






## Lattice Boltzmann modeling of fluid-particle interaction based on a two-phase mixture representation

Chrysovalantis Tsigginos , Jianping Meng (孟剑平) ,\* Xiao-Jun Gu , and David R. Emerson   
*Scientific Computing Department, STFC Daresbury Laboratory, Warrington WA4 4AD, United Kingdom*

 (Received 12 September 2019; revised manuscript received 29 October 2019; published 31 December 2019)

In this work, we derive a lattice Boltzmann model for fluid-particle interaction by considering the system as a two-phase mixture. A partially saturated type scheme is achieved rigorously without any viscosity-dependent weight parameter. The scheme is of second-order accuracy in both space and time including the body-force term. Moreover, we devise a scheme suitable for the scenario where two or more particles intersect a single computational cell, typically occurring for particles in contact or close to contact. Good performance is found when the present scheme is validated against three classic problems, namely the flow past a stationary cylinder, a cylindrical particle settling in a channel under gravity, and the flow around two impacting cylinders.

DOI: [10.1103/PhysRevE.100.063311](https://doi.org/10.1103/PhysRevE.100.063311)

### I. INTRODUCTION

Applications involving fluid-particle interaction appear in broad areas, including biological [1,2], chemical [3,4], and civil engineering [5,6]. Due to its importance, a significant amount of analytical, experimental, and numerical procedures have been developed to study fluid-particle interaction.

Numerical modeling of fluid-particle interaction problems usually employ an Eulerian framework for the fluid and a Lagrangian framework for the motion and deformation of the particles. As a consequence, this has led to two approaches being developed. The first can be classified as a direct method, treating the fluid-particle interface as a boundary and generally applying the no-slip boundary condition [7–9]. The direct method is considered to be more accurate but may be computationally expensive due to the requirement for continuous remeshing. In contrast, indirect approaches use other ways (e.g., force [10–12] and treating a solid particle as an undeformable fluid [13]) to constrain the velocities of the fluid and particle at their interface. Such a method does not require remeshing but may be of lower accuracy since the no-slip boundary condition may not be properly enforced at the fluid-particle interface.

For fluid flows, the lattice Boltzmann method (LBM) [14] has been developing rapidly and offers an attractive approach, which is based on tracking and updating distribution functions at the mesoscopic level. The macroscopic variables of interest (e.g., density, velocity, and pressure) are the first few moments of the distribution function. Due to its explicit particle-like nature, which is based on a uniform Cartesian grid, the LBM

is easy to implement and parallelize. In particular, it is an excellent candidate for modeling flows around multiple moving particles. Thus, various direct and indirect schemes have been developed for LBM to solve fluid-particle interaction problems [15–20].

A popular indirect method is the immersed boundary method, originally proposed by Peskin [21]. In this approach, a restoration force is introduced to indirectly impose the boundary condition at the fluid-particle interface. In its earlier version, this force was assumed to be linearly dependant on a user defined parameter that controls the efficiency [16,21]. More recent work [22–24] has suggested a form of force that is dependant on the momentum exchange crossing the fluid-particle interface.

Within the LBM framework, continuous remeshing is not generally required when applying a direct method. Instead, the cut-cell meshing technique is well adapted. The simplest form may be the commonly used staircase approximation of a curved boundary, although it can lead to reduced accuracy in simulations [25]. In 1994, Ladd and Verberg [15] employed this technique to model particulate suspensions by applying the bounce-back boundary condition at the fluid-solid interface. To improve the accuracy, various interpolated bounce-back schemes [18–20,26] were proposed. However, these schemes can have difficulty with mass conservation [27].

Noble and Torczynski [17] proposed the partially saturated method (PSM), which is a second-order accurate scheme and also conserves mass [28]. In this scheme, the lattice nodes are classified as solid, partially saturated, or fluid nodes based on the solid fraction of a computational cell. An additional collision operator is introduced to enforce the solid phase velocity at solid and partially saturated cells. The accuracy of the scheme varies with a weight parameter,  $B$ . If this weight parameter is chosen as the solid fraction, then first-order accuracy is observed. In contrast, second-order accuracy is achieved when this weight parameter becomes a viscosity-dependent formulation [17]. The accurate representation of the fluid-particle interface, as well as the localized mapping

\*Corresponding author: [jianping.meng@stfc.ac.uk](mailto:jianping.meng@stfc.ac.uk)

*Published by the American Physical Society under the terms of the Creative Commons Attribution 4.0 International license. Further distribution of this work must maintain attribution to the author(s) and the published article's title, journal citation, and DOI.*

of particles into the fluid domain, makes the PSM a popular choice to model the fluid-particle interaction generated in fluidized beds [29–32]. Due to its success, work has often focused on improving the method. Zhou *et al.* [33] replaced the solid fraction with the shortest lattice distance to reduce the computational cost. Chen *et al.* [34] introduced a different form of the weight parameter,  $B$ , and replaced the solid collision operator with the ghost field method [20] in order to enforce the no-slip boundary condition at the fluid-particle interface. Wang *et al.* [35] introduced nonlinear relaxation-time-dependent terms into the  $B$  parameter to reduce the viscosity dependence. These efforts have generally focused on the weight parameter while the body-force term has received less attention. Forcing terms were introduced into the PSM by Strack *et al.* [36], but as reported by Tsigginos *et al.* [37], the scheme demonstrates a lower order of accuracy (i.e., 1.5) when body forces are applied to the whole fluid-particle interaction system.

