An extrapolation-based boundary treatment for using the lattice Boltzmann method to simulate fluid-particle interaction near a wall

Cheng-Hsien Lee, Zhenhua Huang and Yee-Meng Chiew

Department of Civil and Environmental Engineering, National University of Singapore, Singapore; Department of Ocean and Resources Engineering, School of Ocean and Earth Science and Technology, University of Hawaii at Manoa, Honolulu, USA; School of Civil and Environmental Engineering, Nanyang Technological University, Singapore

(Received 13 May 2014; final version received 13 May 2015)

Two important issues need to be addressed when using lattice Boltzmann methods to simulate particulate flows: complex moving boundaries and near contact of two solid objects. This study introduces a new boundary treatment method for using lattice Boltzmann methods to study fluid-particle interaction problems in which solid objects are close to a wall or two solid objects are near contact. The new method does not need to use any additional lubrication correction to total hydrodynamic force, instead implementing the no-slip boundary condition by introducing an appropriate fictitious flow field outside the fluid domain. The new method was verified by simulating a spherical particle falling down in fluids with and without rebound. It is concluded that the proposed boundary treatment method can accurately simulate the fluid-particle interaction near a rigid wall and collision in fluid.

Keywords: lattice Boltzmann method; solid-fluid interactions; fluid-particle interaction; boundary treatment

1. Introduction

Particulate flows are ubiquitous in industrial and natural processes, but they are still not well understood. Numerical simulations can further our understanding of particulate flows. Variations of the lattice Boltzmann method (LBM), a class of kinetic-theory-based method (Higuera & Succi, 1989; Higuera, Succi, & Benzi, 1989; McNamara & Zanetti, 1988; McNa- mara & Zanetti, 1988; Succi, Foti, & Higuera, 1989), have been applied to the simulation of particulate flows (Derk- sen, 2011; Z. Feng & Michaelides, 2005, 2009; Jafari, Yamamoto, & Rahmna, 2011; Ladd, 1994a, 1994b; Lin, Lin, Chin, & Tai, 2011; Nguyen & Ladd, 2002; Suzuki & Inamuro, 2013; ten Cate, Nieuwstad, Derksen, & Van den Akker, 2002), and a key component in LBM is the so-called particle distribution function (PDF), which represents a number of fluid particles with certain velocities. When applying an LBM scheme, it is necessary to transform the velocities into the values of PDFs at boundaries. Several methods are available to perform this transformation, including immersed-boundary methods, bounce-back methods, and extrapolation methods.

In most of the immersed-boundary lattice Boltzmann methods (IBLBMs; Dash & Lee, 2014; Y. Feng, Han, & Owen, 2010; Z. Feng & Michaelides, 2005; ten Cate et al., 2002), fluid is assumed to be filled within the solid domain and Lagrangian markers are distributed around the boundaries. Appropriate extra body-forces are applied to the makers to ensure the no-slip boundary condition. The advantage of Eulerian-Lagrangian-based IBLBMs is that it is easy to trace the motion of a boundary. However, Suzuki and Inamuro (2013) indicated that Eulerian-Lagrangian-based IBLBMs introduce a discontinuity in the velocity gradient at boundary, reducing their numerical accuracy. Suzuki and Inamuro further proposed a high-order Eulerian-Lagrangian-based IBLBM to improve the numerical accuracy by distributing the Lagrangian makers inside the solid, but the implementation of their model become complicated.

Bounce-back methods (BBMs; Cornubert, d’Humières, & Levermore, 1991; Frisch et al., 1987) assume that the fluid particles bounce back in their reversed directions when they encounter a solid surface, and are applicable only to stationary solid boundaries. To apply BBMs to moving boundaries, Ladd (1994a) introduced an adjustment term to account for the velocity of a moving boundary. However, this method assumes that the boundary always coincides with the lattice points or lattice midpoints, leading to low numerical resolution for general surface geometries. The numerical resolution at boundary can be improved by the following methods: considering the volume fraction for each boundary fluid node (Verberg & Ladd, 2001), interpolating the PDFs at the boundary (Bouzidi, Firdaouss, & Lallemand, 2001; Filippova & Hänel, 1998; Mei, Luo, & Shyy, 1999) and extrapolating/interpolating the velocity at the lattice midpoints of the boundary lattice links (Yin & Zhang, 2012). Each of these
methods can improve the accuracy of BBMs to second order.

Extrapolation methods introduce ghost nodes outside the fluid domain but adjacent to boundaries. The PDFs in the ghost nodes are decomposed into two parts: equilibrium and non-equilibrium. Equilibrium parts are computed using the velocity and density extrapolated from the fluid domain, and non-equilibrium parts are directly extrapolated from the fluid domain. An extrapolation can be link-based (Guo, Zheng, & Shi, 2002) or normal-direction-based (Tiwari & Vanka, 2012). Extrapolation methods are also of second-order accuracy.

In applying the improved BBMs and extrapolation methods, the total hydrodynamic force exerting on a particle is the sum of the momentum changes occurring on the particle surface. If the distance between two particles or between a particle and a wall is less than a grid size, momentum exchanges at some boundary links crossing two solid boundaries cannot be calculated, and the hydrodynamic force exerting on the particles will be underestimated. Lubrication corrections (Ding & Aidun, 2003; Nguyen & Ladd, 2002) have been proposed, but few studies have checked the accuracy of lubrication corrections by comparing the simulation results with experimental data. The lubrication corrections may significantly influence the behavior of particles near a boundary, as well as the particle-particle interaction and the particle-wall interaction.

This study aims to develop an easy-to-implement and accurate method to simulate fluid-particle interaction near a wall by mirroring flow field near a boundary antisymmetrically into the non-fluid domain. This method can not only treat complex moving boundaries, but it can also accurately determine the hydrodynamic force exerting on a particle that is very close to or in contact with a solid boundary without any additional lubrication correction.

