Corrected momentum exchange method for lattice Boltzmann simulations of suspension flow

Eric Lorenz,* Alfonso Caiazzo,† and Alfons G. Hoekstra‡
University of Amsterdam, Kruislaan 403, 1098 SJ Amsterdam, The Netherlands
(Received 15 August 2008; published 26 March 2009)

Standard methods for lattice Boltzmann simulations of suspended particles, based on the momentum exchange algorithm, might lack accuracy or violate Galilean invariance in some particular situations. Aiming at simulations of dense suspensions in high-shear flows, we motivate and investigate necessary correction terms. We propose an approach which, combining accurate treatments of fluid-structure interaction and moving boundaries, is able to preserve Galilean invariance in relevant orders and to improve the physical behavior of the system. We validate the approach in a comparison with standard methods in simple test problems.

DOI: 10.1103/PhysRevE.79.036705 PACS number(s): 02.70.−c, 47.57.E−, 47.11.Qr

I. INTRODUCTION

The lattice Boltzmann method (LBM) is by now a well established approach to simulate incompressible Navier-Stokes equations [1]. Since the very beginning, its possibility and flexibility for different modeling setups and applications have been widely explored.

We focus on simulation of dense liquid-particle suspensions. The main approaches used for this class of problems are the method by Ladd [2–4] and the ALD method (from Aidun, Lu, Ding [5–7]) both based on the momentum exchange algorithm (MEA) [2]. The two techniques offer different treatments of the fluid-solid interaction, including explicit lubrication forces.

We are interested in the rheology of sheared suspensions, i.e., the change in viscosity when the system is subject to an external shear flow. In such situations, it becomes useful to demand Galilean invariance of the computation of fluid-solid interactions to ensure homogeneous behavior of the particles, not dependent on the external velocity and therefore not dependent on the position in a sheared system (see Fig. 1).

An important tool in simulating sheared suspensions is the Lees-Edwards boundary condition (LEbc) [8], commonly used to maintain a constant shear rate over a finite domain. Briefly, the LEbc is implemented as a periodic boundary condition in the directions perpendicular to a given shear velocity gradient, where, at the same time, the periodic copies of the system are moving with respect to each other. From the computational point of view, suspended particles crossing the boundary are simulated in different reference systems.

It is therefore important to set up a method which reduces the effect of numerical non-Galilean invariance. Without a careful treatment of the fluid-particle interaction, this can be amplified, resulting in relatively large errors in the dynamics of the particles, and therefore that of the suspensions. In this paper we investigate in detail the effect of non Galilean invariance of the LBM methods on the dynamics of sheared particle-fluid systems. Selecting particular cases, we show that standard methods for suspension simulations might lack sufficient accuracy. Using asymptotic expansion techniques, we motivate the need for a correction term and implement a possible solution. The obtained corrected momentum exchange for suspension (CMES) preserves local consistency and Galilean invariance in relevant orders.

In Sec. II we introduce the LBM, the standard approaches for suspension flows, and a short description of Lees-Edwards boundary conditions. Some preliminary numerical tests motivate the need of a corrected algorithm, constructed in Sec. III using the asymptotic expansion technique. Comparisons with existing approaches and further numerical results are discussed in Sec. III A. In Sec. IV we draw the conclusions.

II. LATTICE BOLTZMANN METHOD

The lattice Boltzmann method [9,1] is an alternative approach to approximate solutions of the incompressible Navier-Stokes equations. It can be derived starting from a finite velocity model Boltzmann equation discretized on a regular lattice.

The general iteration of the algorithm reads

$$f_i(n+1,\mathbf{j}+\mathbf{c}_i) = f_i(n,\mathbf{j}) + A(f_i^{\mathrm{eq}}(f) - f_i)(n,\mathbf{j}), \qquad (1)$$

where $\mathbf{c}_i$, $i=0,\ldots,b-1$ is a finite set of b discrete velocities and f_i is the numerical solution for the density of particles moving in direction $\mathbf{c}_i$. Focusing on a D2Q9 model (nine

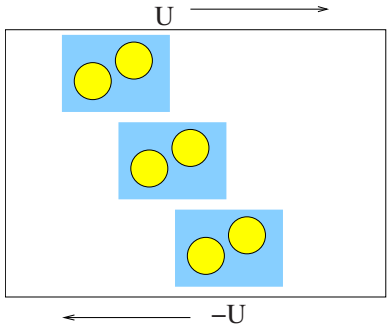

FIG. 1. (Color online) Simulating sheared suspensions particles are immersed in a varying velocity field. Galilean invariance plays a relevant role: dynamical properties of the system must be maintained independent from the flow velocity or gradient.

*E.Lorenz@uva.nl
†A.Caiazzo@uva.nl
‡A.G.Hoekstra@uva.nl

 ©2009 The American Physical Society

discrete velocities in two dimensions), we have

$$\mathbf{c}_1=(1,0),\quad \mathbf{c}_2=(0,1),\quad \mathbf{c}_3=(-1,0),\quad \mathbf{c}_4=(0,-1),$$

$$\mathbf{c}_5=(1,1),\quad \mathbf{c}_6=(-1,1),\quad \mathbf{c}_7=(-1,-1),\quad \mathbf{c}_8=(1,-1),$$

and $\mathbf{c}_0=(0,0)$ for resting fluid densities.

The matrix A on the right hand side of Eq. (1) specifies the collision operator. We employ a *two-relaxation-times* (TRT) model [10] where

$$A[f_i^{\mathrm{eq}}(f)-f_i]=\frac{1}{\tau}[f_i^{\mathrm{eq}}(f)-f_i]^+ + \frac{1}{\tau_-}[f_i^{\mathrm{eq}}(f)-f_i]^-. \quad (2)$$

F_i^+ and F_i^- denote the even and (respectively) the odd part of a function on the velocity space:

$$F_i^+=\frac{F_i+F_{i^*}}{2},\quad F_i^-=\frac{F_i-F_{i^*}}{2},$$

where i^* is such that $\mathbf{c}_{i^*}=-\mathbf{c}_i$.

We consider Eq. (1) written in a *dimensionless lattice units reference system*. In particular, sites in the computational domain are indexed by integers $\mathbf{j}\in\mathbf{Z}$, representing the spatial lattice, and $n\in\mathbf{N}$, counting the time steps. Space and time units are represented by grid size and time step. We remark that in problem relevant units time step Δt and space step h satisfy

$$\Delta t=h^2.$$

The last relationship, called *diffusive scaling* in the literature (more generally, $\frac{\Delta t}{h^2}=$const), is a prerequisite to recover the incompressible Navier-Stokes equations in the limit $h\to 0$ [11].