In this work, we discuss the modeling of fluid-particle interactions within the lattice Boltzmann framework. First, we introduce a continuous Boltzmann-BGK (Bhatnagar-Gross-Krook) equation for the fluid-particle interaction system which is consistent with mixture theory. For incompressible flows, the lattice Boltzmann model can be derived from this equation by using a standard procedure [38–40]. In this way, we can obtain a second-order accurate scheme in both space and time including the body-force term. Moreover, we discuss the treatment of the case where multiple particles intersect a single computational cell and present the details of the derivation in Sec. II and its numerical validation in Sec. III.

## II. MESOSCOPIC MIXTURE REPRESENTATION OF FLUID-PARTICLE INTERACTION

In this work, we focus on particle resolved modeling of the fluid-particle interaction problem. In the fluid part, computational cells are typically set to be smaller than the characteristic size of the immersed particles so that the flow details are fully captured. The motion of a particle can be resolved by using a suitable Lagrangian solver, e.g., the discrete element method (DEM). The difficulty in such modeling is related to the necessity of mapping the particle into the fluid domain and tracking the fluid-particle interface.

### A. Lattice Boltzmann model for fluid flows around immersed particles

The fluid flow around immersed particles is modeled as a two-phase mixture. A *virtual* fluid is introduced to fill the space occupied by the particles. This virtual fluid has the same density as the fluid phase but has an infinite viscosity for mimicking the solid behavior. The impact of the solid phase on the fluid behavior (e.g., the momentum exchange) will be first transferred to the virtual fluid and then felt by the fluid phase (see Sec. IIC). The fluid phase together with the virtual fluid are considered as interpenetrating continua [41]. To describe such a mixture, we employ the concept of porosity,  $\varepsilon$ , and solid fraction,  $\varphi = 1 - \varepsilon$  [42]. Thus, particles can be mapped into the fluid domain, and three different types of computational cells are identified, i.e., pure fluid cells,

$\varepsilon = 1$ , pure virtual fluid (solid) cells,  $\varepsilon = 0$ , or partially saturated cells,  $0 < \varepsilon < 1$ , where each type may occupy a significant portion of the computational cell. At the mesoscopic level, a Boltzmann-BGK equation can be written for the mixture, i.e.,

$$\frac{\partial f}{\partial t} + \boldsymbol{\xi} \cdot \frac{\partial f}{\partial \mathbf{x}} + \varepsilon \frac{\mathbf{F}_b}{\rho} \cdot \frac{\partial f}{\partial \boldsymbol{\xi}} = \varepsilon \Omega = \frac{\varepsilon}{\tau} (f^{\text{eq}} - f), \quad (1)$$

where a distribution function,  $f(\mathbf{x}, \boldsymbol{\xi}, t)$ , is employed to describe the motion of the mixture in a space of dimension,  $d$ , which represents the number of fluid particles in a volume,  $d\mathbf{x}$ , centered at position,  $\mathbf{x}$ , with molecular velocities within,  $d\boldsymbol{\xi}$ , around velocity,  $\boldsymbol{\xi}$ , at time,  $t$ . The density is denoted by  $\rho$  and  $\tau$  is the relaxation time. A BGK collision term,  $\Omega = (f^{\text{eq}} - f)/\tau$  [43], is introduced to model the complicated molecular interaction of the fluid phase, which describes the relaxation process to the Maxwellian distribution,

$$f^{\text{eq}} = \frac{\rho}{(2\pi RT)^{d/2}} \exp \left[ -\frac{(\boldsymbol{\xi} - \mathbf{u})^2}{2RT} \right]. \quad (2)$$

For the virtual fluid, the collision term is ignored due to its infinite viscosity (i.e., the collision term becomes zero). The body force  $\mathbf{F}_b$  (typically the gravitational force,  $\mathbf{g}$ ) is also considered for the fluid phase, where its effect on the virtual fluid will be introduced during the fluid-particle interaction stage through the updated solid velocities, cf. Eq. (17). For the isothermal and incompressible flows discussed in this work, the temperature,  $T$ , is constant and equal to the reference temperature,  $T_0$ . The pressure,  $p$ , is equal to  $\rho RT_0 = \rho c_s^2$ , where the sound speed is defined as  $c_s = \sqrt{RT_0}$  and  $R$  is the specific gas constant.