This study introduces a method to accurately model forces exerting on moving particles near or in contact with a rigid boundary. To check its robustness in dealing with complex boundaries, the present method is applied to three problems: cylindrical Couette flows, flows past a sphere with Reynolds numbers ranging from 25 to 200, and sedimentation of a spherical particle in a viscous fluid with and without rebound. The accuracy of the proposed method is validated by comparisons of the sedimentation trajectories and velocities with those reported in published literature (Li, Hunt, & Colonius, 2012; ten Cate et al., 2002).

2. Numerical scheme

2.1. Lattice Boltzmann equation

The core of the lattice Boltzmann method is the so-called particle distribution function, \( f_i(x, t) \), which represents the probability of a particle appearing at location \( x \) at time \( t \). The particle moves at a velocity \( \mathbf{c}_i \), which represents the

\[ \rho = \sum_{i=0}^{14} f_i \quad \text{and} \quad \rho \mathbf{u} = \sum_{i=0}^{14} \mathbf{c}_i f_i. \]

The lattice Boltzmann equation, which describes the evolution of \( f_i(x, t) \), is a discrete form of the following single-relaxation-time Boltzmann equation (He & Luo, 1997a, 1997c):

\[ f_i(x + \mathbf{c}_i \Delta t, t + \Delta t) - f_i(x, t) = -\frac{1}{\tau} [f_i(x, t) - f_i^{eq}(x, t)], \]

where \( \tau \) is the relaxation time and \( f_i^{eq} \) is the equilibrium distribution function to be specified. Eq. (3) recovers to the Navier-Stokes equation to the second order in space.

---

**Figure 1.** D3Q15 lattice.

\( i \)th component of a discrete velocity set. For simplicity, this study adopts a three-dimensional discrete velocity set with 15 components (D3Q15; Qian, Succi, & Orszag, 1995) for three-dimensional simulations. The velocities in D3Q15, as shown in Figure 1, are given by

\[ \mathbf{c}_i = c \times \begin{cases} (0, 0, 0) & i = 0 \\ (\pm 1, 0, 0), (0, \pm 1, 0), & i = 1, 2, \ldots, 6 \\ (0, 0, \pm 1) & i = 7, 8, \ldots, 14 \\ (\pm 1, \pm 1, \pm 1) & \end{cases} \]
and time by applying the Chapman-Enskog expansion with low-Mach-number and with \( \tau \) being given by

\[
\tau = \frac{3v}{c^2 \Delta t} + 0.5, \tag{4}
\]

where \( v \) is the kinematic viscosity of the fluid (He & Luo, 1997b). Since \( v, c, \) and \( \Delta t \) are all positive, Eq. (4) implies that \( \tau \geq 0.5 \). The equilibrium distribution functions depend on the local fluid state and are given by

\[
f_{i}^{eq} = \omega_i \rho \left[ 1 + \frac{3}{c^2} c_i \cdot u + \frac{9}{2c^4} (c_i \cdot u)^2 - \frac{3}{2c^2} u \cdot u \right] \tag{5}
\]

where \( \omega_i \) are weighting factors and are given by (Aidun & Clausen, 2010; Chen & Doolen, 2003; He & Luo, 1997c)

\[
\omega_i = \begin{cases} 
\frac{2}{5} & i = 0 \\
\frac{1}{5} & i = 1, 2, \ldots, 6, \\
\frac{1}{14} & i = 7, 8, \ldots, 14
\end{cases} \tag{6}
\]

Implementation of a lattice Boltzmann method has two steps: a collision step and a streaming step. At the collision step

\[
\tilde{f}_i(x, t) - f_i(x, t) = -\frac{1}{\tau} \left[ f_i(x, t) - f_i^{eq}(x, t) \right] \tag{7}
\]

and at the streaming step

\[
f_i(x + c_i \Delta t, t + \Delta t) = \tilde{f}_i(x, t) \tag{8}
\]

where \( \tilde{f}_i \) represents the post-collision state of the PDF. The collision step simulates fluid particle collisions and their scatters into different directions. The streaming step advances the fluid particles to the next lattice node along their directions of motion. Since both steps take little computational effort, the lattice Boltzmann method is highly efficient and simple to implement.

As aforementioned, the lattice Boltzmann equation (Eq. 3) can recover to the Navier-Stokes equation only for small \( \text{Ma} \equiv u_{\text{max}} / c \) with \( u_{\text{max}} \) being the maximum velocity. Therefore, it is desirable to maintain \( \text{Ma} \) as close to zero as possible; in this regard, He and Luo (1997b) suggest \( \text{Ma} < 0.15 \), and Y. Feng, Han, and Owen (2007) suggest \( \text{Ma} < 0.1 \). Y. Feng et al. also mention that numerical instability will occur when \( \tau \rightarrow 0.5 \). Mathematically, the computational Mach number (Ma) and the relaxation time \( \tau \) represent dimensionless grid size (\( \Delta x \)) and time step (\( \Delta t \)), thus both can affect the numerical accuracy and stability (Chen & Doolen, 2003; Y. Feng et al., 2007). Therefore, \( \Delta x \) and \( \Delta t \) should be chosen such that \( \tau \) should be larger than 0.5 and Ma should be as small as possible. However, the spatial resolution of a flow field is controlled by grid size \( \Delta x \). For a given grid size, both Ma and \( \tau \) decrease with decreasing \( \Delta t \). For a high value of \( \text{Re} \), a small \( \Delta t \) is required to ensure a small Ma. Therefore this method may not be suitable for high Reynolds number flows.

This study applies the LBM to solving moving-particle problems. The previous study (Ladd, 1994b) suggested using an LBM in both the fluid and the solid domains by assuming that the solid domain is also filled with fluid, but that the fluid in the solid domain will reduce the numerical stability (Nguyen & Ladd, 2002). In this study, an LBM is performed only in the fluid domain as suggested by previous studies (Aidun & Lu, 1995; Nguyen & Ladd, 2002). A moving particle leads to a time-varying fluid domain. The density of the new fluid node which is occupied by the particle at the previous time step is given by the initial fluid density for simplicity, and the velocity of the new fluid node is given by the solid velocity at the previous time step. Finally, \( f_i \) of the new fluid node is computed using Eq. (5) as in Aidun and Lu (1995).

