

Lubrication corrections for lattice-Boltzmann simulations of particle suspensions

N.-Q. Nguyen and A. J. C. Ladd*

Department of Chemical Engineering, University of Florida, Gainesville, Florida 32611-6005

(Received 5 July 2002; published 30 October 2002)

The lattice-Boltzmann method has been refined to take account of near-contact interactions between spherical particles. First, we describe a comprehensive solution to the technical problems that arise when two discretized surfaces come into contact. Second, we describe how to incorporate lubrication forces and torques into lattice-Boltzmann simulations, and test our method by calculating the forces and torques between a spherical particle and a plane wall. Third, we describe an efficient update of the particle velocities, taking into account the possibility that some of the differential equations are stiff.

DOI: 10.1103/PhysRevE.66.046708

PACS number(s): 83.85.Pt, 47.11.+j, 47.15.Pn

I. INTRODUCTION

Lattice-Boltzmann simulations [1,2] are being increasingly used to calculate hydrodynamic interactions [3–9], by direct numerical simulation of the fluid motion generated by the moving interfaces. However, as two particles approach each other the calculation of the hydrodynamic force breaks down in the gap between the particles, typically when the minimum distance between the two surfaces is of the order of the grid spacing. For example, it is impractical to use sufficient mesh resolution to resolve the flow in the small gaps that can result from an imposed shear flow. At high shear rates the rheology of a suspension of spheres is qualitatively affected by lubrication forces between particles separated by gaps less than $0.01a$, where a is the particle radius [10,11]. A simulation at this resolution ($\approx 10^7$ grid points per particle) is unfeasible for more than a few particles. The number of grid points can be reduced by using body fitted coordinates [12] or adaptive meshes [13,14], but the small zone size in the gap reduces the time step that can be used to integrate the flow field [15]. The problem is exacerbated by the uniform grid used in lattice-Boltzmann simulations, but it should be noted that similar techniques, using particles embedded in regular grids, are becoming increasingly popular in computational fluid dynamics [16,17]. Some aspects of this work may therefore be applicable to such methods as well.

Simulations of hydrodynamically interacting particles typically use multipole expansions of the Stokes equations [18,19]. Again the calculations break down when the particles are close to contact, unless an impractical number of multipole moments are used [20]. A solution to this problem is to calculate the lubrication forces between pairs of particles for a range of small interparticle gaps, and then patch these solutions on to the friction coefficients calculated from the multipole expansion. The method exploits the fact that lubrication forces can be added pair-by-pair, and has been shown to work well in practice [18]. A simplified version of this approach has already been adopted to include normal lubrication forces in lattice-Boltzmann simulations [21]. In this paper we extend our previous work to include all com-

ponents of the lubrication force and torque, and test the numerical scheme for the interactions between a spherical particle and a planar wall. We find that the hydrodynamic interactions can be well represented over the entire range of separations by patching only the most singular components of the lubrication force onto the force calculated from the lattice-Boltzmann model. This is simpler than the Stokesian dynamics approach, where the patch is calculated at every separation.

We begin this paper with a summary of the lattice-Boltzmann algorithm for particle suspensions [1,2]. We discuss recent innovations by other groups [6,22], and describe some additional improvements to these methods. In particular, we propose a comprehensive solution to the technical difficulties that arise when particles are close to contact, in particular the loss of mass conservation. The bulk of our numerical results are a series of detailed tests of the hydrodynamic interactions between two surfaces in near contact. We demonstrate that after including corrections for the lubrication forces we obtain accurate results over a wide range of fluid viscosities. Finally, we describe an efficient implicit algorithm for updating the particle velocities even in the presence of stiff lubrication forces. An explicit solution of these differential equations requires either that the particles are much denser than the surrounding fluid [2], or that very small time steps are used to update the particle velocities. On the other hand, if the particle velocities are updated implicitly, the resulting system of linear equations severely limits the number of particles that can be simulated. In this paper we describe what we call a “cluster-implicit” method, whereby groups of closely interacting particles are grouped in individual clusters and interactions between particles in the same cluster are updated implicitly, while all other lubrication forces are updated explicitly. In fluidized suspensions clusters typically contain less than ten particles, even at high concentrations, so that the implicit updates are a small overhead. Our simulations efficiently cope with clusters of several hundred particles by using conjugate-gradient methods, and only slow down if the number of particles in the cluster exceeds 10^3 .

II. THE LATTICE-BOLTZMANN METHOD

The lattice-Boltzmann equation describes the time evolution of a discretized velocity distribution function $n_i(\mathbf{r}, t)$

*Email address: ladd@che.ufl.edu

$$n_i(\mathbf{r}+\mathbf{c}_i\Delta t, t+\Delta t) = n_i(\mathbf{r}, t) + \Delta_i[\mathbf{n}(\mathbf{r}, t)], \quad (1)$$

where Δ_i is the change in n_i due to molecular collisions, and Δt is the time step. This one-particle distribution function describes the mass density of particles with velocity \mathbf{c}_i , at a lattice node \mathbf{r} , at a time t ; \mathbf{r} , t , and \mathbf{c}_i are discrete, whereas n_i is continuous. The hydrodynamic fields, mass density ρ , momentum density $\mathbf{j}=\rho\mathbf{u}$, and momentum flux $\mathbf{\Pi}$, are moments of this velocity distribution,

$$\rho = \sum_i n_i, \quad \mathbf{j} = \rho\mathbf{u} = \sum_i n_i\mathbf{c}_i, \quad \mathbf{\Pi} = \sum_i n_i\mathbf{c}_i\mathbf{c}_i. \quad (2)$$

In three dimensions, isotropy requires a multispeed model; for example, the 18-velocity model described in Ref. [23], which uses the [100] and [110] directions of a simple cubic lattice. In this work the 18-velocity model is augmented with stationary particles, which enables it to account for small deviations from the incompressible limit, although in simulations of stationary flows we have found the numerical differences to be small [24].

A computationally useful form for the collision operator $\Delta_i(\mathbf{n})$ can be constructed by linearizing about the local equilibrium \mathbf{n}^{eq} [23],

$$\Delta_i(\mathbf{n}) = \Delta_i(\mathbf{n}^{eq}) + \sum_j \mathcal{L}_{ij}n_j^{neq}, \quad (3)$$

where \mathcal{L}_{ij} are the matrix elements of the linearized collision operator, $n_j^{neq} = n_j - n_j^{eq}$, and $\Delta_i(\mathbf{n}^{eq}) = 0$. The form of the equilibrium distribution is constrained by the moment conditions required to reproduce the inviscid (Euler) equations on large length scales and time scales. In particular, the second moment of the equilibrium distribution should be equal to the inviscid momentum flux, $\mathbf{\Pi}^{eq} = \sum_i n_i^{eq}\mathbf{c}_i\mathbf{c}_i = p\mathbf{1} + \rho\mathbf{u}\mathbf{u}$. A suitable form for the equilibrium distribution of the 19-velocity model is [25]

$$n_i^{eq} = a^i \left[\rho + \frac{\mathbf{j} \cdot \mathbf{c}_i}{c_s^2} + \frac{\rho\mathbf{u}\mathbf{u} : (\mathbf{c}_i\mathbf{c}_i - c_s^2\mathbf{1})}{2c_s^4} \right], \quad (4)$$

where $c_s = \sqrt{c^2/3}$, $c = \Delta x/\Delta t$, $p = \rho c_s^2$, and the coefficients of the three speeds are

$$a^0 = \frac{1}{3}, \quad a^1 = \frac{1}{18}, \quad a^{\sqrt{2}} = \frac{1}{36}. \quad (5)$$

In our suspension simulations we use a 3-parameter collision operator, allowing for separate relaxation of the five shear modes, one bulk mode, and nine kinetic modes. The postcollision distribution $n_i^* = n_i + \Delta_i$ is written as a series of moments [Eq. (2)], as for the equilibrium distribution [Eq. (4)] [24],

$$n_i^* = a^i \left(\rho + \frac{\mathbf{j} \cdot \mathbf{c}_i}{c_s^2} + \frac{(\rho\mathbf{u}\mathbf{u} + \mathbf{\Pi}^{neq,*}) : (\mathbf{c}_i\mathbf{c}_i - c_s^2\mathbf{1})}{2c_s^4} \right). \quad (6)$$

The zeroth (ρ) and first ($\mathbf{j}=\rho\mathbf{u}$) moments [Eq. (2)] are unchanged, but the nonequilibrium second moment, $\mathbf{\Pi}^{neq}$, is modified by the collision process,

$$\mathbf{\Pi}^{neq,*} = (1+\lambda)\overline{\mathbf{\Pi}^{neq}} + \frac{1}{3}(1+\lambda_v)(\mathbf{\Pi}^{neq}:\mathbf{1})\mathbf{1}, \quad (7)$$

where $\mathbf{\Pi}^{neq} = \mathbf{\Pi} - \mathbf{\Pi}^{eq}$; the eigenvalues λ and λ_v are related to the shear and bulk viscosities and lie in the range $-2 < \lambda < 0$.