The macroscopic variables including the mixture density,  $\rho$ , the mixture momentum,  $\rho \mathbf{u}$ , and the mixture pressure,  $p$ , can be calculated by integrating the distribution function over the whole molecular velocity space,  $\boldsymbol{\xi}$ , i.e.,

$$\rho = \int f d\boldsymbol{\xi}; \quad \rho \mathbf{u} = \int \boldsymbol{\xi} f d\boldsymbol{\xi}; \quad p = \frac{d}{2} \int (\boldsymbol{\xi} - \mathbf{u})^2 f d\boldsymbol{\xi}. \quad (3)$$

It is worth noting that the mixture velocity,  $\mathbf{u}$ , is the average of the fluid phase velocity,  $\mathbf{u}^f$ , and virtual fluid phase velocity,  $\mathbf{u}^s$ , i.e.,

$$\mathbf{u} = \varepsilon \mathbf{u}^f + (1 - \varepsilon) \mathbf{u}^s. \quad (4)$$

Through the Chapman-Enskog expansion (see Appendix A), the following Navier-Stokes equations

$$\frac{\partial \rho}{\partial t} + \frac{\partial(\rho u_i)}{\partial x_i} = 0, \quad (5a)$$

$$\frac{\partial(\rho u_i)}{\partial t} + \frac{\partial(\rho u_j u_i)}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \frac{\partial}{\partial x_j} \left[ \mu_e \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \right] + \varepsilon F_{bi}, \quad (5b)$$

are recovered for an isothermal and incompressible mixture system. The mixture viscosity,  $\mu_e$ , is

$$\mu_e = \frac{\mu_f}{\varepsilon} = \frac{p\tau}{\varepsilon}, \quad (6)$$

where the term  $\mu_f = p\tau$  corresponds to the viscosity of the fluid phase. It can be seen that the mixture viscosity becomes infinite for the virtual fluid (i.e.,  $\varepsilon = 0$ ) as assumed.

The derived macroscopic equations are consistent with those previously proposed for mixtures, see Eqs. (99) and (100) in Ref. [44], where the diffusion velocity is zero since the densities of the two phases are identical [44]. Moreover, the mixture viscosity corresponds to the Reuss viscosity of a mixture [45]. The mixture equations are generally applicable to colloidal suspensions [13], particle-laden flows with low loading, bubbly flows, sedimentation, and cyclone separators.

It is possible to numerically solve Eq. (1) by a direct discretization in physical space,  $\mathbf{x}$ , and molecular velocity space,  $\xi$ . However, such simulations are very expensive and unnecessary for a range of continuum flows, in particular, isothermal and incompressible flows. For such flows, the continuous BGK equation, Eq. (1), can be reduced to a much simpler discrete equation by expanding the equilibrium distribution function on a Hermite polynomial basis and discretizing the molecular velocity space,  $\xi$ , accordingly [38–40]. Since the discretization procedure is standard, the details are not repeated here and only the main equations are summarized. By choosing an appropriate Gauss-Hermite quadrature with abscissae,  $\xi_\alpha$ , and weights,  $w_\alpha$ , the discrete BGK equation is written as

$$\frac{\partial f_\alpha}{\partial t} + \xi_\alpha \frac{\partial f_\alpha}{\partial \mathbf{x}} = \varepsilon \Omega_\alpha + \varepsilon \mathcal{F}_\alpha, \quad (7)$$

where the collision term and the force term are discretized as  $\Omega_\alpha = (f_\alpha^{\text{eq}} - f_\alpha)/\tau$  and  $\mathcal{F}_\alpha$ , respectively. For isothermal and incompressible flows, a second-order expansion is considered to be sufficient, i.e.,

$$f_\alpha^{\text{eq}} = w_\alpha \rho \left[ 1 + \frac{\xi_\alpha \cdot \mathbf{u}}{c_s^2} + \frac{1}{2} \left( \frac{\xi_\alpha \cdot \mathbf{u}}{c_s^2} \right)^2 - \frac{\mathbf{u} \cdot \mathbf{u}}{2c_s^2} \right]. \quad (8)$$

Accordingly a second-order force term,

$$\mathcal{F}_\alpha = w_\alpha \left[ \frac{\xi_\alpha \cdot \mathbf{F}_b}{c_s^2} + \frac{(\xi_\alpha \cdot \mathbf{u})(\xi_\alpha \cdot \mathbf{F}_b)}{c_s^4} - \frac{\mathbf{u} \cdot \mathbf{F}_b}{c_s^2} \right], \quad (9)$$

is adopted herein [38]. For two-dimensional flows, it is common to employ the so-called D2Q9 lattice, where the abscissae are chosen as

$$\begin{aligned} \xi_0 &= \sqrt{3}c_s(0, 0), \\ \xi_1 &= \sqrt{3}c_s(1, 0), \quad \xi_2 = \sqrt{3}c_s(0, 1), \\ \xi_3 &= \sqrt{3}c_s(-1, 0), \quad \xi_4 = \sqrt{3}c_s(0, -1), \\ \xi_5 &= \sqrt{3}c_s(1, 1), \quad \xi_6 = \sqrt{3}c_s(-1, 1), \\ \xi_7 &= \sqrt{3}c_s(-1, -1), \quad \xi_8 = \sqrt{3}c_s(1, -1), \end{aligned} \quad (10)$$

and the weight factors are

$$w_0 = \frac{4}{9}, \quad w_{1..4} = \frac{1}{9}, \quad w_{5..8} = \frac{1}{36}. \quad (11)$$