2.2. Boundary conditions

We propose a new extrapolation method for treating the no-slip boundary condition, which is similar to the method of images (Graebel, 2001). At first, we consider a simple case in which a boundary is static and located at the midpoint of a lattice link, as shown in Figure 2, where \( B \) represents the boundary; the subscripts \( S \) and \( F \) represent, respectively, the ghost node in the solid domain and the fluid node adjacent to the boundary. We assume that a fictitious flow field exists outside the fluid domain, and the velocity of the fictitious flow field is anti-symmetric to that in the fluid domain. This assumption yields \( \text{u}_S = -u_F, \) \( \partial (-\text{u}_S)/\partial (-\text{x}) = \partial \text{u}_F/\partial \text{x}, \) and a zero velocity at the boundary. Based on this assumption, the amount of fluid particles leaving the fluid domain, \( \tilde{f}_a(x_F, t) \), equals that of fictitious fluid particles entering the fluid domain, \( \tilde{f}_{-a}(x_S, t) \), i.e.,

\[
\tilde{f}_{-a}(x_S, t) = \tilde{f}_a(x_F, t) \tag{9}
\]

Eq. (9) suggests \( \tilde{f}_{-a}(x_F, t + \Delta t) = \tilde{f}_a(x_F, t) \), which is the same as the BBM. Now, we further divide \( \tilde{f}_i \) into a component for the equilibrium state, \( \tilde{f}_i^{eq} \), and a component for the non-equilibrium state, \( \tilde{f}_i^{\text{neq}} \). The component for the equilibrium state can be calculated using Eq. (5). Because

![Figure 2. Illustration of the method of images for boundary treatment in the LBM when the stationary wall is located at the midpoint of a boundary link.](image-url)
\[ u_S = -u_F \text{ and } c_a = -c_a, \]

\[ \tilde{f}_{\alpha}^{eq}(x_S, t) = \tilde{f}_{\alpha}^{eq}(x_F, t), \quad (10) \]

and consequently

\[ \tilde{f}_{\alpha}^{neq}(x_S, t) = \tilde{f}_{\alpha}^{neq}(x_F, t), \quad (11) \]

\[ f_\alpha \text{ is associated with the shear stress of the flow (Mei, Yu, Shyy, & Luo, 2002). Since a shear stress of an incompressible Newtonian fluid depends only on the deformation rate of the fluid element, } f_{\alpha}^{neq}(x_S, t) = f_{\alpha}^{neq}(x_F, t) \text{ holds when } \partial(-u_S)/\partial(-x) = \partial u_F/\partial x. \]

Now, we generalize the above-mentioned boundary treatment method to a moving boundary. For the case in which a boundary moves and does not locate at the midpoint of a link, we assume that the fictitious velocity field relative to the velocity of the boundary is anti-symmetric to the relative velocity field in the fluid domain for each link, as shown in Figure 3. Consequently, one can extrapolate \( u_S \) from the velocity in the fluid domain near the boundary. For numerical stability, as the fraction of the boundary lattice link, \( \Delta \equiv |x_B - x_F|/|x_S - x_F| \), exceeds 0.5, \( u_S \) is calculated using

\[ u_S = \frac{(u_B - u_F)}{\Delta} + u_F. \quad (12) \]

For \( \Delta < 0.5 \),

\[ u_S = \frac{2(u_B - u_F)}{1 + \Delta} + u_{FF}, \quad (13) \]

where \( u_{FF} \) represents the velocity at \( x_F + c_{\alpha/2} \Delta t \). This extrapolation method is also adopted in calculating the velocity at a midpoint of a boundary link (Yin, Le, & Zhang, 2012). We use \( u = u_S \) in Eq. (5) to calculate \( \tilde{f}_{\alpha}^{eq}(x_S, t) \). As a particle is close to a wall or two particles are near contact, \( u_F \) in Eq. (13) may not be available; in this situation we approximate \( u_{FF} \) by the velocity of the solid instead.

For the non-equilibrium component, \( \tilde{f}_{\alpha}^{neq}(x_S, t) \) should be given by \( \tilde{f}_{\alpha}^{neq}(x_B - \Delta c_a[1 - \Delta], t) \) strictly, and \( f_{\alpha}^{neq}(x_B - \Delta c_a[1 - \Delta], t) \) can be interpolated/extrapolated from \( \tilde{f}_{\alpha}^{neq}(x_F, t) \) and \( f_{\alpha}^{neq}(x_F, t) \). Since the non-equilibrium component of the PDF is much smaller than the equilibrium component (He & Luo, 1997b), applying a high-order interpolation/extrapolation method may not be necessary. For simplicity, we approximate \( \tilde{f}_{\alpha}^{neq}(x_S, t) \) by \( f_{\alpha}^{neq}(x_F, t) \). Therefore, \( \tilde{f}_{\alpha}^{neq}(x_S, t) \) is given by

\[ \tilde{f}_{\alpha}(x_S, t) = \tilde{f}_{\alpha}^{eq}(u_S) + \tilde{f}_{\alpha}^{neq}(x_F, t). \quad (14) \]

As mentioned in the introduction, the existing extrapolation methods can be further classified into two groups: link-based methods (Guo et al., 2002) and normal-direction-based methods (Tiwari & Vanka, 2012). The advantage of the normal-direction-based method over the link-based method is its generality, but its extrapolation process is complicated. Before extrapolating \( u_S \) along a direction normal to a boundary, the velocity at a point in the normal direction needs to be interpolated in multiple directions. As a result, the normal-direction-based method is numerically expensive for three-dimensional problems with moving boundaries. The advantage of the link-based method is that it is easy to implement and numerically inexpensive due \( u_S \) only being extrapolated along the lattice link. The present boundary treatment method is in essence a link-based method. However, the proposed method differs from the existing link-based extrapolation method proposed by Guo et al. (2002) with respect to how \( \tilde{f}_{\alpha}^{neq}(x_S, t) \) is determined. The extrapolation method of Guo et al. (2002) extrapolates \( \tilde{f}_{\alpha}^{neq}(x_S, t) \) from \( f_{\alpha}^{neq}(x_F, t) \) and \( \tilde{f}_{\alpha}^{neq}(x_F, t) \). The present extrapolation method is based on the method of images (Graebel, 2001) with \( \tilde{f}_{\alpha}^{neq}(x_S, t) = \tilde{f}_{\alpha}^{eq}(x_F, t) \). The boundary treatment is performed after each streaming step (see Eq. 8). At the streaming step, \( \tilde{f}_{\alpha}(x_F, t) = \tilde{f}_{\alpha}^{eq}(x_F, t) + \tilde{f}_{\alpha}^{neq}(x_F, t) \) is advanced from \( x_F \) to \( x_S \); thus, determining \( \tilde{f}_{\alpha}^{neq}(x_S, t) \) is straightforward.