The macrodynamical behavior arising from the lattice-Boltzmann equation can be found from a multitime-scale analysis [24,26]. A complete error analysis is rather lengthy [24], but it can be shown that for sufficiently low velocities the convergence is quadratic in the lattice spacing. The Navier-Stokes equations

$$\partial_t \rho + \nabla \cdot (\rho\mathbf{u}) = 0,$$

$$\partial_t(\rho\mathbf{u}) + \nabla \cdot (\rho\mathbf{u}\mathbf{u}) + \nabla \rho c_s^2 = \eta \nabla^2 \mathbf{u} + (\eta_v + \eta) \nabla(\nabla \cdot \mathbf{u}) \quad (8)$$

are recovered in the low velocity limit, with viscosities

$$\eta = -\rho c_s^2 \Delta t \left(\frac{1}{\lambda} + \frac{1}{2} \right) \quad \text{and} \quad \eta_v = -\rho c_s^2 \Delta t \left(\frac{2}{3\lambda_v} + \frac{1}{3} \right). \quad (9)$$

III. SOLID-FLUID BOUNDARY CONDITIONS

Boundary conditions in the lattice-Boltzmann model are straightforward to implement, even for nonplanar surfaces [1]. Solid particles are defined by a surface, which cuts some of the links between lattice nodes. Fluid particles moving along these links interact with the solid surface at boundary nodes placed halfway along the links. Thus we obtain a discrete representation of the particle surface, which becomes more and more precise as the particle gets larger (Fig. 1). In the past we have treated the lattice nodes on either side of the boundary surface in an identical fashion, so that fluid fills the whole volume of space, both inside and outside the solid particles. Because of the relatively small volume inside each particle, the interior fluid quickly relaxes to rigid-body motion, characterized by the particle velocity and angular velocity, and closely approximates a rigid body of the same mass [24]. However, at short times the inertial lag of the fluid is noticeable, and the contribution of the interior fluid to the particle force and torque reduces the stability of the particle velocity update. For these reasons we have followed the suggestion in Ref. [6], and removed the fluid from the interior of the particles. A somewhat different implementation of the same idea is described in Ref. [22].

The moving boundary condition [1] without interior fluid [6] is then implemented as follows. We take the set of fluid nodes \mathbf{r} just outside the particle surface, and for each node all the velocities \mathbf{c}_b such that $\mathbf{r} + \mathbf{c}_b\Delta t$ lies inside the particle surface. An example of a set of boundary node velocities is shown by the arrows in Fig. 1(a). Each of the corresponding population densities is then updated according to a simple

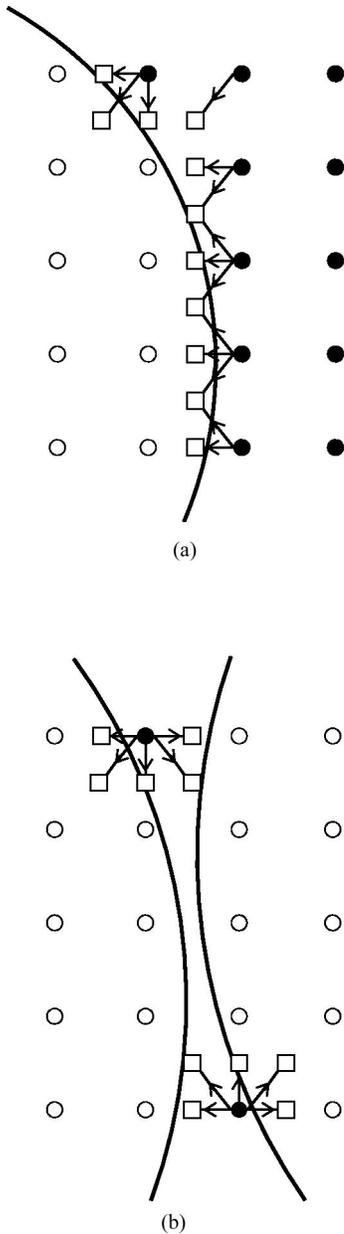


FIG. 1. Location of boundary nodes for a curved surface (a) and two surfaces in near contact (b). The velocities along links cutting the boundary surface are indicated by arrows. The locations of the boundary nodes are shown by open squares, and the fluid nodes by solid circles.

rule which takes into account the motion of the particle surface [1];

$$n_b'(\mathbf{r}, t + \Delta t) = n_b^*(\mathbf{r}, t) - \frac{2a^{c_b} \rho_0 \mathbf{u}_b \cdot \mathbf{c}_b}{c_s^2}, \quad (10)$$

where $n_b^*(\mathbf{r}, t)$ is the postcollision distribution at (\mathbf{r}, t) in the direction \mathbf{c}_b , and $\mathbf{c}_b' = -\mathbf{c}_b$. The local velocity of the particle surface,

$$\mathbf{u}_b = \mathbf{U} + \boldsymbol{\Omega} \times (\mathbf{r}_b - \mathbf{R}), \quad (11)$$

is determined by the particle velocity \mathbf{U} , angular velocity $\boldsymbol{\Omega}$, and center of mass \mathbf{R} ; $\mathbf{r}_b = \mathbf{r} + \frac{1}{2} \mathbf{c}_b \Delta t$ is the location of the boundary node. We have used the mean density ρ_0 in Eq. (10) instead of the local density $\rho(\mathbf{r}, t)$ since it simplifies the update procedure. The differences between the two methods are small, of the same order ($\rho \mathbf{u}^3$) as the error terms in the lattice-Boltzmann model. Test calculations show that even large variations in fluid density (up to 10%) have a negligible effect on the force (less than 1 part in 10^4).

As a result of the boundary node updates, momentum is exchanged locally between the fluid and the solid particle, but the combined momentum of solid and fluid is conserved. The forces exerted at the boundary nodes can be calculated from the momentum transferred in Eq. (10),

$$\mathbf{f}(\mathbf{r}_b, t + \frac{1}{2} \Delta t) = \frac{\Delta x^3}{\Delta t} \left[2n_b^*(\mathbf{r}, t) - \frac{2a^{c_b} \rho_0 \mathbf{u}_b \cdot \mathbf{c}_b}{c_s^2} \right] \mathbf{c}_b. \quad (12)$$

The particle forces and torques are then obtained by summing $\mathbf{f}(\mathbf{r}_b)$ and $\mathbf{r}_b \times \mathbf{f}(\mathbf{r}_b)$ over all the boundary nodes associated with a particular particle. It can be shown analytically that the force on a planar wall in a linear shear flow is exact [1], and several numerical examples of lattice-Boltzmann simulations of hydrodynamic interactions are given in Ref. [2].

To understand the physics of the moving boundary condition, one can imagine an ensemble of particles, moving at constant speed \mathbf{c}_b , impinging on a massive wall oriented perpendicular to the particle motion. The wall itself is moving with velocity $\mathbf{u}_b \ll \mathbf{c}_b$. The velocity of the particles after collision with the wall is $-\mathbf{c}_b + 2\mathbf{u}_b$, and the force exerted on the wall is proportional to $\mathbf{c}_b - \mathbf{u}_b$. Since the velocities in the lattice-Boltzmann model are discrete, the desired boundary condition cannot be implemented directly, but we can instead modify the density of returning particles so that the momentum transferred to the wall is the same as in the continuous velocity case. It can be seen that this implementation of the no-slip boundary condition leads to a small mass transfer across a moving solid-fluid interface. This is physically correct and arises from the discrete motion of the solid surface. Thus during a time step Δt the fluid is flowing continuously, while the solid particle is fixed in space. If the fluid cannot flow across the surface there will be large artificial pressure gradients, arising from the compression and expansion of fluid near the surface. For a uniformly moving particle, it is straightforward to show that the mass transfer across the surface in a time step Δt [Eq. (10)] is exactly recovered when the particle moves to its new position. For example, each fluid node adjacent to a planar wall has five links intersecting the wall. If the wall is advancing into the fluid with a velocity \mathbf{U} , then the mass flux across the interface [from Eq. (10)] is $\rho_0 \mathbf{U}$. Apart from small compressibility effects, this is exactly the rate at which fluid mass is absorbed by the moving wall. For sliding motion, Eq. (10) correctly predicts no net mass transfer across the interface.