A second-order strategy [46] is used to integrate Eq. (7) where a trapezoidal integration is applied to the right-hand side and the left-hand side can be integrated precisely, and we obtain

$$\begin{aligned} &f_\alpha(\mathbf{x} + \xi_\alpha \Delta t, t + \Delta t) - f_\alpha(\mathbf{x}, t) \\ &= \varepsilon \frac{\Delta t}{2} [\Omega_\alpha(\mathbf{x} + \xi_\alpha \Delta t, t + \Delta t) + \Omega_\alpha(\mathbf{x}, t)] \\ &+ \varepsilon \frac{\Delta t}{2} [\mathcal{F}_\alpha(\mathbf{x} + \xi_\alpha \Delta t, t + \Delta t) + \mathcal{F}_\alpha(\mathbf{x}, t)]. \end{aligned} \quad (12)$$

To eliminate the implicitness, a new variable,  $\bar{f}_\alpha$  [46], can be introduced, i.e.,

$$\bar{f}_\alpha = f_\alpha - \frac{\varepsilon \Delta t}{2} \Omega_\alpha - \frac{\varepsilon \Delta t}{2} \mathcal{F}_\alpha, \quad (13)$$

and Eq. (12) is rewritten as

$$\begin{aligned} &\bar{f}_\alpha(\mathbf{x} + \xi_\alpha \Delta t, t + \Delta t) \\ &= \bar{f}_\alpha - \frac{\varepsilon \Delta t}{\tau + 0.5\varepsilon \Delta t} (\bar{f}_\alpha - f_\alpha^{\text{eq}}) + \frac{\varepsilon \tau \Delta t}{\tau + 0.5\varepsilon \Delta t} \mathcal{F}_\alpha. \end{aligned} \quad (14)$$

By using  $\bar{f}_\alpha$ , the density and momentum can be calculated as

$$\rho = \sum_\alpha \bar{f}_\alpha, \quad (15a)$$

$$\rho \mathbf{u} = \sum_\alpha \xi_\alpha \bar{f}_\alpha + \frac{\varepsilon}{2} \mathbf{F}_b \Delta t. \quad (15b)$$

A second-order accurate lattice Boltzmann scheme is therefore obtained for flows around immersed particles based on a mixture representation of the fluid-particle interaction system. For later discussions, it is also convenient to rewrite Eq. (13)

$$f_\alpha = \frac{2\hat{\tau} \bar{f}_\alpha + \varepsilon f_\alpha^{\text{eq}} + \varepsilon \hat{\tau} \Delta t \mathcal{F}_\alpha}{2\hat{\tau} + \varepsilon}, \quad (16)$$

where the normalized relaxation time,  $\hat{\tau}$ , is defined as  $\hat{\tau} = \tau/\Delta t$ .

## B. Motion of particles

In this study, we assume that particles do not deform so that their motion can be described through translation and rotation. For a particle,  $k$ , of mass,  $m_k$ , and moment of inertia,  $I_k$ , Newton's second law states

$$m_k \dot{\mathbf{v}}_k = m_k \mathbf{g} + \mathbf{F}, \quad (17a)$$

$$I_k \dot{\boldsymbol{\omega}}_k = \mathbf{T}, \quad (17b)$$

where  $\mathbf{v}_k$  and  $\boldsymbol{\omega}_k$  are translation and rotational velocity vectors of its center of mass. The immersed particle is subjected to a force,  $\mathbf{F}$ , and a torque,  $\mathbf{T}$ , exerted by the surrounding fluid and the gravitational acceleration,  $\mathbf{g}$ . It is also convenient to introduce the velocity of a point,  $P$ , in the particle,  $\mathbf{V}_P = \mathbf{v}_k + \boldsymbol{\omega}_k \times \mathbf{r}$ , where  $\mathbf{r}$  is the radius vector of the point relative to the mass center.

## C. Momentum exchange between fluid and particle

The core of modeling the fluid-particle interaction system as a two-phase mixture is the transfer of the solid phase response into the virtual fluid phase. To achieve this, the virtual fluid velocities of solid cells,  $\varepsilon = 0$ , and partially saturated cells,  $0 < \varepsilon < 1$ , i.e., the second term of the right-hand side of Eq. (4), will be set to the solid body velocity. These operations are equivalent to enforcing a no-slip boundary condition at the fluid-particle interface, and are implemented in conjunction with other boundary conditions (e.g., wall boundaries) in this work (see Ref. [37] for detailed discussions).