The density ρ and the velocity $\mathbf{u}$ are obtained by a calculation of the zeroth and first velocity moments of the distribution f:

$$\rho(n,\mathbf{j})=\sum_i f_i(n,\mathbf{j}),$$

$$\mathbf{u}(n,\mathbf{j})=\sum_i \mathbf{c}_i f_i(n,\mathbf{j}). \quad (3)$$

Conservation of mass and momentum during collision are constrained by the form of the equilibrium distribution f^{eq}, which depends on f through its density and velocity:

$$f_i^{\mathrm{eq}}(f)=E_i(\rho,\mathbf{u})=w_i\left(\rho+\frac{1}{c_s^2}\mathbf{c}_i\cdot\mathbf{u}+\frac{1}{2c_s^4}(\mathbf{c}_i\cdot\mathbf{u})^2-\frac{1}{2c_2^2}u^2\right). \quad (4)$$

In the implementation the mass units of measure are renormalized in order to have a unitary reference density. The parameters w_i and c_s appearing in the previous relations depend on the particular realization of the LBM scheme. The relaxation time τ is related to the fluid viscosity ν via

$$\tau=\frac{1}{2}+c_s^{-2}\nu. \quad (5)$$

The second relaxation parameter τ_- is set to 1 (This choice is due to stability issues. Dealing with dense suspension flows at relative high Reynolds numbers, setting $\tau_-=1$ effectively damps checkerboard effects and numerical pressure waves therefore improving the robustness. A comparison among different choices of τ_-, e.g., to control the position of the numerical boundaries [12], is under investigation.)

In practice, the implementation of Eq. (1) is split into a local *collision step* (2) and a *propagation step*:

$$f_i(n+1,\mathbf{j})=f_i(n_+,\mathbf{j}-\mathbf{c}_i). \quad (6)$$

Asymptotic analysis. Using the asymptotic expansion technique [11], it can be shown that the numerical solution of Eq. (1) can be approximated by the expansion

$$F_h(n,\mathbf{j})=f^{(0)}+hf^{(1)}(nh^2,h\mathbf{j})+h^2f^{(2)}(nh^2,h\mathbf{j})+O(h^3) \quad (7)$$

with smooth and h-independent coefficients defined as

$$f_i^{(0)}=w_i,$$

$$f_i^{(1)}=w_ic_s^{-2}\mathbf{c}_i\cdot\mathbf{u}_{NS},$$

$$f_i^{(2)}=w_ic_s^{-2}p_{NS}+\frac{w_ic_s^{-4}}{2}(|\mathbf{c}_i\cdot\mathbf{u}_{NS}|^2-c_s^2\mathbf{u}_{NS}^2)\tau w_ic_s^{-2}\mathbf{c}_i\cdot\nabla\mathbf{u}_{NS}\cdot\mathbf{c}_i, \quad (8)$$

where $\mathbf{u}_{NS}$ and p_{NS} solve a Navier-Stokes problem.

From Eq. (8), we conclude that Eq. (3) yields a second order accurate velocity, while a first order accurate pressure can be obtained with

$$p=c_s^2\frac{\rho-1}{h^2}. \quad (9)$$

Similarly, a first order approximation of the viscous stress tensor $\mathbf{S}[\mathbf{u}]=\nu(\nabla\mathbf{u}+\nabla\mathbf{u}^T)$ can be extracted using

$$S_{\alpha\beta}=-\frac{\nu}{h^2c_s^2\tau}\sum_i w_i[f_i-f_i^{\mathrm{eq}}(f)]\mathbf{c}_{i\alpha}\mathbf{c}_{i\beta}, \quad (10)$$

i.e., the second order moment of the *nonequilibrium part* with respect to the velocity space.

The results of the analysis actually provide more than these accuracy results. They contain relevant information concerning the structure of the solution, which can be used to improve the algorithm.

A. LBM for moving particles

Two-dimensional particles are represented by moving disks,

$$P_m(t)=\{\mathbf{x}\in\Omega|\|\mathbf{x}-\mathbf{x}_m(t)\|<R_m\}, \quad (11)$$

with centers $\mathbf{x}_m$ and radii R_m, for $m=1,\ldots,N_{\mathrm{susp}}$. The lattice nodes belonging to a disk $P_m(t_n)$ at a certain time step are marked as solid (Fig. 2).

Besides Eq. (1), boundary condition algorithms are needed to update the density distribution at the *boundary links*, i.e., the links connecting fluid and solid domains. The most popular choice is the *bounce-back* (BB) rule, which in the standard formulation reverts a distribution at a node $\mathbf{j}$ if

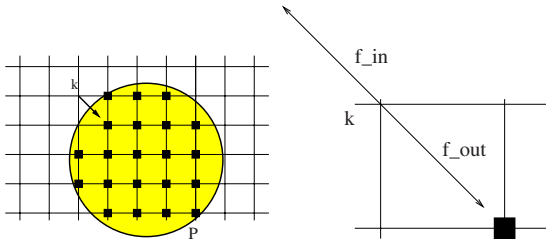

FIG. 2. (Color online) Sketch of computational fluid and solid domains. Boundary conditions (12) are applied at the *boundary links*, intersecting the solid-fluid interface, and the momentum exchange algorithm (13) is used to evaluate the fluid-particle interaction.

the associated link $\mathbf{c}_i$ intersects a particle surface; see Fig. 2. This leads to a computational boundary lying at the middle of the fluid-solid link. More accurate boundary conditions which take the exact intersection on the link into account [13] are also available. These are based on extrapolation rules for the missing populations, and might increase the complexity of the implementation, particularly in the case of high particle density.

Therefore, and for simplicity in demonstrating the improvement of our corrected method, we restrict to BB with its extension to moving boundaries. In practice it can be implemented [2] redefining the propagation step (6) as

$$f_i(n+1,\mathbf{j}) = f_{i*}(n_+,\mathbf{j}) - 2hc_s^{-2}w_i\mathbf{c}_i\cdot\mathbf{u_b} \tag{12}$$

if $\mathbf{c}_i$ is a boundary link. The index i^* is such that $\mathbf{c}_{i*}=-\mathbf{c}_i$ and $\mathbf{u_b}$ is the velocity of the point on the particle where the link $\mathbf{c}_i$ intersects the surface.