Although the link-bounce-back rule is second-order accurate for planar walls oriented along lattice symmetry directions, it is only first-order accurate for channels oriented at

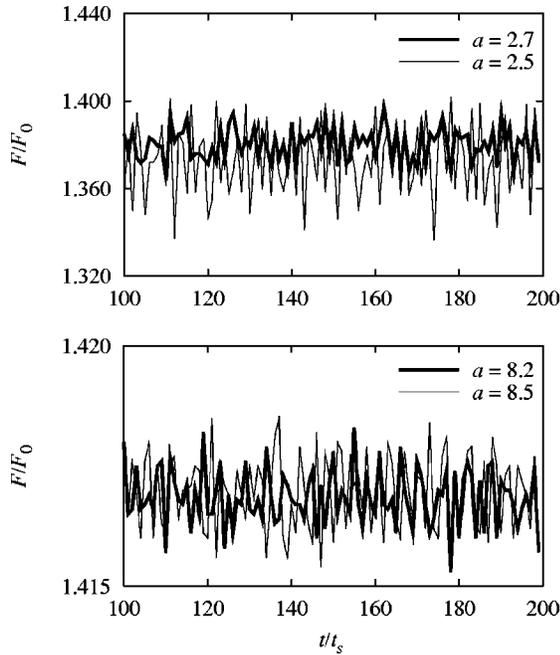


FIG. 2. The drag force F as a function of time, normalized by the drag force on an isolated sphere, $F_0 = 6\pi\eta aU$. Time is measured in units of the Stokes time, $t_s = a/U$. The particle is moving along a randomly chosen direction in a periodic unit cell with $Re = 1$.

arbitrary angles [27,28]. Thus for large channels, the hydrodynamic boundary is displaced by an amount Δ from the physical boundary, where Δ is independent of channel width but depends on the orientation of the channel with respect to the underlying lattice. Convex bodies sample a variety of boundary orientations, so that it is not possible to make an analytical determination of the displacement of the hydrodynamic boundary from the solid particle surface. Nevertheless, the displacement of the boundary can be determined numerically from simulations of flow around isolated particles. By considering the size of the particles to be the hydrodynamic radius, $a_{hy} = a + \Delta$, rather than the physical radius a , approximate second-order convergence can be obtained, even for dense suspensions [2].

The hydrodynamic radius can be determined from the drag on a fixed sphere in a pressure-driven flow [2]. Galilean invariance can be demonstrated by showing that the same force is obtained for a moving particle in a stationary fluid [2]. Since the particle samples different boundary node maps as it moves on the grid, it is important to sample different particle positions when determining the hydrodynamic radius, especially when the particle radius is small ($< 5\Delta x$). Rather than averaging over many fixed configurations, we chose to have the particle move slowly across the grid, at constant velocity, sampling different boundary node maps as it goes. The changing boundary node maps lead to fluctuations in the drag force, as shown in Fig. 2. The force has been averaged over a Stokes time t_s so that the relative fluctuations in force are comparable to the relative fluctuations in velocity of a neutrally buoyant particle in a constant force field. The force fluctuations, $\delta_F = \sqrt{(\langle F^2 \rangle - \langle F \rangle^2) / \langle F \rangle}$, are

TABLE I. Variance in the computed drag force $\delta_F = \sqrt{(\langle F^2 \rangle - \langle F \rangle^2) / \langle F \rangle}$ for a particle of radius a moving along a random orientation with respect to the grid. The results are averaged for $50 - 200t_s$, after the system reaches a steady state. The dimensionless viscosity $\nu^* = 1/6$. The numbers in brackets indicate powers of 10.

$a/\Delta x$	2.7	2.5	8.2	8.5
δ_F	5.738[-3]	1.208[-2]	4.332[-4]	5.674[-4]

of the order of 1% for particles of radius $2 - 3\Delta x$, and are considerably smaller for larger particles (Table I). More sophisticated boundary conditions have been developed using finite-volume methods [29,30] and interpolation [13,14,31]. Both methods reduce the force fluctuations by at least an order of magnitude from those observed here, but even with the simple bounce-back scheme, the fluctuations in force can be reduced by an appropriate choice of particle radius. We have noticed that fluctuations in particle force are strongly correlated with fluctuations in particle volume. Thus we choose the radius of the boundary node map so as to minimize fluctuations in particle volume for random locations on the grid. It can be seen from Table I that a two fold reduction in the force fluctuations is possible by this procedure, although for sufficiently large particles the difference is minimal. A set of optimal particle radii is given in Table II.

The bounce-back rule leads to a velocity field in the region of the boundary that is first-order accurate in the grid spacing Δx . The hydrodynamic boundary (the surface where the fluid velocity field matches the velocity of the particle) is displaced from the particle surface by a constant, Δ (Fig. 3), that depends on the viscosity of the fluid [2]. For the range of kinematic viscosities used in this work, $1/6 \leq \nu^* \leq 1/1200$, Δ varies from 0 to $0.5\Delta x$ (Table II); the dimensionless kinematic viscosity $\nu^* = \nu\Delta t/\Delta x^2$. For small particles ($a < 5\Delta x$), Δ also depends weakly on the particle radius (Table II). Although the difference between the hydrodynamic boundary and the physical boundary is small, it is important in obtaining accurate results with computationally useful particle sizes. Calibration of the hydrodynamic radius leads to approximately second-order convergence from an inherently first-order boundary condition; it will not be necessary when more accurate boundary conditions are implemented.

The hydrodynamic radii (a_{hy}) in Table II were determined from the drag force on a single sphere in a periodic cubic cell [2]. The Reynolds number in these simulations (< 0.2) was sufficiently small to have a negligible effect on the drag force. The time averaged force was used to determine the effective hydrodynamic radius for three different

TABLE II. Hydrodynamic radius a_{hy} (in units of Δx) for various fluid viscosities; a is the input particle radius.

$a/\Delta x$	1.24	2.7	4.8	6.2	8.2
$\nu^* = 1/6$	1.10	2.66	4.80	6.23	8.23
$\nu^* = 1/100$	1.42	2.92	5.04	6.45	8.46
$\nu^* = 1/1200$	1.64	3.18	5.31	6.72	8.75

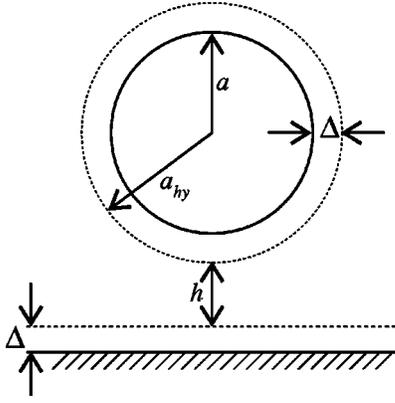


FIG. 3. Actual (solid lines) and hydrodynamic (dashed lines) surfaces for a particle settling onto a wall.

kinematic viscosities: $\nu^* = 1/6$, $\nu^* = 1/100$, and $\nu^* = 1/1200$. In each case the cell dimensions were ten times the particle radius and the corrections for periodic boundary conditions (about 40%) were made as described in Ref. [2].

The difference between the hydrodynamic radius and the input radius, $\Delta = a_{hy} - a$, is independent of particle size for radii $a > 5\Delta x$, and its magnitude increases with decreasing kinematic viscosity [24]. The kinematic viscosity $\nu^* = 1/6$ gives a hydrodynamic radius that is the same as the input radius (for sufficiently large particles), and is the natural choice for simulations at low Reynolds number. At higher Reynolds numbers a lower viscosity is necessary to maintain incompressibility [2,24], and for accurate results it is then essential to use the calibrated hydrodynamic radius.

In order to implement the hydrodynamic radius in a multiparticle suspension, all distance calculations are based on the hydrodynamic radius (as shown in Fig. 3); the input radius a is only used to determine the location of the boundary nodes. It should be noted that not all combinations of particle radius and viscosity can be used. Table II indicates that particle radii less than $3\Delta x$ cannot be used with a kinematic viscosity $\nu^* = 1/6$, since the hydrodynamic radius is then less than the input radius.

IV. PARTICLE MOTION

An explicit update of the particle velocity

$$\mathbf{U}(t + \Delta t) = \mathbf{U}(t) + \frac{\Delta t}{m} \mathbf{F}(t), \quad (13)$$

$$\mathbf{\Omega}(t + \Delta t) = \mathbf{\Omega}(t) + \frac{\Delta t}{I} \mathbf{T}(t), \quad (14)$$

has been found to be unstable [2] unless the particle radius is large or the particle mass density is much higher than the surrounding fluid. In previous work [2] the instability was reduced, but not eliminated, by averaging the forces and torques over two successive time steps. Subsequently, an implicit update of the particle velocity was proposed [32] as a means of ensuring stability. Here we present a generalized

version of that idea, which can be adapted to situations where two particles are in near contact.