During complex fluid-particle interactions, it is possible for two or more particles with different velocities to intersect

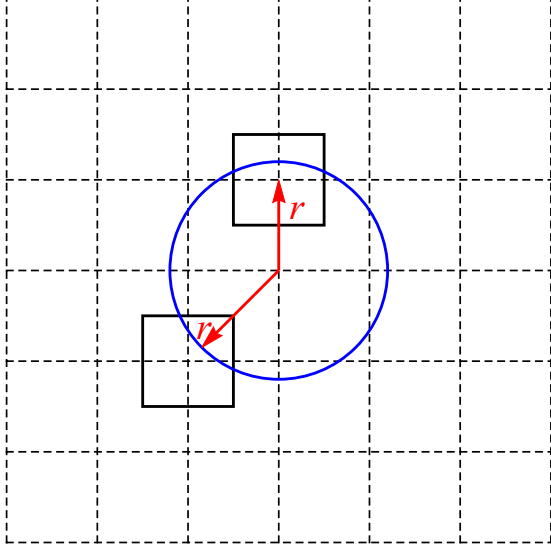


FIG. 1. Illustration of the vector  $r$  for calculating the velocity,  $V_p$ .

a single computational cell, typically for flows around impacting particles. In this case, special treatment is necessary compared to the simpler case where there is only one particle intersecting a computational cell. Therefore, we first discuss the latter case (named as single-particle scheme) in Sec. II C 1 and further develop the scheme (named as multiple-particle scheme) for the former one in Sec. II C 2.

### 1. Single-particle scheme

To enforce the velocity of the virtual fluid phase,  $\mathbf{u}^s$ , to be the velocity of the solid,  $\mathbf{V}_p$ , the collision-like operator developed in Ref. [17], i.e.,

$$f_\alpha^* = f_\alpha + (1 - \varepsilon)[f_{\bar{\alpha}} - f_{\bar{\alpha}}^{\text{eq}}(\rho, \mathbf{u})] - (1 - \varepsilon)[f_\alpha - f_\alpha^{\text{eq}}(\rho, \mathbf{V}_p)], \quad (18)$$

will be adapted to the evolution equation (14) based on  $\bar{f}_\alpha$ , where  $f_{\bar{\alpha}}$ ,  $f_{\bar{\alpha}}^{\text{eq}}$  refers to  $f$  and  $f^{\text{eq}}$  for the direction  $\bar{\alpha}$  such that  $\bar{\xi}_\alpha = -\xi_\alpha$ . This operator reflects the nonequilibrium part of the distribution function, and conserves mass. For partially saturated nodes, the calculation of velocity,  $\mathbf{V}_p$ , is implemented by using the vector  $r$  as illustrated by Fig. 1 according to whether the grid point is inside the particle or not.

To aid the analysis, we make the reasonable assumption that momentum exchange between the fluid and solid is faster than the flow variation so that the flow conditions are unchanged during the operation prescribed in Eq. (18). Thus, the fluid collision and forcing terms remain unchanged during the momentum exchange stage (i.e., the fluid collision operator,  $\Omega_\alpha^*$ , and forcing term,  $\mathcal{F}_\alpha^*$ , after momentum exchange are equal to the fluid collision operator,  $\Omega_\alpha$ , and forcing term,  $\mathcal{F}_\alpha$ , prior to momentum exchange). Following the definition in Eq. (13), we are able to replace the left-hand side term and the first term at the right-hand side with  $\bar{f}_\alpha$ , i.e.,

$$\begin{aligned} \bar{f}_\alpha^* + \frac{1}{2}\varepsilon\Omega_\alpha^*\Delta t + \frac{1}{2}\varepsilon\mathcal{F}_\alpha^*\Delta t \\ = \bar{f}_\alpha + \frac{1}{2}\varepsilon\Omega_\alpha\Delta t + \frac{1}{2}\varepsilon\mathcal{F}_\alpha\Delta t \\ + (1 - \varepsilon)\underbrace{[f_{\bar{\alpha}} - f_{\bar{\alpha}}^{\text{eq}}(\rho, \mathbf{u})]}_{\textcircled{1}} - (1 - \varepsilon)\underbrace{[f_\alpha - f_\alpha^{\text{eq}}(\rho, \mathbf{V}_p)]}_{\textcircled{2}}. \end{aligned} \quad (19)$$

Substituting Eq. (16) into  $\textcircled{1}$  and  $\textcircled{2}$ , we obtain

$$\textcircled{1} = \frac{\hat{\tau}}{\hat{\tau} + 0.5\varepsilon}(\bar{f}_{\bar{\alpha}} - f_{\bar{\alpha}}^{\text{eq}}) + \frac{1}{2}\frac{\varepsilon\hat{\tau}}{\hat{\tau} + 0.5\varepsilon}\mathcal{F}_{\bar{\alpha}}\Delta t \quad (20)$$

and

$$\begin{aligned} \textcircled{2} = \frac{\hat{\tau}}{\hat{\tau} + 0.5\varepsilon}\left[\bar{f}_\alpha + \frac{\varepsilon}{2\hat{\tau}}f_\alpha^{\text{eq}} - \frac{\hat{\tau} + 0.5\varepsilon}{\hat{\tau}}f_\alpha^{\text{eq}}(\rho, \mathbf{V}_p)\right] \\ + \frac{1}{2}\frac{\varepsilon\hat{\tau}}{\hat{\tau} + 0.5\varepsilon}\mathcal{F}_\alpha\Delta t. \end{aligned} \quad (21)$$