2.3. Particle dynamics

When a solid object is close to a wall or two solid objects are near contact, the dynamic coupling between fluid and particles become important, but was not addressed by the previous methods (Guo et al., 2002; Tiwari & Vanka, 2012). The motion of a sphere in a viscous fluid is governed by

\[ m \frac{d^2x_p}{dt^2} = F^h + F^c + mg_{eff}. \quad (15) \]

where \( x_p \) is the location of the sphere, \( F^h \) is the hydrodynamic force exerting on the sphere, \( F^c \) is the contact force, \( m \) is the mass of the sphere, and \( g_{eff} \) represents the effective acceleration due to gravity. A linear spring-dashpot model (Silbert et al., 2001) is adopted to calculate the normal contact force for simplicity and the tangential contact force is ignored here. \( g_{eff} \) is defined as \( g_{eff} = (1 - \)
\( \rho / \rho_p \) \( \mathbf{g} \), which incorporates both the buoyant and gravity forces. The angular velocity of a sphere in viscous fluid is governed by

\[
\mathbf{I} \cdot \frac{d\omega_p}{dt} = \mathbf{T}^b, \tag{16}
\]

where \( \omega_p \) is the angular velocity, \( \mathbf{I} \) is the moment of inertia, and \( \mathbf{T}^b \) represents the hydrodynamic torque exerting on the particle. Because the tangential force is not considered in \( \mathbf{F}^b \), the torque resulting from inter-particle contact is zero. In this study, Eq. (15), Eq. (16) and the contact model (Silbert et al., 2001) are solved using an open source code Large-scale Atomix/Molecular Massively Parallel Simulator (LAMMPS; Plimpton, 1995) which adopts the second-order Verlet integration.

The hydrodynamic force, \( \mathbf{F}^b \), on a solid surface can be evaluated by the net momentum flux through the solid surface (Ladd, 1994b), which is given for each boundary lattice link, \( \mathbf{l} \), by

\[
\mathbf{F}(x_B, t) = c_a \left[ f_{\alpha}(x_S, t) - f_{-\alpha}(x_F, t) \right] \frac{\Delta x^2}{\Delta t}, \tag{17}
\]

The total force \( \mathbf{F}^b \) is simply the summation of \( \mathbf{F}(x_B, t) \) for all boundary links

\[
\mathbf{F}^b = \sum \mathbf{F}(x_B, t). \tag{18}
\]

Similarly, the total torque exerting on a particle, \( \mathbf{T}^b \), can be computed by

\[
T^b = \sum (\mathbf{x}_B - \mathbf{x}_p) \times \mathbf{F}(x_B, t). \tag{19}
\]

When the gap between two particles or between a particle and a wall becomes less than \( \Delta x \), as shown in Figure 4, we modify Eq. (17) to calculate \( \mathbf{F}(x_B, t) \). Due to lacking the information for the non-equilibrium component of PDF, the non-equilibrium component of PDF at \( x_B \) and \( x_B \) is neglected. Consequently, Eq. (17) becomes

\[
\mathbf{F}(x_B, t) = c_a \left[ \tilde{f}_{\alpha}^{eq}(\mathbf{u}_{S_1}) - \tilde{f}_{-\alpha}^{eq}(\mathbf{u}_{S_2}) \right] \frac{\Delta x^2}{\Delta t}. \tag{20}
\]

\( \mathbf{u}_{S_1} \) and \( \mathbf{u}_{S_2} \) in Eq. (20) can be extrapolated from \( \mathbf{u}_{B_1} \) and \( \mathbf{u}_{B_2} \). However, when \( |x_B - x_B| \) is too small or \( |\mathbf{u}_{B_1} - \mathbf{u}_{B_2}| \) is too large, the extrapolation may lead to numerical instability. Consequently, \( \mathbf{u}_{S_1} \) and \( \mathbf{u}_{S_2} \) are given by the velocities of the solid at \( x_S \) and \( x_S \) for simplicity and numerical stability.

We stress here that using Eq. (20) to calculate \( \mathbf{F}(x_B, t) \) is one of the novelties of the present boundary treatment method. In previous studies, empirical formulas for lubrication force (Ding & Aidun, 2003; Nguyen & Ladd, 2002) were adopted to correct the total hydrodynamic forces exerting on particles in regions where \( \mathbf{F}(x_B, t) \) cannot be calculated. A detailed comparison between the present boundary treatment method and the lubrication-correction method is presented below.

3. Validation and discussion

To examine its capabilities, the proposed method is applied to four problems in this section: a cylindrical Couette flow, flow past a sphere, sedimentation of a sphere in a viscous fluid, and a sphere rebounding in a viscous fluid.