Regarding hydrodynamical properties of suspended particles, it has been shown that the use of the bounce-back rule for boundaries not orientated along the lattice results in an effective hydrodynamic boundary slightly displaced from the physical boundary. In the case of spherical particles with a radius a this leads to a hydrodynamic radius $a_{\mathrm{hd}}=a+\Delta$, where the deviation Δ depends on the fluid viscosity [2,4]. An *a priori* correction of a_{hd} can achieve more accurate dynamics of the solid particles.

Fluid-particles forces. We begin focusing on two approaches to deal with the interaction between fluid flow and solid suspensions, the method proposed by Ladd [2,3], and the ALD method [5–7] (Fig. 3). Both techniques employ the momentum exchange algorithm [2] to deal with hydrodynamical forces exerted on the particles. According to its original formulation, the MEA is used to approximate the momentum given to the particle by the surrounding fluid. In detail, for a boundary link i^* at a node $\mathbf{j}$ (Fig. 2) where Eq. (12) is applied, the momentum

$$\begin{aligned}\mathbf{p}_{i*}(n_+,\mathbf{j}) &= \mathbf{c}_{i*}f_i(n_+,\mathbf{j}) - \mathbf{c}_i f_i(n+1,\mathbf{j}) \\ &= 2\mathbf{c}_{i*}[f_i(n_+,\mathbf{j}) - hc_s^{-2}w_i\mathbf{c}_{i*}\cdot\mathbf{u_b}]\end{aligned} \tag{13}$$

is transferred to the particle. Force and torque exerted on a particle are computed by a summation of the contributions (13) over all the boundary link i_b around its surface:

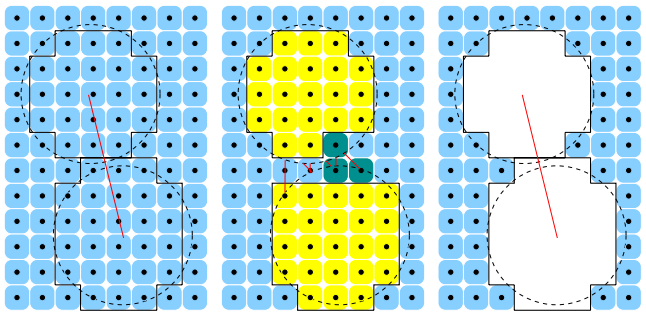

FIG. 3. (Color online) LBM approaches for suspended particles. Left: Ladd method: Particles are represented by a *solid shell*. Inner points, belonging to the physical solid domain, are treated as normal fluid nodes. Center: ALD method. The inner fluid has no physical meaning, but it is still used to initialize new fluid nodes. Solid nodes of different particles are turned into fluid nodes if they are in contact. Bridge links are created via which lubrication force elements are applied. Right: In the present work we use an improved method applying a corrected momentum exchange (21) in combination with a refill method (22). The latter removes the need for an explicit simulation of inner fluid.

$$\mathbf{F}_{\mathrm{hd}}(n) = \sum_{i_b}\mathbf{p}_{i_b}(n_+,\mathbf{j}),$$

$$\mathbf{T}_{\mathrm{hd}}(n) = \sum_{i_b}[\mathbf{x_j}-\mathbf{x}_C]\times\mathbf{p}_{i_b}(n_+,\mathbf{j}). \tag{14}$$

In its original version [2] also the nodes inside the solid particles are treated as fluid, which contribute to the hydrodynamical forces as well. In other words, the summations (14) contain both fluid-solid and solid-fluid links. A particle is therefore composed of a rigid shell and the inner fluid in this description.

However, for solid phase densities $\rho_s \leqslant \rho_f$ the explicit update of particle positions has been found to become unstable [3] and improvements on the update have been proposed [4,14]. The applicability was later extended [4] to smaller ratios ρ_s/ρ_f by omitting the action of the inner fluid on the shell.

In the ALD method, the inner fluid is only virtual, but is still used to bypass an explicit treatment of the nodes which become fluid when the particle moves. In this description, the fluid momentum of a new fluid node has to be taken from the particles momentum, while, when the particle "absorbs" a fluid site its momentum is given to the particle. Formally, the momentum exchange

$$\mathbf{p}_{\mathrm{f}\to\mathrm{s}} = \rho(n,\mathbf{j})[\mathbf{u}(n,\mathbf{j}) - \mathbf{U}_p] \tag{15}$$

leads then to another contribution $\mathbf{F}_{\mathrm{f}\to\mathrm{s}}$ to force and torque on the particle.

When particles come close to each other the fluid pressure between the particles might change drastically. This is hardly captured by the lattice Boltzmann flow if the grid resolution is not fine enough. To overcome the lubrication breakdown problem, an explicit lubrication model has to be included, based on (asymptotic) lubrication theory [4]. In most suspension models a single particle-particle force term is used. With

TABLE I. Comparisons of the approaches of Ladd [2,3], ALD [5–7], and CMES combined with an accurate refill (Sec. III), concerning the main algorithmic issues for LB suspension flows. In the case of Ladd method, the algorithm from [2,3] has been combined with the later proposed lubrication correction [4]. Some features are formally independent from each other, which leads to numerous combinations. For simplicity, we restricted this work to the original formulations.

	Ladd	ALD	Present paper
Inner fluid	present, active	present, virtual	no
Refill	(un)cover of inner fluid nodes	(un)cover of inner fluid nodes	equil+noneq. refill [15]
Lubrication	linear theory, single two-part. term [4]	linear theory, link-wise force terms	linear theory, single two-part. term [16]
Force computation	MEA	MEA	corrected MEA [17]

the ALD method linkwise lubrication corrections [7] were introduced which provide a better resolution of lubrication forces between arbitrary shaped particles or walls.

Combining hydrodynamical forces $\mathbf{F}_{\mathrm{hd}}$, possible $\mathbf{F}_{\mathrm{f}\to\mathrm{s}}$, lubrication forces $\mathbf{F}_{\mathrm{lub}}$; and according torques, positions, and velocities of the particles are updated by an integration of the equations of motion for rigid bodies.

To summarize the methods, in Table I the main features of the approaches are compared. Note that some features are independent from others (e.g., refilling new fluid nodes and lubrication corrections) and this produces several combinations. For simplicity, we focused on setups reflecting the original ideas.

B. Lees-Edwards boundary conditions

Lees-Edwards boundary conditions (LEbc's) [8] are commonly used in MD simulations [18] to maintain a constant shear over the simulation domain. LEbc's are analogous to periodic boundary conditions, with the difference that in one spatial dimension, for example, y in two dimensions, the upper periodic copy of the system is moving with a constant horizontal speed $\mathbf{u}_{\mathrm{LE}}=(u_{\mathrm{LE}},0)$, while the lower one moves in the opposite direction with $-(u_{\mathrm{LE}},0)$ [Fig. 4(a)]. This induces a shear with a rate $\dot{\gamma}=u_{\mathrm{LE}}/L_y$, where L_y is the height of the periodic domain. The use of LEbc's in suspension simulations allows us to remove the influence of the shearing walls in the commonly used Couette flow scheme, which results in a depletion zone and slip, observable in real experiments and simulations.