After a few algebraic manipulations, Eq. (19) can be written as

$$\bar{f}_\alpha^* = \bar{f}_\alpha + \frac{\hat{\tau}(1 - \varepsilon)}{\hat{\tau} + 0.5\varepsilon}\left\{\underbrace{\bar{f}_{\bar{\alpha}} - f_{\bar{\alpha}}^{\text{eq}}}_{\Omega_{\text{I}}^s} - \underbrace{\left[\bar{f}_\alpha + \frac{\varepsilon}{2\hat{\tau}}f_\alpha^{\text{eq}} - \frac{\hat{\tau} + 0.5\varepsilon}{\hat{\tau}}f_\alpha^{\text{eq}}(\rho, \mathbf{V}_p)\right]}_{\Omega_{\text{II}}^s} + \underbrace{\frac{\varepsilon\Delta t}{2}(\mathcal{F}_{\bar{\alpha}} - \mathcal{F}_\alpha)}_{\Omega_{\text{III}}^s}\right\}. \quad (22)$$

Equation (22) is an exact transformation of Eq. (19) in terms of  $\bar{f}_\alpha$ , which can be grouped into three subterms  $\Omega_{\text{I}}^s$ ,  $\Omega_{\text{II}}^s$ , and  $\Omega_{\text{III}}^s$ . The term  $\Omega_{\text{I}}^s$  represents the reflection of the nonequilibrium part of  $\bar{f}_\alpha$ , while in  $\Omega_{\text{II}}^s$  the momentum is transferred from the solid body to the virtual fluid. The term  $\Omega_{\text{III}}^s$  is due to the dependency of  $\bar{f}_\alpha$  on the body force,  $\mathcal{F}_\alpha$ . However, for partially saturated nodes, Eq. (22) can lead to an inconsistency between the virtual fluid phase velocity,  $\mathbf{u}^s$ , and solid point velocities,  $\mathbf{V}_p$ , i.e., the virtual fluid phase velocity should be driven to  $\mathbf{V}_p$  by the momentum exchange rather than a mix of  $\mathbf{u}$  and  $\mathbf{V}_p$ . This can make the simulation relatively inaccurate,

as identified in Sec. III. To improve the accuracy, the term  $\Omega_{\text{II}}^s$  is revised as

$$\Omega_{\text{II}}^s = \bar{f}_\alpha - f_\alpha^{\text{eq}}(\rho, \mathbf{V}_p) \quad (23)$$

and Eq. (22) becomes

$$\begin{aligned} \bar{f}_\alpha^* = \bar{f}_\alpha + \frac{(1 - \varepsilon)\hat{\tau}}{\hat{\tau} + 0.5\varepsilon}[\bar{f}_{\bar{\alpha}} - f_{\bar{\alpha}}^{\text{eq}} - \bar{f}_\alpha + f_\alpha^{\text{eq}}(\rho, \mathbf{V}_p)] \\ + \frac{1}{2}\frac{\varepsilon(1 - \varepsilon)\hat{\tau}}{\hat{\tau} + 0.5\varepsilon}\Delta t(\mathcal{F}_{\bar{\alpha}} - \mathcal{F}_\alpha). \end{aligned} \quad (24)$$



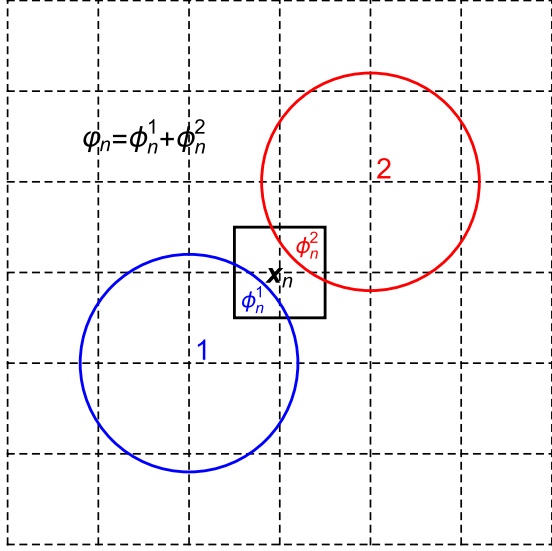


FIG. 2. Illustration of two particles “1” and “2” intersecting a computational cell around grid point  $x_n$ .

Based on the form of  $\Omega_{\text{II}}^s$ , we denote the scheme utilizing Eq. (22) as the momentum exchange scheme I (MES-I), and the second one employing Eq. (24) as the momentum exchange scheme II (MES-II). The performance of the two schemes will be further discussed in Sec. III.