3.1. Cylindrical Couette flow

To study the capacity of the proposed method to treat a fixed complex boundary, a cylindrical Couette flow, which is confined by two co-axial cylinders, is simulated. An inner cylinder of radius \( R_1 \) is rotated at the angular velocity \( \omega_o \), and the outer cylinder of radius \( R_2 \) remains stationary. The analytical solution of the tangential velocity is given by

\[
\omega_o = \frac{\omega R_1}{R_2 - R_1} \left( \frac{R_2}{r} - \frac{r}{R_2} \right), \tag{21}
\]

where \( r \) is the distance to the cylinder center (Guo & Zhao, 2002). The discrete velocity set of D2Q9 (He & Luo, 1997b), \( \tau = 0.6 \), and \( \Delta x / R_1 = 1 / 8 \) were used to simulate the cylindrical Couette flow. The computed and analytical velocity profiles for \( R_1 / R_2 = 0.1, 0.2, \) and \( 0.5 \) are shown in Figure 5 with \( R_e = 10 \), where the Reynolds number is defined as \( R_e \equiv R_2^2 \omega_o / \nu \). Good agreements between the simulated and analytical results can be observed. To study the convergence rates of the present boundary treatment method, an error analysis was performed by varying \( \Delta x \). As shown in Figure 6, the slopes of the curves for the different values of \( R_e \) are all close to 2, indicating that the boundary treatment is of second-order accuracy. The two previous extrapolation methods (Guo et al., 2002; Tiwari & Vanka, 2012) were also applied to the same problem, and the numerical results showed that they are of second-order accuracy too. For the value of \( E \), we remark that the present boundary treatment method gives numerical errors similar to the method of Guo et al. (2002). As an
3.2. Flow past a sphere

A schematic diagram of a flow past a sphere of diameter \(d\) is shown in Figure 7, where the sphere is fixed at the location \((2.5d, 2.5d, 2.5d)\) in a domain of dimensions \(15d \times 5d \times 5d\). A uniform flow field with velocity \(= U_1\) in the entire simulation domain is specified as the initial condition, and a uniform flow with velocity \(= U_1\) is given at the inlet \(U_1 = 0\), and the gradient-free PDF in the \(U_1\)-direction \((U_1)\) is imposed at the outlet \(U_1 = 15d\). For this problem, a drag coefficient for the sphere can be defined by

\[
Cd \equiv \frac{8F_D}{\pi \rho d^2 U_1^2},
\]

(22)

where \(F_D\) is the drag force (the projection of \(F^h\) in the \(x_1\)-direction). Previous studies (Abraham, 1970; Johnson & Patel, 1999) have shown that the drag coefficients only vary with a Reynolds number defined by \(Re = \frac{U_1 d}{\nu}\), and using the following empirical formula for \(Cd\):

\[
Cd = \frac{64}{Re} + \frac{6}{1 + \sqrt{Re}} + 0.4,
\]

(23)

To simulate the problem using the present boundary treatment method, the following parameters are used: \(\nu = 1 \times 10^{-4} m^2/\text{sec}, d = 0.2 m,\) the grid size \(\Delta x/R_1 = 0.01\), and \(\Delta t = 7.5 \times 10^{-5} \text{sec}\), which give \(\tau = 0.59\). The stationary state is achieved when the relative error in the calculated drag coefficient is less than 1.e.5 between two consecutive time steps.

Figure 8 shows the streamlines computed for \(Re = 50, 100, 150,\) and \(200\), indicating that the flow patterns are similar in the range of \(50 < Re < 200: all have two symmetric vortexes in the leeside of the sphere.

The values of \(Cd\) computed for \(25 < Re < 200\) are shown in Figure 9, together with the numerical results reported in the previous study (Johnson & Patel, 1999), and the empirical relationship Eq. (23). The good agreement suggests that the proposed method can adequately treat stationary complex boundaries.

Figure 10 shows the effects of relaxation time \(\tau\) on the values of \(Cd\). It seems that the drag coefficient is not sensitive to the values of relaxation time: the maximum relative difference in the computed drag coefficients is only about \(1\%\) in the range of \(0.53 < \tau \leq 0.68\).

When choosing the grid size \((\Delta x)\) and the time step \((\Delta t)\) for an LBM simulation, in addition to the requirements on the dimensionless parameters \(Ma\) and \(\tau\) there is also a requirement on the maximum grid size. For this problem, if \(d/\Delta x\) is too small, the flow separation cannot be resolved well, leading to underestimated or overestimated values of \(Cd\). As shown in Figure 11, the grid example, using \(Re = 30\) and \(\Delta x/R_1 = 0.625\), the values of \(E\) obtained by the method of Guo et al. (2002) and the present method are \(7 \times 10^{-3}\) and \(5 \times 10^{-3}\), respectively. Since the numerical errors given in the study of Tiwari and Vanka (2012) is the difference of the two numerical solutions using different grid sizes, no comparison with it is made here.
size has negligible effects on the values of Cd as long as $d/\Delta x \geq 10$.

We remark that the values of Ma for $Re = 50, 100, 150, \text{ and } 200$ using $\tau = 0.59$ are 0.0848, 0.182, 0.266 and 0.358, respectively. Even though large values of Ma ($0.15 \leq Ma \leq 0.358$) can give acceptable results for flow past a sphere, we still adopt $Ma \leq 0.15$ as a constraint in this study. Therefore, in this study, the grid size ($\Delta x$) and the time step ($\Delta t$) are chosen such that $d/\Delta x \geq 10$, $0.53 < \tau \leq 0.68$, and $Ma \leq 0.15$ to ensure numerical accuracy and stability.

3.3. Sedimentation of a spherical particle in a viscous fluid

Sedimentation of a sphere in a tank has been studied experimentally by ten Cate et al. (2002). This is a good example for illustrating how well the proposed method can deal with moving-particle problems. Figure 12 shows a schematic diagram of the experimental setup of ten Cate et al., in which the dimensions of the tank are $0.1 \text{ m} \times 0.1 \text{ m} \times 0.16 \text{ m}$. A sphere of $d = 1.5 \times 10^{-2} \text{ m}$ and $\rho_p = 1120 \text{ kg/m}^3$ was released from a specified location ($0.05 \text{ m}, 0.05 \text{ m}, 0.1275 \text{ m}$) in a fluid, and four fluids were examined in the experiments. In Figure 12, $h$ represents the length of the vertical distance between the sphere and the bottom, and $g = \text{gravitational acceleration}$. 
The details of the fluid properties and the corresponding numerical setup are given in Table 1. A fixed $\Delta x$ is used for all four fluids. The boundaries of the tank lie in the midpoints of the boundary links. The values of $\tau$ were selected to ensure that $\tau$ lies within 0.6 and 0.65. The Stoke number, $St$, in Table 1 is defined as

$$St = \frac{1}{9} \left( \frac{\rho_p}{\rho} \right) \frac{w_{\text{max}} d}{\nu},$$

with $w_{\text{max}}$ being the maximum falling velocity of the sphere, and the rebound of a sphere occurs only for large Stoke numbers ($St > 10$; Ardekani & Rangel, 2008; Joseph, Zenit, Hunt, & Rosenwinkel, 2001; Li et al., 2012). The Stoke numbers for the cases in this section ranged from 0.17 to 3.89, and no rebound occurred (ten Cate et al., 2002).