Implementation of LEbc's exists also for CFD methods (see, for example, [19]). For lattice Boltzmann flows with suspensions, results with LEbcs have been reported in [20]. In a recent publication [21] LEbc's were used for shear flow simulations of suspensions of deformable particles. In both cases the standard MEA was used, either together with the suspension method by Ladd, or, in the latter, with the ALD method.

A short description of our LEbc formulation for LBM is given below. For more details on the method, and for a wider discussion of the specific applications, we refer the reader to [22].

Implementing LEbc's within the LBM framework, one has to deal with density distributions of the LB fluid and solid particles which cross the Lees-Edwards boundary, i.e., the boundary of the system moving with a speed u_{LE}. Three major practical problems have to be solved: (i) the mapping of densities between lattices with subgrid displacement, (ii) the transform of densities that are based on a set of speeds of a certain lattice, and (iii) subgrid boundary conditions due to different representations of one particle on two subgrid shifted lattices.

The issue (i) can be resolved using a first order interpolation, mapping the density by allocating it to the two nodes that partly overlap the nonexisting destination node of the density, e.g.,

$$f_i(\mathbf{x}_{l2}) = s f_i(\mathbf{x}_{l1} + \mathbf{c}_i) + (1-s) f_i(\mathbf{x}_{l1}) \tag{16}$$

(omitting the time dependence for brevity), where $s = \mathrm{mod}(s_{\mathrm{LE}},1)$ depends on the shift s_{LE} between different copies of the system, and the sites $\mathbf{x}_{l1},\mathbf{x}_{l2}$ belong to the two lattices $l1$ and $l2$.

Regarding (ii), in [23] it could be shown that it is sufficient to approximate the Galilean transform of a density into a different reference frame by the transform of its equilibrium part. Since $\mathbf{u}_{l2}=\mathbf{u}_{l1}+\mathbf{u}_{\mathrm{LE}}$, we have

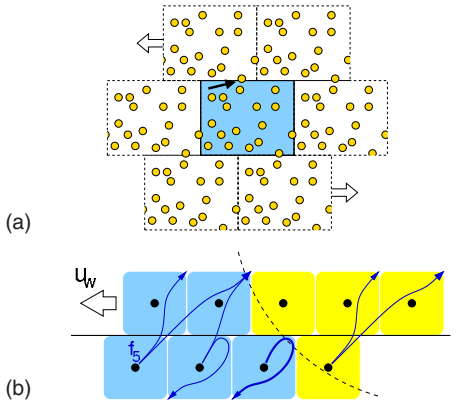

FIG. 4. (Color online) (a) Applying Lees-Edwards boundary conditions, periodic copies of the system move with horizontal velocity. (b) Zoomed into the region where the black arrow points to in (a), the propagation of densities f_5 from nodes at the top of the original reference frame in the case a particle crosses the LE boundary are shown.

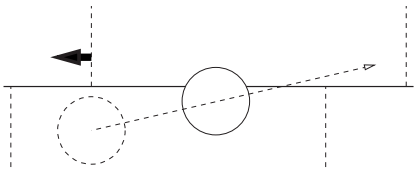

FIG. 5. The benchmark D_1: single particle crossing the boundary between different reference frames.

$$f_i(\mathbf{x}_{l2}) = f_i(\mathbf{x}_{l1}) + f_i^{\mathrm{eq}}(\rho, \mathbf{u}_{l2}) - f_i^{\mathrm{eq}}(\rho, \mathbf{u}_{l1}). \tag{17}$$

Particles are described by the coordinates of their centers. To approach (iii) a combination of the reallocation (16) and a fractional reflection rule for densities at solid boundaries, inspired by the boundary conditions presented in [24], can be applied together with the extension for moving boundaries (12).

The approach is illustrated in Fig. 4(b) on the basis of densities f_5 that have to cross a LE boundary or are (partly) reflected at it.

C. First numerical tests

We aim for a method which resolves the dynamics of particles independently from any constant velocity superposed to the movement of local centers of mass as present in all nontrivial flow problems. In the case of errors due to the violation of Galilean invariance, such errors would be amplified if the same physical problem is described using coupled reference systems, like in the application of LEbc's. We used the LEbc proposed in [22] (and outlined in the previous section) to allow simulations of homogeneously sheared systems and to superpose them with a small, constant velocity.

For the simulations presented in the following we used a system size $L_x = L_y = 128$ and a particle radius $R = 4.8$ (all numbers given in lattice units). The density ratio was set to $\rho_s/\rho_f = 10$ and the kinematic viscosity of the fluid $\nu = 1/6$.

Single disk (D_1). We simulate a single particle in a periodic shear flow. In detail, initially a particle is placed in the midpoint of a squared domain, with a velocity equal to the fluid velocity at that point $\mathbf{v}_p(t=0) = \mathbf{u}_s(t=0) = (u_x, 0.005)$ (Fig. 5). Additionally the flow is initialized and maintained at a horizontal shear $\dot{\gamma} = u_{\mathrm{LE}}/L_y$ and the particle's angular speed is accordingly initialized to $\omega(t=0) = \dot{\gamma}/2$. With that an equilibration of near-particle fluid field and ω was reached after approximately $t = 3000$. In any case, this equilibration process is physical and should therefore be Galilean invariant, too.

Throughout the simulation we measure the vertical velocity of the particle $\mathbf{v}_{p,y}(t)$, which should remain constant in the ideal case. However, as shown in Fig. 6, in the case of the ALD method a significant deviation from $\mathbf{v}_{p,y}(t=0) = 0.005$ can be observed already from early times on, not having reached the LEbc yet. When the Ladd method is used, no deviations can be seen during that time, which finds its explanation in the fact that every error made in linkwise calculations is compensated by the use of physical inner fluid, where every momentum exchange at a boundary link has its inner inverted counterpart. Using the ALD method, e.g., applying MEA only at the outer surface, the particles feel a nonphysical vertical force in situations where points at the upper surface move at a different speed than points at the lower surface.

