thermodynamics of Flowing Systems* (Oxford: Oxford University Press)
[81] Doi M and Edwards S 1989 *The Theory of Polymer Dynamics* (Oxford: Clarendon)
[82] Denniston C, Orlandini E and Yeomans J M 2001 *Phys. Rev.* E **63** 056702
[83] Denniston C 1996 *Phys. Rev.* B **54** 6272
[84] Denniston C, Orlandini E and Yeomans J M 2000 *Europhys. Lett.* **52** 481
[85] Wagner A J and Yeomans J M 1997 *Int. J. Mod. Phys.* C **8** 773
[86] Wagner A J and Yeomans J M 1999 *Phys. Rev.* E **59** 4366