The computed elevation ($h$) and falling velocity ($w_p$) of the sphere are plotted in Figure 13, where the experimental results of ten Cate et al. (2002) are also shown for comparison. Good agreement is found between the simulated and experimental results for all four fluids. To compare the sensitivity of the numerical results with the relaxation time, Case 4 was also simulated using $\tau = 0.68$ ($\Delta t = 6 \times 10^{-4}$ sec and $\Delta x = 1 \times 10^{-3}$ m), and nearly identical results were obtained.

As pointed out in the introduction, when the sphere is so close to the bottom that the distance between the particle and a wall is less than a grid size, the hydrodynamic forces exerting on particles for some boundary links cannot be calculated due to the resolution issue. One remedy is to use the lubrication force $F_{\text{lub}}$ proposed by Nguyen and Ladd (2002) to correct the force acting on the sphere when $h$ (the distance between the sphere and the bottom) is smaller than a threshold distance $h_{\Delta}$:

$$F_{\text{lub}} = -3\pi \rho v u_\perp \left( \frac{d}{h} - \frac{d}{h_{\Delta}} \right) \tag{24}$$

where $u_\perp$ is the velocity of the sphere perpendicular to the bottom. To examine the ability of the present boundary treatment method to simulate particle movement near a wall, the computed velocity of the sphere near the bottom is shown in Figure 14 for Case 1 in Table 1. Both the experimental and computational results of ten Cate et al. (2002) are given in Figure 14 for comparison. An IBLBM was used by ten Cate et al., and Eq. (24) was adopted to correct the total hydrodynamic force exerting on the sphere. Some previous studies (Z. Feng & Michaelides, 2009; Lin et al.,...
using Eq. (24) increases sharply as \( h \) decreases on the value of \( \Delta \). The outcome correction model proposed by Ding and Aidun (2000) is able to simulate the lubrication force correctly. The lubrication correction has the same shortcoming as that of Nguyen and Ladd. (Silbert et al., 2001) was set to be 3.54 \times 10^{-7} \text{ N/m}, and the coefficient of restitution was set to 0.97 for the collision between a steel sphere and a zero-dur wall (Joseph et al., 2001). Since the timescale for the collision between the particle and the wall is much smaller than the timescale for flows, the time step for solving the equations for the 3.4. Rebound of a spherical particle in a viscous fluid

When \( St > 10 \), the spherical particle will rebound from a wall (Ardekani & Rangel, 2008; Joseph et al., 2001; Li et al., 2012). The rebound of a spherical particle has been studied experimentally (Li et al., 2012). Even though some computational fluid dynamics simulations of rebound problems can be found in the literature (Ardekani & Rangel, 2008; Li et al., 2012), no attempt has been made at the present to simulate rebound problems using LBMs.

We now use the present boundary treatment method to reproduce the experiments of Li et al. (2012). The experiments were performed in a glass rectangular tank filled with a liquid of density 1203 (kg/m^3) and kinematic viscosity 4.173 \times 10^{-5} \text{ m}^2/\text{sec}. A steel sphere of \( d = 9.5 \times 10^{-3} \text{ m} \) and \( \rho_p = 7780 \text{ kg/m}^3 \) was released from an electromagnetic plate (a release mechanism) under the fluid surface. Seven release heights \( (h_r) \) were examined in the experiments; however, only release heights where the computational Mach numbers are smaller than 0.18 and the relaxation times are greater than 0.53 can be simulated efficiently using LBMs with acceptable accuracy and without the numerical stability problem: \( h_r = 5.5 \times 10^{-3} \text{ m} \) and \( h_r = 1.05 \times 10^{-2} \text{ m} \).

The computational domain is similar to the one shown in Figure 12, but with the width of the tank set to 3.8 \times 10^{-2} \text{ m}. The bottom of the electromagnetic plate was located at the midpoints of numerical boundary links. The elastic constant in the linear spring-dashpot contact model (Silbert et al., 2001) was set to be 3.54 \times 10^{-7} \text{ N/m}, and the coefficient of restitution was set to 0.97 for the collision between a steel sphere and a zero-dur wall (Joseph et al., 2001). Since the timescale for the collision between the particle and the wall is much smaller than the timescale for flows, the time step for solving the equations for the
Table 2. Numerical setup for simulating the rebound of a sphere released from various heights.

| Case | \( h_r (m) \) | \( \Delta x (m) \) | \( \Delta t (sec) \) | \( d/\Delta x \) | \( \tau \) | \( M_a \) | \( S_t \) |
|------|----------------|----------------|----------------|-------------|-------|--------|--------|
| 1    | \( 5.5 \times 10^{-3} \) | \( 9.5 \times 10^{-4} \) | \( 5 \times 10^{-4} \) | 10 | 0.569 | 0.12 | 37.91 |
| 2    | \( 5.5 \times 10^{-3} \) | \( 9.5 \times 10^{-4} \) | \( 2.5 \times 10^{-4} \) | 10 | 0.535 | 0.06 | 37.91 |
| 3    | \( 5.5 \times 10^{-3} \) | \( 6.33 \times 10^{-4} \) | \( 2.21 \times 10^{-4} \) | 15 | 0.569 | 0.08 | 39.37 |
| 4    | \( 1.05 \times 10^{-2} \) | \( 9.5 \times 10^{-4} \) | \( 5 \times 10^{-4} \) | 10 | 0.569 | 0.17 | 52.84 |
| 5    | \( 1.05 \times 10^{-2} \) | \( 9.5 \times 10^{-4} \) | \( 2.5 \times 10^{-4} \) | 10 | 0.535 | 0.08 | 54.08 |
| 6    | \( 1.05 \times 10^{-2} \) | \( 6.33 \times 10^{-4} \) | \( 2.21 \times 10^{-4} \) | 15 | 0.569 | 0.11 | 53.76 |