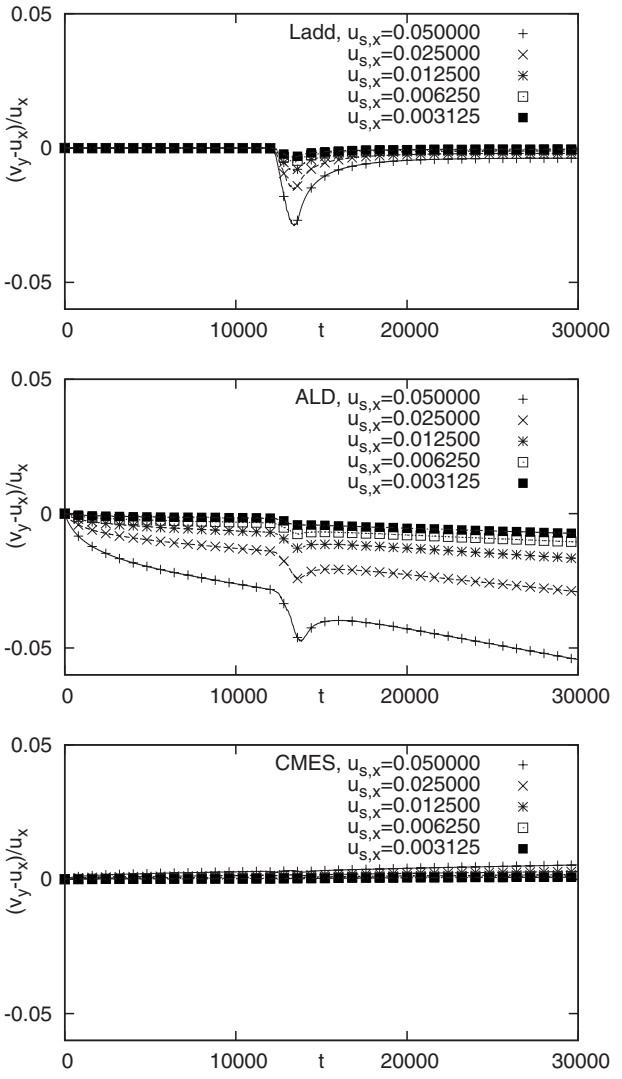

FIG. 6. The vertical speed v_y of a single particle in a sheared system ($\dot{\gamma} = 5 \times 10^{-5}$) crossing a LEbc for different superposed velocities $u_{s,x}$. At about $t = 12\,300$ the particle reaches the LEbc and crosses it within a time $t_c \approx 2000$.

Deviations in both methods appear when the particle crosses the LEbc. Notice that in this situation boundary links exist that connect nodes in different reference frames [in practice, different $\mathbf{u}_b$ are used in Eq. (13)]. Here, errors cannot be compensated by the inner fluid. The non-Galilean effect grows linearly when varying u_x. A shear rate $\dot{\gamma} = 5 \times 10^{-5}$ was present in all these measurements.

Figure 7 shows similar measurements. Here the curves are recorded for different $\dot{\gamma}$ and the superposed initial velocity $\mathbf{u}_s = (0.05, 0.005)$ was kept constant. All the curves show similar behavior compared to the measurements in Fig. 6, both in absolute size and order. MEA in its form (13) leads to a nonzero force integration over the surface depending on both the absolute particle speed as well as the difference

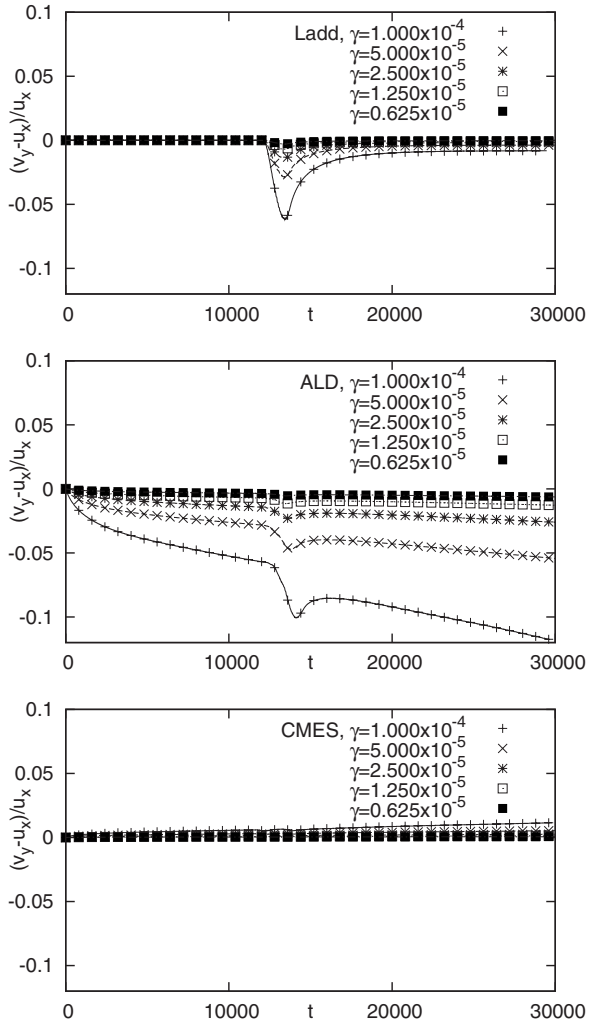

FIG. 7. The vertical speed v_y of a single particle in a sheared system crossing a LEbc for different shear rates $\dot{\gamma}$. The particle is moving with the fluid at a superposed background velocity $u_s = (0.05, 0.005)$.