The particle force and torque can be separated into a component that depends on the incoming velocity distribution and a component that depends, via \mathbf{u}_b , on the particle velocity and angular velocity [Eqs. (11) and (12)],

$$\mathbf{F} = \mathbf{F}_0 - \boldsymbol{\zeta}^{\text{FU}} \cdot \mathbf{U} - \boldsymbol{\zeta}^{\text{F}\Omega} \cdot \mathbf{\Omega}, \quad (15)$$

$$\mathbf{T} = \mathbf{T}_0 - \boldsymbol{\zeta}^{\text{TU}} \cdot \mathbf{U} - \boldsymbol{\zeta}^{\text{T}\Omega} \cdot \mathbf{\Omega}. \quad (16)$$

The velocity independent forces and torques are determined at the half-time step

$$\mathbf{F}_0(t + \frac{1}{2}\Delta t) = \frac{\Delta x^3}{\Delta t} \sum_b 2n_b^*(\mathbf{r}, t) \mathbf{c}_b, \quad (17)$$

$$\mathbf{T}_0(t + \frac{1}{2}\Delta t) = \frac{\Delta x^3}{\Delta t} \sum_b 2n_b^*(\mathbf{r}, t) (\mathbf{r}_b \times \mathbf{c}_b), \quad (18)$$

where the sum is over all the boundary nodes b describing the particle surface, with \mathbf{c}_b representing the velocity associated with the boundary node b and pointing towards the particle center. The matrices

$$\boldsymbol{\zeta}^{\text{FU}} = \frac{2\rho_0 \Delta x^3}{c_s^2 \Delta t} \sum_b a^{cb} \mathbf{c}_b \mathbf{c}_b, \quad (19)$$

$$\boldsymbol{\zeta}^{\text{F}\Omega} = \frac{2\rho_0 \Delta x^3}{c_s^2 \Delta t} \sum_b a^{cb} \mathbf{c}_b (\mathbf{r}_b \times \mathbf{c}_b), \quad (20)$$

$$\boldsymbol{\zeta}^{\text{TU}} = \frac{2\rho_0 \Delta x^3}{c_s^2 \Delta t} \sum_b a^{cb} (\mathbf{r}_b \times \mathbf{c}_b) \mathbf{c}_b, \quad (21)$$

$$\boldsymbol{\zeta}^{\text{T}\Omega} = \frac{2\rho_0 \Delta x^3}{c_s^2 \Delta t} \sum_b a^{cb} (\mathbf{r}_b \times \mathbf{c}_b) (\mathbf{r}_b \times \mathbf{c}_b) \quad (22)$$

are high-frequency friction coefficients, and describe the instantaneous force on a particle in response to a sudden change in velocity.

The magnitude of the friction coefficients can be readily estimated, thereby establishing bounds on the stability of an explicit update. Apart from irregularities in the discretized surface, $\boldsymbol{\zeta}^{\text{FU}}$ and $\boldsymbol{\zeta}^{\text{T}\Omega}$ are diagonal matrices, while $\boldsymbol{\zeta}^{\text{F}\Omega} = \boldsymbol{\zeta}^{\text{TU}} = 0$. For a node adjacent to a planar wall $\sum_i a^{ci} c_i^2 = \frac{5}{18} c^2$, where the sum is over the five directions that cross the wall. The number of such nodes is approximately $4\pi a^2 / \Delta x^2$, so that

$$\zeta^{\text{FU}} \sim \frac{20\pi}{9} \frac{\rho_0 \Delta x a^2}{\Delta t}. \quad (23)$$

Similarly,

$$\zeta^{\text{T}\Omega} \sim \frac{8\pi}{9} \frac{\rho_0 \Delta x a^4}{\Delta t}. \quad (24)$$

These estimates of the translational and rotational friction coefficients are within 20% and 50% of numerically computed values, respectively. The stability criterion for an explicit update $\zeta^{\text{FU}}\Delta t/m < 2$ then reduces to a simple condition involving the particle radius and mass density,

$$\frac{5}{3} \frac{\rho_f \Delta x}{\rho_s a} < 2. \quad (25)$$

The corresponding condition for the torque leads to the same stability criterion

$$\frac{\zeta^{\text{T}\Omega}\Delta t}{I} \sim \frac{5}{3} \frac{\rho_f \Delta x}{\rho_s a} < 2, \quad (26)$$

whereas with interior fluid the numerical factors were six and ten, respectively [2], showing that interior fluid destabilizes an explicit update.

The friction coefficients in Eqs. (15) and (16) are essentially constant, fluctuating slowly in time as the particle moves on the underlying grid; thus the particle velocities can be updated assuming these friction matrices are constant. The equations of motion can then be written in matrix form as

$$\begin{aligned} \begin{bmatrix} \mathbf{U}(t+\Delta t) \\ \boldsymbol{\Omega}(t+\Delta t) \end{bmatrix} &= \begin{bmatrix} \mathbf{U}(t) \\ \boldsymbol{\Omega}(t) \end{bmatrix} + \begin{bmatrix} \frac{m}{\Delta t} \mathbf{1} + \alpha \boldsymbol{\zeta}^{\text{FU}} & \alpha \boldsymbol{\zeta}^{\text{F}\Omega} \\ \alpha \boldsymbol{\zeta}^{\text{TU}} & \frac{I}{\Delta t} \mathbf{1} + \alpha \boldsymbol{\zeta}^{\text{T}\Omega} \end{bmatrix}^{-1} \\ &\times \begin{bmatrix} \mathbf{F}_0 - \boldsymbol{\zeta}^{\text{FU}}\mathbf{U}(t) - \boldsymbol{\zeta}^{\text{F}\Omega}\boldsymbol{\Omega}(t) \\ \mathbf{T}_0 - \boldsymbol{\zeta}^{\text{TU}}\mathbf{U}(t) - \boldsymbol{\zeta}^{\text{T}\Omega}\boldsymbol{\Omega}(t) \end{bmatrix}, \end{aligned} \quad (27)$$

where α is a parameter that controls the degree of implicitness. An explicit update [2] corresponds to $\alpha=0$, an implicit update [32] corresponds to $\alpha=1$, and a second-order semi-implicit update corresponds to $\alpha=\frac{1}{2}$. The explicit, implicit, and semi-implicit updates evaluate the velocity-dependent force at t , $t+\Delta t$, and $t+\frac{1}{2}\Delta t$, respectively. In practice we find only small differences between semi-implicit and fully implicit methods and we use the fully implicit method ($\alpha=1$) in this work. The boundary node map is updated infrequently (every 10–100 time steps) and the 6×6 matrix inversion need only be done when the map is updated. We note that in the limit of $\zeta^{\text{FU}}\Delta t/m \gg 1$ only the fully implicit ($\alpha=1$) version of Eq. (27) reduces to the steady state force and torque balance, $\mathbf{F}_0 - \boldsymbol{\zeta}^{\text{FU}}\mathbf{U}(t+\Delta t) = \mathbf{T}_0 - \boldsymbol{\zeta}^{\text{T}\Omega}\boldsymbol{\Omega}(t+\Delta t) = 0$. The semi-implicit method ($\alpha=\frac{1}{2}$) produces an oscillating solution and the explicit method ($\alpha=0$) a diverging solution.

An implicit update of the particle velocities requires two passes through the boundary nodes. On the first pass the population densities are used to calculate \mathbf{F}_0 and \mathbf{T}_0 . The implicit equations [Eq. (27)] are then solved for $\mathbf{U}(t+\Delta t)$ and $\boldsymbol{\Omega}(t+\Delta t)$ for the given implicit parameter α . These velocities are then used to calculate the new population densities in a second sweep through the boundary nodes. The computational overhead incurred by the boundary node up-

dates is typically less than 100%, even at high concentrations (see, Ref. [24] for a detailed discussion of computational performance).

The key drawback to excluding fluid from the interior of the particle is that the update of the boundary node map introduces stronger discontinuities in the flow field than when the particle interior is filled with fluid. If the interior fluid is to be excluded, then fluid must be removed when the movement of the particle causes a node to be transferred to the interior region. Moreover, interior nodes are exposed by particle motion and must then be filled with fluid with an appropriate velocity distribution. It is essential to make these changes as smoothly as possible to avoid strong pressure pulses in the fluid. When an interior node is uncovered by the motion of the particle, its velocity distribution is taken to be the local equilibrium [Eq. (6)] with a mass density ρ_0 and a velocity given by the local velocity when it last resided inside the particle (assuming rigid-body motion). We note that it is also possible to use interpolation to set the nonequilibrium distribution [31], but we have not yet implemented this idea although it has been shown to lead to very smooth particle trajectories. The momentum transferred by this process is balanced by an appropriate force and torque, which are added to the particle \mathbf{F}_0 and \mathbf{T}_0 at the next time step. When a fluid node is covered by a particle, its momentum is similarly absorbed in the particle force and torque. In addition the excess fluid mass $\Delta M = (\rho - \rho_0)\Delta x^3$ is uniformly redistributed among the boundary nodes at the next time step, in a similar fashion to when particles are in near contact (see Sec. V A). In this way the global mass density of the fluid remains constant, even though the volume occupied by the particles fluctuate as they move on the grid.

V. LUBRICATION INTERACTIONS

A. Surfaces near contact

When two particle surfaces come within one grid spacing, fluid nodes are excluded from regions between the solid surfaces [Fig 1(b)], leading to a loss of mass conservation. This happens because boundary updates at each link produces a mass transfer $(2a^c b \rho_0 \mathbf{u}_b \mathbf{c}_b / c_s^2) \Delta x^3$ across the solid-fluid interface, which is necessary to accommodate the discrete motion of the particle surface (see Sec. III). The total mass transfer in or out of an isolated particle,

$$\Delta M = - \frac{2\Delta x^3 \rho_0}{c_s^2} \left[\mathbf{U} \cdot \sum_b a^c b \mathbf{c}_b + \boldsymbol{\Omega} \cdot \sum_b a^c b \mathbf{r}_b \times \mathbf{c}_b \right] = 0, \quad (28)$$

regardless of the particle's size or shape.