The force  $\mathbf{F}$  and torque  $\mathbf{T}$  exerted from the fluid to the solid body can be calculated as

$$\mathbf{F} = -\frac{\Delta x^d}{\Delta t} \sum_n \frac{(1 - \varepsilon_n) \hat{\tau}}{\hat{\tau} + 0.5 \varepsilon_n} \sum_{\alpha} (\Omega_{\text{I}}^s - \Omega_{\text{II}}^s + \Omega_{\text{III}}^s) \xi_{\alpha}, \quad (25a)$$

$$\mathbf{T} = -\frac{\Delta x^d}{\Delta t} \sum_n \frac{(1 - \varepsilon_n) \hat{\tau}}{\hat{\tau} + 0.5 \varepsilon_n} \mathbf{r}_n \times \sum_{\alpha} (\Omega_{\text{I}}^s - \Omega_{\text{II}}^s + \Omega_{\text{III}}^s) \xi_{\alpha}, \quad (25b)$$

where  $n$  corresponds to grid points that are in contact with the solid body.

It is worthwhile to note that the derived scheme can be used in applications where multiple particles move with different velocities, provided that only a single particle intersects a computational cell at any given time. Moreover, the scheme can also be used in the case where more than one particle intersects the computational cell if all particles have the same velocity. In the following section, we will elaborate on the multiple-particle scheme.

## 2. Multiple-particle scheme

For flows around particles in contact or close to contact, it is likely that multiple particles with different velocities can occupy a single computational cell (see Fig. 2). In this case, the calculation of the fluid-particle interaction based on the single-particle scheme, Eq. (25), is problematic since the intersecting particles are treated as a single entity with an ambiguous solid velocity. Thus, the solid fraction,  $\phi^k$ , and the associated term,  $\Omega_{\text{II}}^s$ , have to be calculated for each particle,  $k$ , that intersects with the grid cell. Meanwhile, the terms  $\Omega_{\text{I}}^s$  and  $\Omega_{\text{III}}^s$  are invariant to the number of particles that intersect

with the cell, since these terms are defined in terms of the total solid fraction,  $\varphi = \sum_k \phi^k$ , and macroscopic fluid properties. Based on the above, Eq. (22) is rewritten as

$$\begin{aligned} \bar{f}_{\alpha}^* &= \bar{f}_{\alpha} + \frac{\varphi \hat{\tau}}{\hat{\tau} + 0.5(1 - \varphi)} \Omega_{\text{I}}^s + \frac{\hat{\tau}}{\hat{\tau} + 0.5(1 - \varphi)} \sum_k \phi_k \Omega_{\text{II}}^{s,k} \\ &+ \frac{\varphi \hat{\tau}}{\hat{\tau} + 0.5(1 - \varphi)} \Omega_{\text{III}}^s, \end{aligned} \quad (26)$$

where  $\Omega_{\text{II}}^{s,k}$  is the  $\Omega_{\text{II}}^s$  component calculated based on the velocity of particle,  $k$ .

The force,  $\mathbf{F}_k$ , and torque,  $\mathbf{T}_k$ , exerted from the fluid to the  $k$ th particle can be calculated as

$$\mathbf{F}_k = -\frac{\Delta x^d}{\Delta t} \sum_n \frac{\phi_n^k \hat{\tau}}{\hat{\tau} + 0.5(1 - \varphi_n)} \sum_{\alpha} (\Omega_{\text{I}}^s - \Omega_{\text{II}}^{s,k} + \Omega_{\text{III}}^s) \xi_{\alpha}, \quad (27a)$$

$$\begin{aligned} \mathbf{T}_k &= -\frac{\Delta x^d}{\Delta t} \sum_n \frac{\phi_n^k \hat{\tau}}{\hat{\tau} + 0.5(1 - \varphi_n)} \mathbf{r}_n^k \\ &\times \sum_{\alpha} (\Omega_{\text{I}}^s - \Omega_{\text{II}}^{s,k} + \Omega_{\text{III}}^s) \xi_{\alpha}, \end{aligned} \quad (27b)$$

where  $\phi_n^k$  is the ratio of the volume occupied by the particle,  $k$ , at the computational cell,  $n$ , i.e., the cell around grid point,  $x_n$ , to the cell volume.

## D. Remarks

For convenience, we list the steps in a full time cycle of the present LBM model in Fig. 3. Although there is a difference on whether treating the momentum exchange between fluid and particle at the collision term (see Ref. [37] for detail), the scheme bears a close resemblance to the PSM proposed by Noble and Torczynski [17]

$$\begin{aligned} f_{\alpha}(\mathbf{x} + \xi_{\alpha} \Delta t, t + \Delta t) \\ = f_{\alpha}(\mathbf{x}, t) + (1 - B) \Omega_{\alpha} \Delta t + B \Omega_{\alpha}^s + (1 - B) \mathcal{F}_{\alpha} \Delta t, \end{aligned} \quad (28)$$

where  $\Omega_{\alpha}^s$  is equal to

$$\Omega_{\alpha}^s = [f_{\bar{\alpha}} - f_{\bar{\alpha}}^{\text{eq}}(\rho, \mathbf{u})] - [f_{\alpha} - f_{\alpha}^{\text{eq}}(\rho, \mathbf{V}_P)]. \quad (29)$$

In both models, the concept of the solid fraction and/or porosity are adopted so that the complex fluid-particle interaction problem can be simulated in a straightforward manner. In the present work, these concepts are linked to the macroscopic mixture theory, Eq. (5), by using a Chapman-Enskog expansion over Eq. (1), see Appendix A. In this way, we gain further understanding of the model, and are able to rigorously derive the lattice Boltzmann scheme.