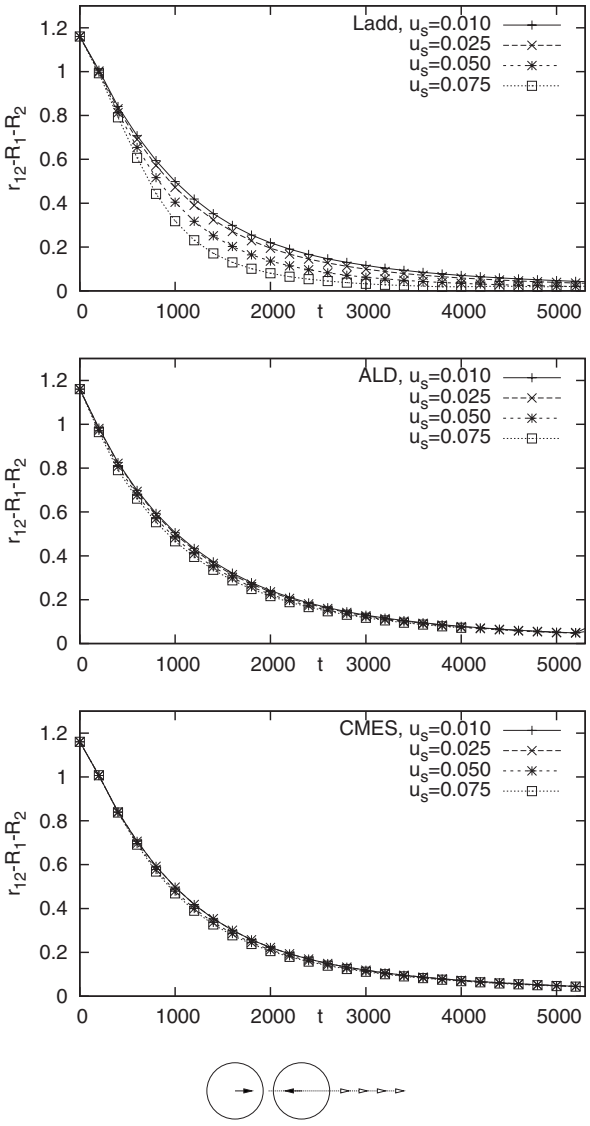

FIG. 8. Test D_2 a: Gap size between two particles that feel horizontal forces pushing them together for different absolute values of the superposed velocity $\mathbf{u}_s$. The direction is $\mathbf{u}_s \| \mathbf{r}_{1,2}$. The three plots show results obtained with different suspension models.

between surface velocities at the upper half and the lower half of a particle.

Approaching disks (D_2). The next example aims to demonstrate the effects of combining LB fluid-structure interaction with suspension dynamics, when different flow velocities are used as background. We consider two equal particles moving toward each other, driven by a small external force (Figs. 8 and 9). The gap at time $t=0$ is $r_{1,2}-R_1-R_2 \approx 1.19$, in order to have at least one fluid node between the particles at the beginning. Without adaptive grid-refinement lubrication forces have to be implemented explicitly. Regarding the different suspension methods, lubrication corrections were applied as given in Table I. For simulations using CMES we applied a lubrication correction in the form proposed in [16] for two-dimensional systems. Such forces were invoked for gaps smaller than $h_c = 1.0 \approx 0.21R$.

Both particles are always aligned in the x direction. To mimic the fact that particle-particle collisions happen at different locations in a shear flow, the flat (nonsheared) flow field is superposed by a constant translational velocity $\mathbf{u}_s$ and the experiment is repeated using different absolute values and directions of $\mathbf{u}_s$. The particles are always initialized with the same velocity as the fluid. Measuring the gap between the particles as a function of time, we can investigate the combined effect of force computation, lubrication corrections, and nodes' re-initialization.

Figure 8 shows the results obtained with superimposed velocities $\mathbf{u}_s = (u_x, 0)$. Looking at the problem from the lattice reference frame, one particle is following the other while they get closer. In an ideal case, the lubrication-damped collision of the particles should not be affected by the absolute velocity of their center of mass. However, using Ladd's method at higher superposed velocities the particles tend to approach themselves faster in comparison to a simulation where the center of mass is at rest (which is the usual way lubrication behavior investigations are carried out in the literature). Fluid between particles is of higher pressure than the fluid at the opposite sides of the particles. When particles

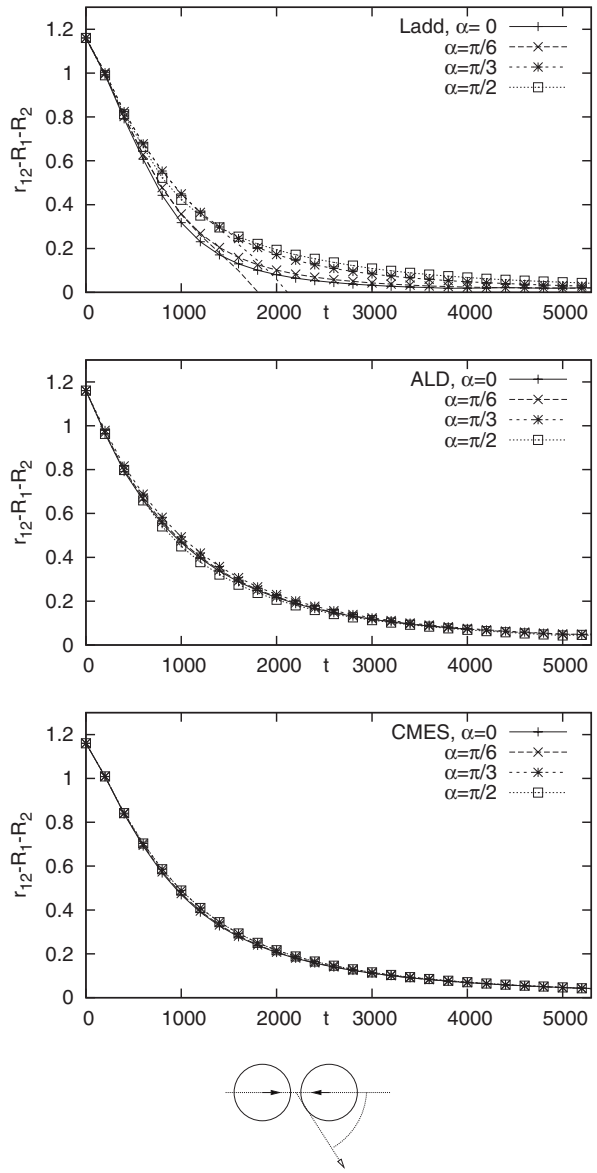

FIG. 9. Test D_2 b: Gap size between two particles for different angles $\alpha = \angle(\mathbf{u}_s, \mathbf{r}_{1,2})$. Results for the three suspension models are shown.

move, fluid nodes will turn into inner fluid. Not instantly in equilibrium with the rest of the inner fluid, such a high-pressure inside fluid causes forces that counteract lubrication forces. As an additional effect, adopting high-pressure gap nodes also leads to a kind of pumping: the following particle gains mass with the increase of inner-fluid pressure which biases the particle dynamics. The "pumping effect" has another consequence: When inner fluid turns into outside fluid behind the particle the increased pressure also contributes to a less damped collision behavior of the particles.

Although the last argument could also apply in the case of simulations using ALD, results obtained with this method show a better behavior. Here, by definition of the model, fluid nodes always exist between the particles. As a consequence we could observe that the pumping effect was less pronounced.

When particles are very close and therefore lubrication forces very sensitive we would have expected that the additional momentum exchanged when nodes are covered or uncovered lead to deviant behavior. However, this seems to have little importance for the particle dynamics in these tests. Although small deviations can be seen, the action of a physical inner fluid turns out to be the most important source of errors.