Although the sums $\sum_b a^c b \mathbf{c}_b$ and $\sum_b a^c b \mathbf{r}_b \times \mathbf{c}_b$ are zero for any closed surface, when two particles are close to contact some of the boundary nodes are missing and the surfaces are no longer closed. In this case $\Delta M \neq 0$ and mass conservation is no longer ensured. Two particles that remain in close proximity never reach a steady state, no matter how slowly they move, since fluid is constantly being added or removed, depending on the particle positions and velocities. If the two

particles move as a rigid body mass conservation is restored, but in general this is not the case.

The accumulation or loss of mass occurs slowly, and in many dynamical simulations it fluctuates with changing particle configuration but shows no long-term drift. However, we enforce mass conservation, particle-by-particle, by redistributing the excess mass among the boundary nodes (cf. Eq. (10))

$$n_{i'}(\mathbf{r}, t + \Delta t) = n_i^*(\mathbf{r}, t) - \frac{2a^{c_b}\rho_0\mathbf{u}_b \cdot \mathbf{c}_b}{c_s^2} - a^{c_b}\rho_0 \frac{\Delta M}{A}, \quad (29)$$

where $A = \Delta x^3 \sum_b a^{c_b} \rho_0$.

The force and torque arising from this redistribution of mass are small, but not exactly zero;

$$\Delta \mathbf{F} = \frac{\Delta x^3 \rho_0}{\Delta t} \left[-\frac{\Delta M}{A} \sum_b a^{c_b} \mathbf{c}_b \right], \quad (30)$$

$$\Delta \mathbf{T} = \frac{\Delta x^3 \rho_0}{\Delta t} \left[-\frac{\Delta M}{A} \sum_b a^{c_b} \mathbf{r}_b \times \mathbf{c}_b \right]. \quad (31)$$

They can be compactly included by a redefinition of the friction coefficients, Eqs. (19)–(22), replacing \mathbf{c}_b and $\mathbf{r}_b \times \mathbf{c}_b$ by their deviation from the mean,

$$\overline{\mathbf{c}_b} = \frac{\sum_b a^{c_b} \mathbf{c}_b}{\sum_b a^{c_b}} \quad \text{and} \quad \overline{\mathbf{r}_b \times \mathbf{c}_b} = \frac{\sum_b a^{c_b} \mathbf{r}_b \times \mathbf{c}_b}{\sum_b a^{c_b}}, \quad (32)$$

so that $\mathbf{c}_b \rightarrow \mathbf{c}_b - \overline{\mathbf{c}_b}$ and $\mathbf{r}_b \times \mathbf{c}_b \rightarrow \mathbf{r}_b \times \mathbf{c}_b - \overline{\mathbf{r}_b \times \mathbf{c}_b}$. Then the force and torque are correctly calculated from Eqs. (15) and (16), even when mass is redistributed.

B. Lubrication forces

When two particles are in near contact, the fluid flow in the gap cannot be resolved. For particle sizes that are typically used in multiparticle simulations ($a < 5\Delta x$), the lubrication breakdown in the calculation of the hydrodynamic interaction occurs at gaps less than $0.1a$. However, in some flows, notably the shearing of a dense suspension, qualitatively important physics occurs at smaller separations, typically down to $0.01a$. Here we describe a method to implement lubrication corrections into a lattice-Boltzmann simulation.

For particles close to contact, the lubrication force, torque, and stresslet can be calculated from a sum of pairwise-additive contributions [18], and if we consider only singular terms, they can be calculated from the particle velocities alone [33]. In the grand-resistance-matrix formulation [34]

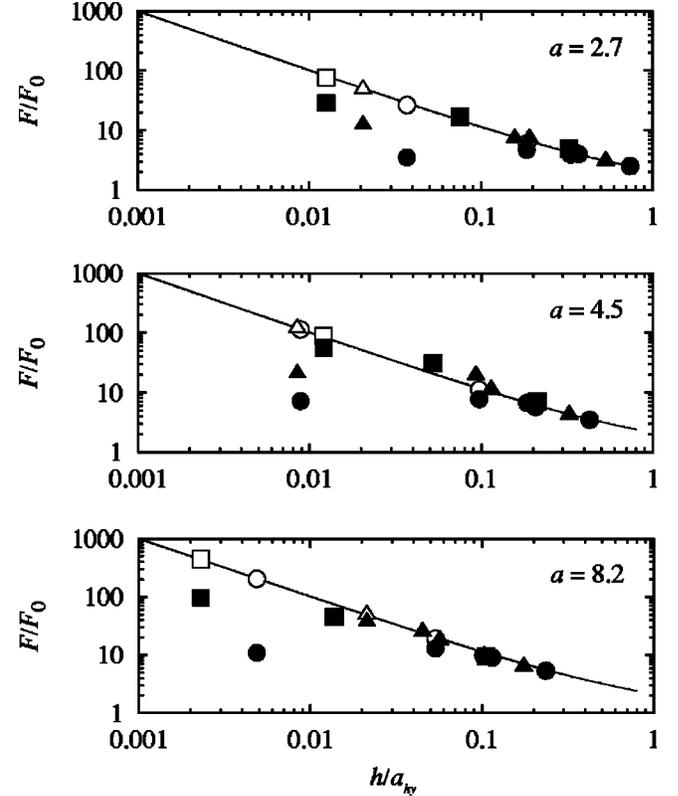


FIG. 4. Normal force on a particle of input radius a settling onto a horizontal planar surface at zero Reynolds number. The drag force determined by lattice-Boltzmann simulation, normalized by the drag on an isolated sphere $F_0 = 6\pi\eta a_{hy}U$, is indicated by solid symbols: $\nu^* = 1/6$ (circles), $\nu^* = 1/100$ (triangles), and $\nu^* = 1/1200$ (squares). The hydrodynamic radii (a_{hy}), taken from Table II, were used to determine the gap h between the particle and the wall (see Fig. 3). Results including the lubrication correction described in Sec. V C are shown by the open symbols. The solid line is the theoretical result from lubrication theory [35].

$$\begin{bmatrix} \mathbf{F}_1 \\ \mathbf{T}_1 \\ \mathbf{T}_2 \\ \mathbf{S}_1 \\ \mathbf{S}_2 \end{bmatrix} = - \begin{bmatrix} \mathbf{A}_{11} & -\mathbf{B}_{11} & \mathbf{B}_{22} \\ \mathbf{B}_{11} & \mathbf{C}_{11} & \mathbf{C}_{12} \\ -\mathbf{B}_{22} & \mathbf{C}_{12} & \mathbf{C}_{22} \\ \mathbf{G}_{11} & \mathbf{H}_{11} & \mathbf{H}_{12} \\ -\mathbf{G}_{22} & \mathbf{H}_{21} & \mathbf{H}_{22} \end{bmatrix} \begin{bmatrix} \mathbf{U}_{12} \\ \boldsymbol{\Omega}_1 \\ \boldsymbol{\Omega}_2 \end{bmatrix}, \quad (33)$$

where $\mathbf{F}_2 = -\mathbf{F}_1$, $\mathbf{U}_{12} = \mathbf{U}_1 - \mathbf{U}_2$, and the friction matrices are as given in Ref. [34]. We have made full use of the symmetries inherent in the friction matrices, but without assuming that the particle radii are the same. Most importantly, the external flow field does not enter into the lubrication calculation, which removes the need for estimates of the local flow field.