\( h_r = 5.5 \times 10^{-3} \) m and \( h_r = 1.05 \times 10^{-2} \) m Before collision, the numerical results are not sensitive to the relaxation time as long as it is greater than 0.535. However, the relaxation time still can slightly affect the simulated trajectory of the particle after collision. Since the effects of surface roughness (Ardekani, Dabiri, & Rangel, 2008; Li et al., 2012) were not considered in our simulations, the agreement between the simulated and experimental results can be regarded very good, indicating that the present boundary treatment method is capable of modeling collision processes. We remark that reasonable results can be obtained even for \( M_a \) as large as 0.17, which is higher than the upper limit suggested by He and Luo (1997b).

4. Conclusion

There are two important issues that need to be addressed when using lattice Boltzmann methods to simulate fluid-particle interaction near a wall: (1) the treatment of moving boundaries and (2) the determination of the hydrodynamic force exerting on a particle that is close to or in contact with a wall. This study introduces a new extrapolation-based boundary treatment method to address these two important issues: the first issue is addressed by introducing an appropriate fictitious flow field outside the fluid domain, and the second issue is addressed by adopting the velocity of the solid to calculate the equilibrium component of the PDF at the ghost node and evaluating the hydrodynamic force for the boundary link using the net exchange of the equilibrium PDFs at two linked ghost nodes. One feature of the present boundary treatment method is that it is easier to implement than the existing extrapolation methods without sacrificing numerical accuracy. Another important feature of the present boundary treatment method is that it does not need to use any empirical lubrication correction in regions close to rigid boundary.

The present boundary treatment method was applied to four problems: a cylindrical Couette flow, flow past a sphere, sedimentation of a sphere in a viscous fluid, and a sphere rebounding from a wall in a viscous fluid. The present boundary treatment is of second-order accuracy when applied to the study of a cylindrical Couette flow. For flow past a sphere, the present boundary treatment method can compute the drag force accurately. For sedimentation of a sphere in a viscous fluid, the present boundary
treatment method can capture the fluid-particle interaction very well, even when the gap between two solid objects is smaller than one half of the grid size. For a sphere rebounding from a wall in a viscous fluid, the present boundary treatment method can accurately model the entire collision process in fluid. It is concluded that the proposed boundary treatment method can adequately handle problems involving particles in near contact and collision of a particle with a wall. The method can easily be applied to the simulation of particulate flows using LBMs.

Acknowledgments

The second author acknowledges the support from the University of Hawaii at Manoa (UHM) and Nanyang Technological University. This is UHM SOEST contribution number 9442.

Disclosure statement

No potential conflict of interest was reported by the authors.

References

Abraham, F. F. (1970). Functional dependence of drag coefficient of a sphere on Reynolds number. *Physics of Fluids, 13*(8), 2194–2195.

Aidun, C. K., & Clausen, J. R. (2010). Lattice-boltzmann method for complex flows. *Annual Review of Fluid Mechanics, 42*, 439–472.

Aidun, C. K., & Lu, Y. (1995). Lattice Boltzmann simulation of solid particles suspended in fluid. *Journal of Statistical Physics, 81*(1), 49–61.

Ardekani, A. M., Dabiri, S., & Rangel, R. H. (2008). Collision of multi-particle and general shape objects in a viscous fluid. *Journal of Computational Physics, 227*(24), 10094–10107.

Ardekani, A. M., & Rangel, R. H. (2008). Numerical investigation of particle-particle and particle-wall collisions in a viscous fluid. *Journal of Fluid Mechanics, 596*, 437–466.

Bouzidi, M. h., Firdaouss, M., & Lallemand, P. (2001). Momentum transfer of a Boltzmann-lattice fluid with boundaries. *Physics of Fluids, 13*, 3452–3459.

Chen, S., & Doolen, G. D. (2003). Lattice Boltzmann method for fluid flows. *Annual Review of Fluid Mechanics, 35*(1), 329–364.

Cornubert, R., d’Humières, D., & Levermore, D. (1991). A Knudsen layer theory for lattice gases. *Physica D: Nonlinear Phenomena, 47*(1), 241–259.

Dash, S. M., & Lee, T. S. (2014). Natural convection from inclined square cylinder using novel flexible forcing IB-LBM approach. *Engineering Applications of Computational Fluid Mechanics, 8*(1), 91–103.

Derksen, J. (2011). Simulations of granular bed erosion due to lamar shear flow near the critical Shields number. *Physics of Fluids, 23*(11), 113303.

Ding, E.-J., & Aidun, C. K. (2000). The dynamics and scaling law for particles suspended in shear flow with inertia. *Journal of Fluid Mechanics, 423*, 317–344.

Ding, E.-J., & Aidun, C. K. (2003). Extension of the lattice-Boltzmann method for direct simulation of suspended particles near contact. *Journal of Statistical Physics, 112*(3–4), 685–708.

Feng, Y., Han, K., & Owen, D. (2007). Coupled lattice Boltzmann method and discrete element modelling of particle transport in turbulent fluid flows: Computational issues. *International Journal for Numerical Methods in Engineering, 72*(9), 1111–1134.

Feng, Y., Han, K., & Owen, D. (2010). Combined three-dimensional lattice Boltzmann method and discrete element method for modelling fluid–particle interactions with experimental assessment. *International Journal for Numerical Methods in Engineering, 81*(2), 229–245.

Feng, Z. G., & Michaelides, E. E. (2005). Proteus: a direct forcing method in the simulations of particulate flows. *Journal of Computational Physics, 202*(1), 20–51.