In Fig. 9 we show the results for the case of a constant absolute superposed velocity $|\mathbf{u}_s| = 0.075$ but for different directions. The curves for $\alpha = 0, \pi/6, \pi/3, \pi/2$ show only slight deviations from each other if ALD is used. In the case of active inner fluid (as in Ladd's method), a strong angle-dependent behavior can be observed. For $\alpha = \pi/6, \pi/3$ another effect appears. If the background velocity has an inclination $0 < \alpha < \pi/2$ with respect to the vector $\mathbf{r}_{1,2} = \mathbf{x}_{p2} - \mathbf{x}_{p1}$, we still observe a small pumping. With the broken symmetry of the problem the different properties of the two particles cause the particles to start to tumble around each other. The results in Fig. 9 show the results for the case of a constraint $y_{p1}(t) = y_{p2}(t)$ was applied and curves that cross the x axis in the case of tumbling.

III. CORRECTED MOMENTUM EXCHANGE FOR SUSPENSIONS

The previous results show that Ladd's method and ALD have shortcomings. Using the asymptotic expansion technique it is possible to identify the main source of errors, and to derive leading order corrections to the original algorithms.

Galilean invariant force computations. Starting from Eq. (8), which approximates the solution of the LBM (1), a prediction for the momentum exchange contributions (13) can be derived.

Let us consider a boundary node $\mathbf{k}$, and an outgoing boundary link i at $\mathbf{k}$. We denote with $\mathbf{b}_i(\mathbf{k})$ the intersection between the link $\mathbf{c}_i$ and the fluid-solid interface. Inserting Eq. (8) into Eq. (13) we obtain (dropping the time dependence for brevity)

$$\mathbf{p}_i(\mathbf{k}) = \mathbf{p}_i^{(0)}(\mathbf{x}_\mathbf{k}) + h^2 \mathbf{p}_i^{(2)}(\mathbf{x}_\mathbf{k}) + O(h^3), \tag{18}$$

with

$$\mathbf{p}_i^{(0)}(\mathbf{x}_\mathbf{k}) = 2 w_i \mathbf{c}_i,$$

$$\mathbf{p}_i^{(2)}(\mathbf{x}_\mathbf{k}) = 2 w_i c_s^{-2} \left(p + \frac{c_s^{-2}}{2} (|\mathbf{c}_i \cdot \mathbf{u}_\mathbf{b}|^2 - c_s^2 \mathbf{u}_\mathbf{b}^2) - c_s^{-2} \nu \mathbf{c}_i \cdot \nabla \mathbf{u}_\mathbf{b} \cdot \mathbf{c}_i \right) \mathbf{c}_i, \tag{19}$$

where the quantities on the right hand sides are evaluated at $\mathbf{b}_i(\mathbf{k})$.

Using this approach, it can be proved [25,17] that MEA yields a first order accurate approximation of the force acting on a particle P,

$$\mathbf{F}_{\mathrm{P}}(t) = \int_{\partial P(t)} [-p(\mathbf{x})\mathbf{n} + \mathbf{n} \cdot \mathbf{S}(\mathbf{x})] d\gamma. \tag{20}$$

Beyond this general result, Eq. (19) provides additional useful information. We focus on the second order coefficient of Eq. (19), which contains a term not related to the boundary force, explicitly depending on the boundary velocity, and responsible for breaking the Galilean invariance.

It must be remarked that in most cases this term produces a small global contribution [25]. However, this does not apply using Lees-Edwards boundary condition, when the solid-fluid interaction of one particle is evaluated in different reference systems within the same time step (in practice, in this case the MEA depends on the contributions of two parts of the interfaces, with different velocities).

Aware of Eq. (19) we can easily define a correction for the momentum exchange algorithm which reads

$$\mathbf{p}_i^{\mathrm{CME}}(\mathbf{k}) = \mathbf{p}_i(\mathbf{k}) - 2w_i\mathbf{c}_i - h^2 w_i c_s^{-4} \times \{|\mathbf{c}_i \cdot \mathbf{u}_{\mathbf{b}}[\mathbf{b}_i(\mathbf{k})]|^2 - c_s^2 \mathbf{u}_{\mathbf{b}}[\mathbf{b}_i(\mathbf{k})]^2\}\mathbf{c}_i. \tag{21}$$

Together with $\mathbf{p}_i(\mathbf{k})$ in Eq. (13) and the summation over the particle surface in Eq. (14) this defines a corrected momentum exchange algorithm (CMES) for the calculation of hydrodynamical forces and the torque on the particle.

The violation of Galilean invariance by MEA, when used *locally*, has been already observed in [26] and cured by the introduction of additional virtual fluid nodes inside the solid domain. However, this is a special solution and care has to be taken with the implementation of such an idea when boundary links cross a Lees-Edwards boundary. The correction (21) offers a consistent analytical solution and, in most cases, allows an easier implementation.

Initialization of new fluid nodes. Reinitialization of new fluid nodes is a common task dealing with moving boundary LBM. Both Ladd's and the ALD approach make use of inner fluid to deal with this. In general, this is justified if inner nodes hold densities which approximate the right characteristics when turning from an inner surface node to an outer surface node. This is likely only in situations where the particle acceleration is low. If a particle is accelerated the fluid directly behind the particle is typically of lower pressure while the adjacent node inside the shell would be of rather high pressure.

The problem can be solved by a more accurate initialization of LB densities according to the expansion coefficients (8). This can be done in an efficient way, separately approximating the *equilibrium* and the *nonequilibrium* part, as described in [22,15]. In detail, denoting with $\mathbf{k}$ a new fluid node, first velocity $\mathbf{u}(\mathbf{k})$ and pressure $p(\mathbf{k})$ are extrapolated using a set of available neighboring fluid node, to construct the equilibrium distribution,

$$\tilde{f}_i^{\mathrm{eq}}(\mathbf{k}) = E_i(\rho_0 + c_s^2 h^2 p(\mathbf{k}), \mathbf{u}(\mathbf{k})).$$

Then, a low order extrapolation for the nonequilibrium part $\tilde{f}_i^{\mathrm{neq}}(\mathbf{k})$ is added, initializing (omitting the time dependence)

$$f_i(\mathbf{k}) = \tilde{f}_i^{\mathrm{eq}}(\mathbf{k}) + \tilde{f}_i^{\mathrm{neq}}(\mathbf{k}). \tag{22}$$

In practice, $\tilde{f}_i^{\mathrm{neq}}(\mathbf{k})$ can be copied from a neighboring node [15]. This approach is by definition locally consistent with the inner LB solution, and yields the same accuracy as the standard LBM [25,15], providing a good balance between computational effort and quality of results. We remark that similar approaches, based on extrapolation techniques, have been proposed in [13,27], which also achieved accurate refill. However, these are based on more complicated extrapolations and become less practical when dealing with the flow of dense suspensions. In practice, in the case of colliding particles it might happen that not enough nodes are available in order to implement algorithm (22) with the required accuracy. In those cases, we have used lower order approximations for the equilibrium distribution, based on simpler averages (but still including a nonequilibrium approximation).