We have noted in previous lattice-Boltzmann simulations [21,24] that the calculated forces follow the Stokes flow results down to a fixed separation, around $0.5\Delta x$, and remain roughly constant thereafter. The solid symbols in Fig. 4, for example, show this behavior for the normal force between a spherical particle and a plane wall. This suggests a lubrication

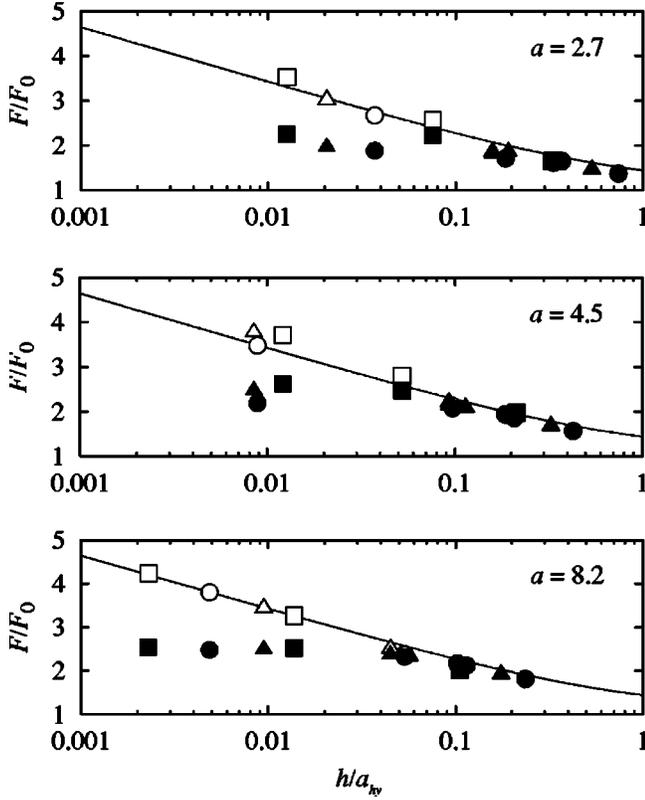


FIG. 5. Tangential force on a particle settling next to a vertical planar surface at zero Reynolds number. The drag force determined by lattice-Boltzmann simulation, normalized by the drag on an isolated sphere $F_0 = 6\pi\eta a_{hy}U$, is indicated by solid symbols. Results including lubrication are shown by the open symbols.

tion correction of the form of a difference between the lubrication force at h and the force at some cut off distance h_N ; i.e.,

$$\mathbf{F}_{lub} = -6\pi\eta \frac{a_1^2 a_2^2}{(a_1 + a_2)^2} \left(\frac{1}{h} - \frac{1}{h_N} \right) \mathbf{U}_{12} \cdot \hat{\mathbf{R}}_{12}, \quad h < h_N$$

$$= 0, \quad h > h_N, \quad (34)$$

where $\mathbf{U}_{12} = \mathbf{U}_1 - \mathbf{U}_2$, $h = |\mathbf{R}_{12}| - a_1 - a_2$ is the gap between the two surfaces, and the unit vector $\hat{\mathbf{R}}_{12} = \mathbf{R}_{12}/|\mathbf{R}_{12}|$.

The friction coefficients in Eq. (33) are all products of a scalar function of the gap between the particles, either $1/h$ or $\ln(1/h)$, multiplied by a polynomial of the unit vector connecting the particle centers [34]. For two spheres of arbitrary size, there are a total of 15 independent scalar coefficients, which fall into three classes. Again using the notation of Ref. [34] these are $X_{11}^A, X_{11}^G, X_{22}^G$ (normal force); $Y_{11}^A, Y_{11}^B, Y_{22}^B, Y_{11}^G, Y_{22}^G$ (tangential force); and $Y_{11}^C, Y_{12}^C, Y_{22}^C, Y_{11}^H, Y_{12}^H, Y_{21}^H, Y_{22}^H$ (rotation). We implement our lubrication correction by calculating a modified form of each scalar coefficient as in Eq. (34); for example,

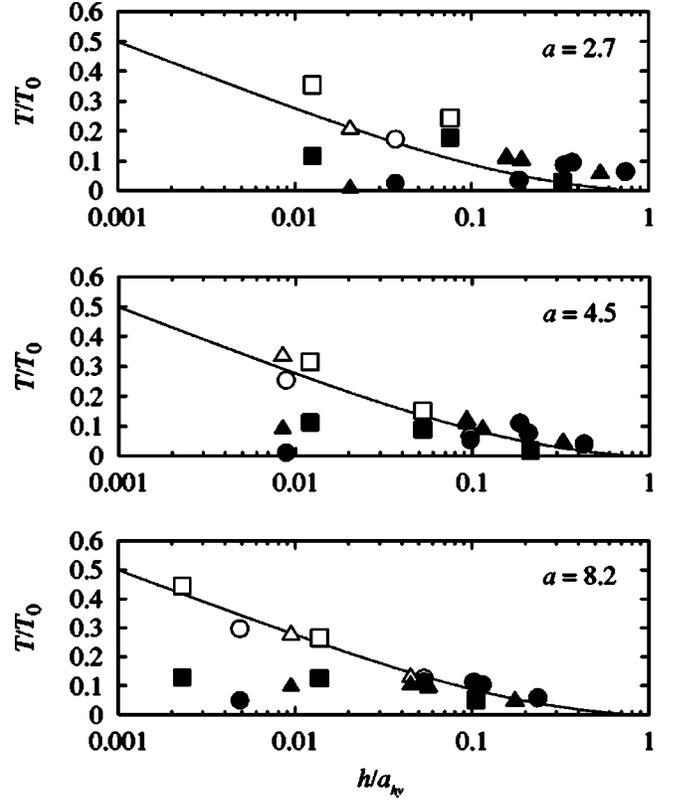


FIG. 6. Torque on a particle settling next to a vertical planar surface at zero Reynolds number. The torque determined by lattice-Boltzmann simulation, normalized by the torque on an isolated sphere $T_0 = 8\pi\eta a_{hy}^2 U$, is indicated by solid symbols. Results including lubrication are shown by the open symbols.

$$\tilde{X}_{11}^A(h) = X_{11}^A(h) - X_{11}^A(h_N) \quad h < h_N,$$

$$\tilde{X}_{11}^A(h) = 0 \quad h > h_N, \quad (35)$$

which vanishes at the cut-off distance $h = h_N$. We allow for different cut off distances for each of the three types of lubrication interaction.

C. Particle wall lubrication

The hydrodynamic interactions between two moving surfaces have been calculated for the simplest geometry, namely, a spherical particle adjacent to a planar wall. We used three different particle sizes, with input radii $a/\Delta x = 2.7, 4.5$, and 8.2 , chosen to minimize volume fluctuations (see Sec. III) with the exception of the results for $a/\Delta x = 4.5$, which were generated before the optimum radius ($4.8\Delta x$) was determined. The hydrodynamic radii, $a_{hy}(a, \nu^*)$, that are used to determine the positions of the lubricating surfaces were taken from Table II. The location of the planar wall was shifted by $\Delta(\nu^*)$, corresponding to the $a \rightarrow \infty$ limit in Table II (see Fig. 3). In this way we ensure that the lubricating surfaces are in the same position as the hydrodynamic boundaries in the lattice-Boltzmann simulations. The unit cell is periodic in four directions with a top and bottom wall, and cell length of ten times the particle

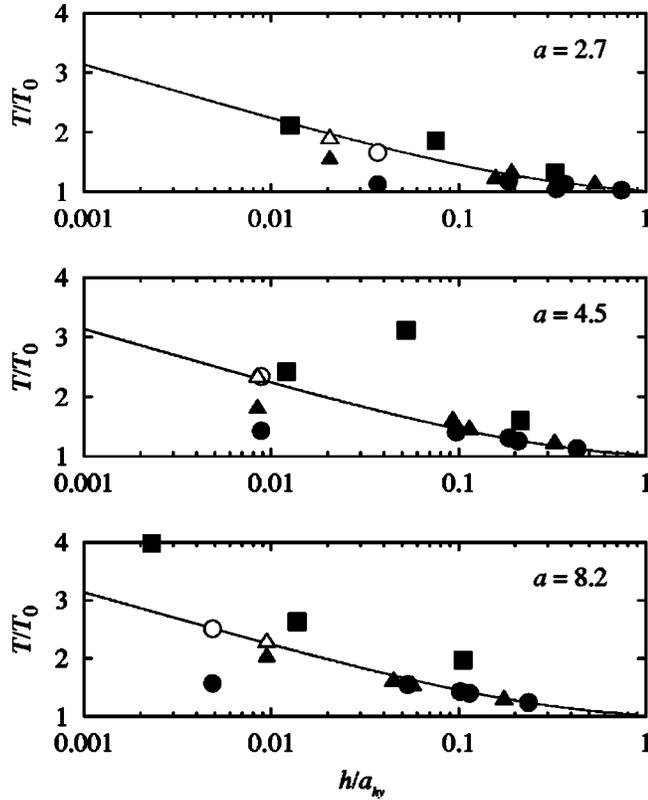


FIG. 7. Torque on a particle rotating next to a vertical planar surface at zero Reynolds number. The torque determined by lattice-Boltzmann simulation, normalized by the torque on an isolated sphere $T_0 = 8\pi\eta a_{hy}^3\Omega$, is indicated by solid symbols. Results including lubrication are shown by the open symbols.

radius, which is sufficiently large that the effects of periodic images were negligible. The simulation determines the steady state force and torque on the particle for a given velocity and angular velocity, which was then used to calculate the friction coefficient as a function of the gap h from the wall. Simulation results for the frictional forces and torques are shown in Figs. 4–7 for three different fluid viscosities $\nu^* = 1/6, 1/100, \text{ and } 1/1200$.