Feng, Z. G., & Michaelides, E. E. (2009). Robust treatment of no-slip boundary condition and velocity updating for the lattice-Boltzmann simulation of particulate flows. *Computers & Fluids, 38*(2), 370–381.

Filippova, O., & Hänel, D. (1998). Grid refinement for lattice-BGK models. *Journal of Computational Physics, 147*(1), 219–228.

Frisch, U., d’Humieres, D., Hasslacher, B., Lallemand, P., Pompeau, Y., & Rivet, J. P. (1987). Lattice gas hydrodynamics in two and three dimensions. *Complex Systems, 1*(4), 649–707.

Graebel, W. P. (2001). *Engineering fluid mechanics*. New York: Taylor & Francis.

Guo, Z., & Zhao, T. (2002). Lattice Boltzmann model for incompressible flows through porous media. *Physical Review E, 66*(3), 036304.

Guo, Z. L., Zheng, C. G., & Shi, B. C. (2002). An extrapolation method for boundary conditions in lattice Boltzmann method. *Physics of Fluids, 14*(6), 2007–2010.

He, X., & Luo, L.-S. (1997a). A priori derivation of the lattice Boltzmann equation. *Physical Review E, 55*(6), R6333–R6336.

He, X., & Luo, L.-S. (1997b). Lattice Boltzmann model for the incompressible Navier–Stokes equation. *Journal of Statistical Physics, 88*(3), 927–944.

He, X., & Luo, L.-S. (1997c). Theory of the lattice Boltzmann method: From the Boltzmann equation to the lattice Boltzmann equation. *Physical Review E, 56*(6), 6811–6817.

Higuera, F., Fucci, S., & Benzi, R. (1989). Lattice gas dynamics with enhanced collisions. *Europhysics Letters, 8*(6), 517–521.

Higuera, F., Fucci, S., & Benzi, R. (1989). Lattice gas dynamics with enhanced collisions. *Europhysics Letters, 9*(4), 345–349.

Jafari, S., Yamamoto, R., & Rahnama, M. (2011). LatticeBoltzmann method combined with smoothed-profile method for particulate suspensions. *Physical Review E, 83*(2), 026702.

Johnson, T. A., & Patel, V. C. (1999). Flow past a sphere up to a Reynolds number of 300. *Journal of Fluid Mechanics, 378*, 19–70.

Joseph, G. G., Zenit, R., Hunt, M. L., & Rosenwinkel, A. M. (2001). Particle-wall collisions in a viscous fluid. *Journal of Fluid Mechanics, 433*, 329–346.

Ladd, A. J. C. (1994a). Numerical simulations of particulate suspensions via a discretized Boltzmann equation. Part 1. Theoretical foundation. *Journal of Fluid Mechanics, 271*, 285–309.

Ladd, A. J. C. (1994b). Numerical simulations of particulate suspensions via a discretized Boltzmann equation. Part 2. Numerical results. *Journal of Fluid Mechanics, 271*, 311–339.

Lin, S. Y., Lin, C. T., Chin, Y. H., & Tai, Y. H. (2011). A direct-forcing pressure-based lattice Boltzmann method for solving...
fluid-particle interaction problems. *International Journal for Numerical Methods in Fluids*, 66(5), 648–670.

Li, X. B., Hunt, M. L., & Colonius, T. (2012). A contact model for normal immersed collisions between a particle and a wall. *Journal of Fluid Mechanics*, 691, 123–145.

McNamara, G. R., & Zanetti, G. (1988). Use of the Boltzmann equation to simulate lattice-gas automata. *Physical Review Letters*, 61(20), 2332–2335.

Mei, R., Luo, L.-S., & Shyy, W. (1999). An accurate curved boundary treatment in the lattice Boltzmann method. *Journal of Computational Physics*, 155(2), 307–330.

Mei, R., Yu, D., Shyy, W., & Luo, L.-S. (2002). Force evaluation in the lattice Boltzmann method involving curved geometry. *Physical Review E*, 65(4), 041203.

Nguyen, N. Q., & Ladd, A. (2002). Lubrication corrections for lattice-Boltzmann simulations of particle suspensions. *Physical Review E*, 66(4), 046708.

Plimpton, S. (1995). Fast parallel algorithms for short-range molecular dynamics. *Journal of Computational Physics*, 117(1), 1–19.

Qian, Y. H., Succi, S., & Orszag, S. (1995). Recent advances in lattice Boltzmann computing. *Annual Reviews of Computational Physics*, 3, 195–242.

Silbert, L. E., Ertas, D., Grest, G. S., Halsey, T. C., Levine, D., & Plimpton, S. J. (2001). Granular flow down an inclined plane: Bagnold scaling and rheology. *Phys. Rev. E*, 64(5), 051302.

Succi, S., Foti, E., & Higuera, F. (1989). Three-dimensional flows in complex geometries with the lattice Boltzmann method. *Europhysics Letters*, 10(5), 433–438.

Suzuki, K., & Inamuro, T. (2013). A higher-order immersed boundary–lattice Boltzmann method using a smooth velocity field near boundaries. *Computers & Fluids*, 105–115.

ten Cate, A., Nieuwstad, C. H., Derksen, J. J., & Van den Akker, H. E. A. (2002). Particle imaging velocimetry experiments and lattice-Boltzmann simulations on a single sphere settling under gravity. *Physics of Fluids*, 14(11), 4012–4025.

Tiwari, A., & Vanka, S. P. (2012). A ghost fluid Lattice Boltzmann method for complex geometries. *International Journal for Numerical Methods in Fluids*, 69(2), 481–498.

Verberg, R., & Ladd, A. J. C. (2001). Accuracy and stability of a lattice-Boltzmann model with subgrid scale boundary conditions. *Physical Review E*, 65(1), 016701.

Yin, X. W., Le, G. G., & Zhang, J. F. (2012). Mass and momentum transfer across solid-fluid boundaries in the lattice-Boltzmann method. *Physical Review E*, 86(2), 026701.

Yin, X. W., & Zhang, J. F. (2012). An improved bounce-back scheme for complex boundary conditions in lattice Boltzmann method. *Journal of Computational Physics*, 231(11), 4295–4303.