A. Numerical results

Including the modifications described in the previous section, both tests D_1 and D_2 are solved more accurately.

Correction (21) for the momentum exchange algorithm significantly reduces non-Galilean deviations in D_1 (see Figs. 6 and 7) in comparison to the results obtained by standard MEA, applied only at the outer surface (ALD). Still, it can be observed that CMES does not behave as accurate as the original method by Ladd if the particle is treated in only one reference frame, but this is mainly due to additional symmetry properties of Ladd's methods in the specific benchmark. In general, Figs. 6 and 7 show that the particle dynamics has significantly improved by the use of CMES when particle-surface interaction has to be treated partly in different reference frames.

The benchmark D_2 demonstrates that the use of an accurate equilibrium+nonequilibrium refill improve the resolution of particle dynamics, avoiding effects that are caused by just turning physical inner fluid nodes into outer fluid nodes and vice versa (Figs. 8 and 9).

We remark that in D_2 the ALD method also produces results almost independent of $\mathbf{u}_s$. However, it still requires the computation of an inner fluid. Aiming at simulations of dense suspensions the cost for this is in the order of $\phi = V_s/V_f$. With the application of a refill procedure as proposed in this paper the computation of inner fluid can be omitted, providing a theoretical speedup factor of $1/(1-\phi)$.

IV. CONCLUSION

Due to the pseudocompressible nature of the LBM, the dynamics of simulated suspended particles may show non-Galilean invariant effects. These contributions are typically small in many applications, and comparable with the size of numerical errors arising in the LB flow.

We have focused on the simulation of particles in a shear flow, investigating the effects of the Lees-Edwards boundary condition, and, in general, of the usage of LBM to resolve suspension dynamics in different reference systems.

Simple numerical experiments constructed to highlight the effects of violated Galilean invariance showed that standard methods may produce unsatisfactory results. Using the asymptotic expansion technique and accurate algorithms for fluid-structure interaction problems, we have been able to identify and remove the main sources of error, which resulted in a corrected momentum exchange method for suspensions (CMES).

In simulations of two approaching particles the effect of a moving reference frame was investigated. We could show that an accurate refill method can remove negative effects arising from the use of inner fluid. We proposed a refill method based on interpolation of equilibrium and nonequilibrium parts which gives results comparable to similar methods, but is easier to implement in many situations.

In an upcoming work [22] we will present an application of CMES together with Lees-Edwards boundary conditions which allow simulations of sheared suspensions not bounded by walls (like in a Couette-type flow).

ACKNOWLEDGMENT

This research was supported by the European Commission, through the COAST project [28] (EU-FP6-IST-FET Contract No. 033664).

[1] S. Succi, *The Lattice Boltzmann Equation for Fluid Dynamics and Beyond* (Oxford University Press, Oxford, 2001).
[2] A. J. C. Ladd, J. Fluid Mech. **271**, 285 (1994).
[3] A. J. C. Ladd, J. Fluid Mech. **271**, 311 (1994).
[4] N. Q. Nguyen and A. J. C. Ladd, Phys. Rev. E **66**, 046708 (2002).
[5] C. Aidun and Y. Lu, J. Stat. Phys. **81**, 49 (1995).
[6] C. Aidun, Y. Lu, and E.-J. Ding, J. Fluid Mech. **373**, 287 (1998).
[7] E.-J. Ding and C. Aidun, J. Stat. Phys. **112**, 685 (2003).
[8] A. W. Lees and S. F. Edwards, J. Phys. Colloq. **5**, 1921 (1972).
[9] G. R. McNamara and G. Zanetti, Phys. Rev. Lett. **61**, 2332 (1988).
[10] I. Ginzburg, Adv. Water Resour. **28**, 1171 (2005).
[11] M. Junk, A. Klar, and L.-S. Luo, J. Comput. Phys. **210**, 676 (2005).
[12] I. Ginzburg and D. d'Humières, Phys. Rev. E **68**, 066614 (2003).
[13] M. Bouzidi, M. Firdaouss, and P. Lallemand, Phys. Fluids **13**, 3452 (2001).
[14] C. P. Lowe, D. Frenkel, and A. J. Masters, J. Chem. Phys. **103**, 1582 (1995).
[15] A. Caiazzo, Prog. Comput. Fluid Dyn. **8**, 3 (2008).
[16] J. Kromkamp, D. T. M. van den Ende, D. Kandhai, R. M. van der Sman, and R. M. Boom, J. Fluid Mech. **529**, 253 (2005).
[17] A. Caiazzo and M. Junk, Comput. Math. Appl. **55**, 1415 (2008).
[18] B. Smit and D. Frenkel, *Understanding Molecular Simulation* (Academic Press Inc., San Diego, 1996).
[19] W. R. Hwang and M. A. Hulsen, J. Non-Newtonian Fluid Mech. **136**, 167 (2006).
[20] S. V. Lishchuk, I. Halliday, and C. M. Care, Phys. Rev. E **74**, 017701 (2006).
[21] R. M. MacMeccan, J. R. Clausen, G. P. Neitzel, and C. K. Aidun, J. Fluid Mech. **618**, 13 (2009).
[22] E. Lorenz, A. Caiazzo, and A. G. Hoekstra, Phys. Rev. E **79**, 036706 (2009).
[23] A. J. Wagner and I. Pagonabarraga, J. Stat. Phys. **107**, 521 (2002).
[24] R. Verberg and A. J. C. Ladd, Phys. Rev. Lett. **84**, 2148 (2000).
[25] A. Caiazzo, Ph.D. thesis, Scuola Normale Superiore, Pisa, Italy, and Technische Universität Kaiserslautern, Germany, 2007.
[26] R. M. MacMeccan, Ph.D. thesis, Georgia Institute of Technology, 2007.
[27] P. Lallemand and L.-S. Luo, J. Comput. Phys. **184**, 406 (2003).
[28] www.complex-automata.org, the COAST project.