The normal force shows the trend discussed in Sec. V B for each particle size and fluid viscosity (Fig. 4). The simulated force (solid symbols) follows the exact result (solid line) down to an interparticle gap, $h_N < \Delta x$, that is independent of particle size. For larger particles the lattice-Boltzmann method captures progressively more of the lubrication force, but even for $a = 8.2\Delta x$ there are noticeable

TABLE III. Lubrication ranges for various kinematic viscosities, determined for a sphere of radius $a = 8.2\Delta x$. The ranges are determined separately for the normal h_N tangential h_T and rotational h_R motions.

	$h_N/\Delta x$	$h_T/\Delta x$	$h_R/\Delta x$
$\nu^* = 1/6$	0.67	0.50	0.43
$\nu^* = 1/100$	0.24	0.50	0.15
$\nu^* = 1/1200$	0.10	0.50	0.00

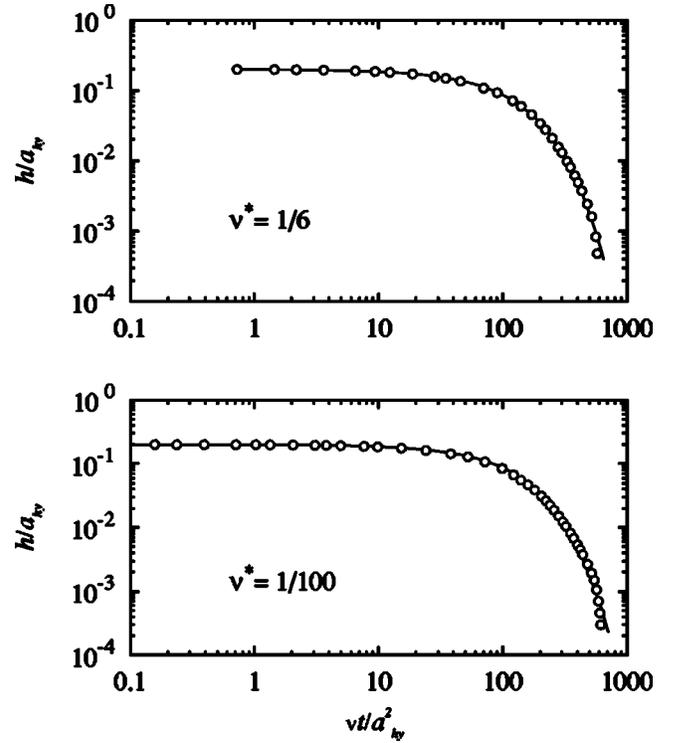


FIG. 8. Settling of a sphere ($a = 4.8$) onto a horizontal wall. The gap between the particle surface and wall, h , relative to the hydrodynamic radius, is plotted as a function of the nondimensional time (open circles). The solid line is based on the frictional force calculated from lubrication theory [35].

deviations for $h/a_{hy} < 0.01$. The simulation reproduces more of the lubrication force at smaller viscosities because the shift in the hydrodynamic radius delays the contact of the particle surfaces. The data obtained for a particle radius of $8.2\Delta x$ was used to determine the lubrication cutoff $h_N(\nu^*)$ for each viscosity, and the numerical values are recorded in Table III. These lubrication corrections bring the simulated normal force into agreement with theory for all the particle sizes, interparticle gaps, and fluid viscosities studied (open symbols in Fig. 4). The corresponding result for the force and torque on a sphere sliding along the wall is shown in Figs. 5 and 6. Again we see that the lubrication correction gives consistently accurate forces and torques, though not quite as accurate as the normal force. The lubrication ranges for tangential motion were found to be independent of the fluid viscosity, as shown in Table III. We also noticed that the reciprocal relations are obeyed; the force on a rotating sphere is similar to the data in Fig. 6. The calculated torque on a rotating sphere (Fig. 7) is in agreement with theory for the higher viscosities $\nu^* = 1/6$ and $\nu^* = 1/100$, but not at the lowest viscosity $\nu^* = 1/1200$. Here the lattice-Boltzmann method over predicts the torque on a rotating sphere by 20–30%. We think that the error is caused by the large difference between the hydrodynamic and input radii ($\Delta = 0.55\Delta x$) and it implies that viscosities ν^* less than 0.01 are not suitable for suspension simulations, at least with bounce-back boundary conditions. In practice this is not a serious limitation: a viscosity $\nu^* = 0.01$ allows simulations with a Reynolds number up to ten per grid point (with a

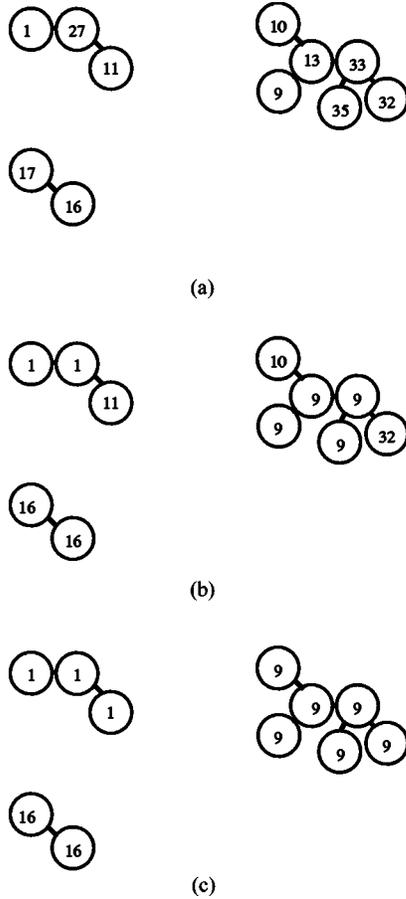


FIG. 9. Illustration of the algorithm to determine the list of clusters.

Mach number ~ 0.1), which is at or beyond the limit of resolution of the flow. In other words, there is little practical value in viscosities less than 0.01.

Finally, we examined the dynamic motion of a particle ($a=4.8\Delta x$) settling onto a solid wall (Fig. 8). The lubrication force was calculated using the ranges given in Table III. The particle was given a finite mass and placed with an initial gap of $h=0.2a_{hy}$ between the particle and wall. The simulations were performed at a Reynolds number Re

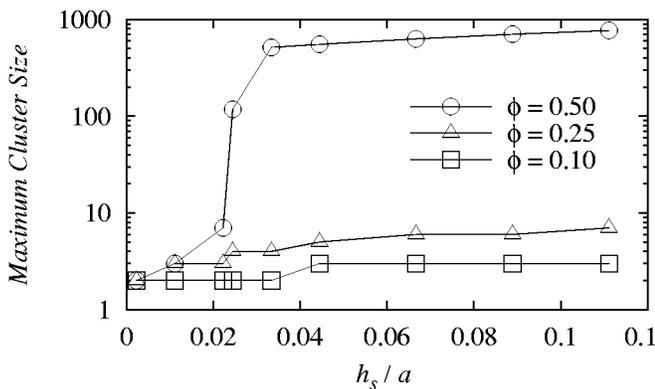


FIG. 10. The maximum cluster size as a function of the cluster cutoff gap h_s/a at varying volume fractions, ϕ .

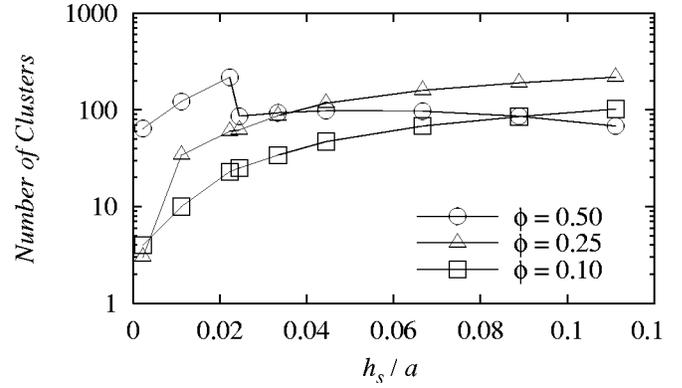


FIG. 11. Number of clusters as a function of the cluster cutoff gap h_s/a at varying volume fractions, ϕ .

~ 0.02 by applying a constant force to the particle. The results show the expected exponential decay of the gap between the particle and wall over time, in quantitative agreement with lubrication theory [35] (shown by the solid line).

D. Cluster implicit method

The lubrication forces complicate the update of the particle velocity because they involve interactions between many particles, especially at higher concentrations. For simplicity we update the particle velocities in two steps; first the lattice-Boltzmann forces and torques [Eq. (27)], followed by the lubrication forces. The overall procedure is still first order accurate, but the lubrication forces can cause instabilities whenever the particles are in near contact. The instability is driven by the normal forces, and the stability criteria

$$\frac{\xi \Delta t}{m} = \frac{6 \pi \eta a^2 \Delta t}{\frac{4}{3} \pi \rho_s a^3 h} = \frac{9}{2} \frac{\eta \Delta t}{\rho_s a h} < 2 \quad (36)$$

is violated when h is less than $\sim 0.1\Delta x$.

It is impractical to solve all the equations implicitly, so we implemented an algorithm which uses an implicit update only where necessary. In our simulations we used the conservative criteria $\xi \Delta t/m < 0.1$. Schematically, we solve the coupled differential equations

$$\dot{\mathbf{x}} = -\mathbf{A} \cdot \mathbf{x} + \mathbf{b} \quad (37)$$

by splitting the dissipative matrix \mathbf{A} into regular and singular components, $\mathbf{A} = \mathbf{A}^R + \mathbf{A}^S$. In our context \mathbf{A}^S only contains components of the normal friction coefficient when the gap between particles is less than the stability cutoff, h_s , determined from Eq. 36. Thus \mathbf{A}^R contains all the nonzero components of the lubrication correction but with the interparticle separation in the normal force regularized by h_s so that the larger of h_{ij} and h_s is used to calculate the force. The remaining normal force is included in \mathbf{A}^S , with the form of Eq. (34), but with h_N replaced by h_s . Using a mixed explicit-implicit differencing,

$$\frac{\mathbf{x}(t + \Delta t) - \mathbf{x}(t)}{\Delta t} = -\mathbf{A}^R \cdot \mathbf{x}(t) - \mathbf{A}^S \cdot \mathbf{x}(t + \Delta t) + \mathbf{b}, \quad (38)$$

we obtain the first-order update

$$(\mathbf{1} + \mathbf{A}^S \Delta t) \cdot \mathbf{x}(t + \Delta t) = \mathbf{x}(t) - \mathbf{A}^R \Delta t \cdot \mathbf{x}(t) + \mathbf{b} \Delta t. \quad (39)$$

The important point is that, by a suitable relabeling of the particle indices, \mathbf{A}^S can be cast into a block diagonal form, with the potential for an enormous reduction in the computation time for the matrix inversion. The relabeling is achieved by a cluster analysis. First, all pairs of particles that are closer than the stability cutoff are identified, and a list is made of all such pairs. An illustration is shown in Fig. 9(a), where pairs of particles with separations less than h_s are indicated by the solid lines. The cluster labels are initialized to the particle index; each particle is then relabeled by giving it the smallest label of all the particles it is connected to. After one pass, the labels are as shown in Fig. 9(b) and after two passes three distinct clusters have been identified, each with a unique label [Fig. 9(c)]. The algorithm stops when no further relabeling takes place. Although more efficient schemes are possible, this simple scheme is more than adequate for our purposes. Once the clusters have been identified, the implicit equations can be solved for each cluster. We use conjugate gradients to exploit the sparseness of \mathbf{A}^S , which is extensive even within each diagonal block.

The computational cost of the cluster-implicit algorithm depends primarily on the maximum cluster size, which is shown in Fig. 10 as a function of the cluster cutoff gap h_s . A random distribution of 1000 particles was sampled in a periodic box at volume fractions 0.10, 0.25, and 0.50. At low and moderate volume fractions the cluster size is only weakly dependent on h_s/a , ranging from 2–7, and clusters of this size impose a negligible computational overhead. However, at high volume fractions there is a percolation threshold at $h_s/a \sim 0.02$ beyond which a single cluster more or less spans the whole volume. In this case the cluster will grow to encompass almost all the particles in the system. Thus at high densities computational efficiency requires that $h_s/a < 0.02$. When combined with the stability criteria, which implies

$h_s/a \approx 1/a^2$, we find a minimum radius of $a = 10\Delta x$ to keep $\xi \Delta t/m < 0.5$. A simulation of several hundred such particles is possible on a personal computer or desktop workstation.

In Fig. 11 we show the corresponding number of clusters. In general there is a steep rise in the number of clusters with increasing h_s/a , leveling off to around 100 clusters. The sharp drop in the number of clusters at the highest volume fraction is associated with the percolation transition, as seen in Fig. 10.

VI. CONCLUSION

In this work we have described and tested several extensions to the lattice-Boltzmann method for particle suspensions, which enable reasonably accurate force calculations to be made even for particles in near contact. In particular, we have shown how to deal with problems of mass conservation when two particles are in near contact, and how to account for the lubrication forces between closely spaced particles. Numerical tests show that the forces and torques between a particle and a plane wall can be computed to within a few percent of the exact result for Stokes flow. We note that the torque on a rotating sphere adjacent to a plane wall is seriously in error (30%) when the fluid viscosity is very small (1/1200). This suggests that the calibration procedure may break down when the hydrodynamic boundary is displaced by more than $\Delta x/2$ from the physical one.

Inclusion of the lubrication forces leads to large forces and stiff differential equations for the particle velocities. We have developed a mixed explicit-implicit velocity update, which minimizes the number of linear equations that must be solved, while maintaining absolute stability.

ACKNOWLEDGMENT

This work was supported by the American Chemical Society Petroleum Research Fund (34142-AC9).

-
- [1] A.J.C. Ladd, *J. Fluid Mech.* **271**, 285 (1994).
 - [2] A.J.C. Ladd, *J. Fluid Mech.* **271**, 311 (1994).
 - [3] A.J.C. Ladd, H. Gang, J.X. Zhu, and D.A. Weitz, *Phys. Rev. Lett.* **74**, 318 (1995).
 - [4] A.J.C. Ladd, *Phys. Rev. Lett.* **76**, 1392 (1996).
 - [5] P.N. Segré, O.P. Behrend, and P.N. Pusey, *Phys. Rev. E* **52**, 5070 (1995).
 - [6] C.K. Aidun, Y.N. Lu, and E. Ding, *J. Fluid Mech.* **373**, 287 (1998).
 - [7] D.W. Qi, *J. Fluid Mech.* **385**, 41 (1999).
 - [8] P. Raskinmäki, A. Shakib-Manesh, A. Koponen, A. Jäsberg, M. Kataja, and J. Timonen, *Comput. Phys. Commun.* **129**, 185 (2000).
 - [9] A.J.C. Ladd, *Phys. Rev. Lett.* **88**, 048301 (2002).
 - [10] J.F. Brady, *J. Chem. Phys.* **99**, 567 (1993).
 - [11] J.R. Melrose and R.C. Ball, *Europhys. Lett.* **32**, 535 (1995).
 - [12] R. Mei and W. Shyy, *J. Comput. Phys.* **143**, 426 (1998).
 - [13] O. Filippova and D. Hänel, *J. Comput. Phys.* **147**, 219 (1998).
 - [14] R.W. Mei, L.S. Luo, and W. Shyy, *J. Comput. Phys.* **155**, 307 (1999).
 - [15] C.K. Ghaddar, *Phys. Fluids* **7**, 2563 (1995).
 - [16] R. Glowinski, T.W. Pan, T.I. Hesla, D.D. Joseph, and J. Périaux, *J. Comput. Phys.* **169**, 363 (2001).
 - [17] S.O. Unverdi and G. Tryggvason, *J. Comput. Phys.* **100**, 295 (1992).
 - [18] J.F. Brady and G. Bossis, *Annu. Rev. Fluid Mech.* **20**, 111 (1988).
 - [19] A.J.C. Ladd, *J. Chem. Phys.* **93**, 3484 (1990).
 - [20] B. Cichocki and B.U. Felderhof, *J. Chem. Phys.* **89**, 1049 (1988).
 - [21] A.J.C. Ladd, *Phys. Fluids* **9**, 491 (1997).
 - [22] M.W. Heemels, M.H.J. Hagen, and C.P. Lowe, *J. Comput. Phys.* **164**, 48 (2000).
 - [23] F. Higuera, S. Succi, and R. Benzi, *Europhys. Lett.* **9**, 345 (1989).
 - [24] A.J.C. Ladd and R. Verberg, *J. Stat. Phys.* **104**, 1191 (2001).

- [25] Y.H. Qian, D. d'Humières, and P. Lallemand, *Europhys. Lett.* **17**, 479 (1992).
- [26] U. Frisch, D. d'Humières, B. Hasslacher, P. Lallemand, Y. Pomeau, and J.-P. Rivet, *Complex Syst.* **1**, 649 (1987).
- [27] I. Ginzbourg and P.M. Adler, *J. Phys. II* **4**, 191 (1994).
- [28] I. Ginzbourg and D. d'Humières, *J. Stat. Phys.* **84**, 927 (1996).
- [29] H.D. Chen, C. Teixeira, and K. Molvig, *Int. J. Mod. Phys. C* **9**, 1281 (1998).
- [30] H. Chen, *Phys. Rev. E* **58**, 3955 (1998).
- [31] M. Bouzidi, M. Firdaouss, and P. Lallemand, *Phys. Fluids* **13**, 3452 (2001).
- [32] C.P. Lowe, D. Frenkel, and A.J. Masters, *J. Chem. Phys.* **103**, 1582 (1995).
- [33] I.L. Claeys and J.F. Brady, *PCH, PhysicoChem. Hydrodyn.* **11**, 261 (1989).
- [34] S. Kim and S. J. Karrila, *Microhydrodynamics: Principles and Selected Applications* (Butterworth-Heinemann, Boston, MA, 1991).
- [35] B. Cichocki and R.B. Jones, *Physica A* **258**, 273 (1998).