In the PSM [17], two forms of the parameter  $B$  were suggested, i.e., the solid fraction  $\varphi$  and

$$B = \frac{\hat{\tau} \varphi}{\hat{\tau} + (1 - \varphi)}, \quad (30)$$

where the normalized relaxation time,  $\hat{\tau}_N$ , of Noble and Torczynski [17] in the absence of body forces is defined as  $\hat{\tau}_N = \hat{\tau} + 0.5$ .

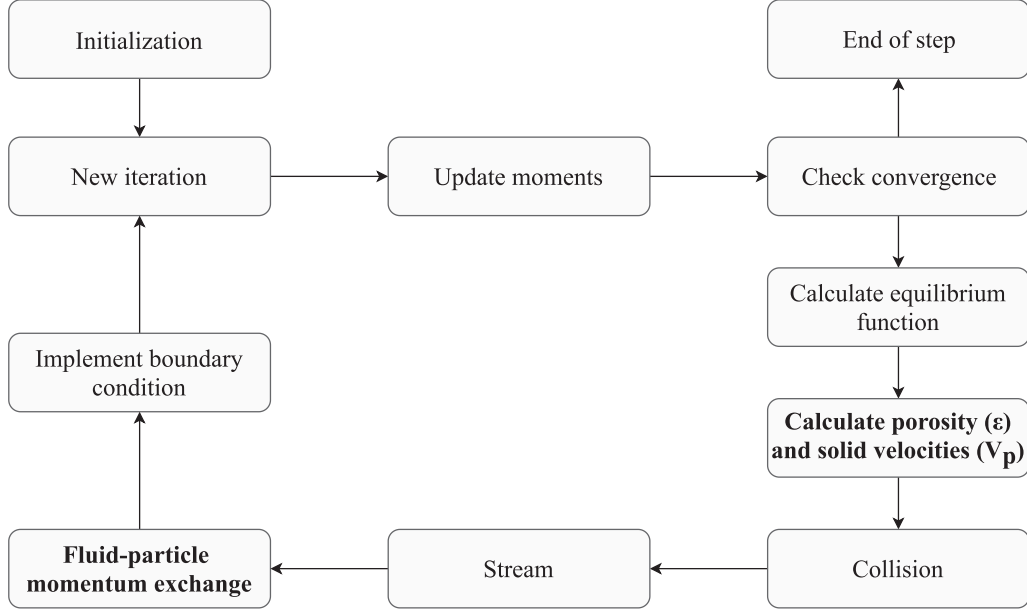


FIG. 3. A time cycle of the present LBM modeling of fluid-particle interaction.

Based on the process of deriving Eq. (14) from Eq. (7), it is not difficult to identify that the PSM using,  $B = \varphi$ , is a first-order discretization in space and time of Eq. (7). This explains the observed first-order accuracy in numerical simulations [17]. To achieve second-order accuracy, the formulation of Eq. (30) was suggested in a heuristic way, which can lead to an inaccurate treatment of the body force. Instead, a consistent formulation is derived for the body force in the present work, where improved performance is observed in Sec. III B. Importantly, there is no viscosity-dependent free parameter like Eq. (30); only porosity and/or solid fraction are introduced and then all formulations can be obtained from the numerical discretization procedure.

### III. NUMERICAL VALIDATIONS

To numerically validate the proposed model, three benchmark cases have been considered, i.e., flow around a cylinder, particle settling in a channel, and flow around two impacting cylinders. For simplicity, we will carry out two-dimensional simulations although the extension to three dimensions is straightforward and only requires the calculation of the solid fraction in three dimensions. To calculate the solid fraction, several strategies have been proposed including numerical integration [37,47], cell decomposition [48], and polygonal approximation [31]. In the following simulations, a fourth-order Gauss integration scheme is employed which is sufficient for the present second-order lattice Boltzmann scheme.

#### A. Flow past a cylinder

We employ a computational domain of length,  $L$ , and width,  $W$ , to simulate both confined and unconfined flows past a stationary infinite cylinder of diameter,  $D$ , which is placed at the center ( $L/2, W/2$ ) of the domain, see Fig. 4. If the flow is confined in a channel, then the top and bottom boundaries are set to be solid walls. Moreover, a wall correction factor,  $\lambda$ ,

dependent on the aspect ratio,  $\kappa = D/W$ , is often introduced to describe the confined effects of a channel wall. It is defined as the ratio of the drag force,  $F$ , to the drag force developed in unconfined flows, i.e.,

$$\lambda = \frac{F(\kappa)}{u_m \mu_f}, \quad (31)$$

where the drag force in unconfined flows is considered to be proportional to the product of the maximum velocity,  $u_m$ , at the inlet and the fluid viscosity. Faxén [49] obtained a form of correction factor,  $\lambda_F$ , i.e.,

$$\lambda_F = \frac{4\pi}{h_1(\kappa) + g_1(\kappa)}, \quad (32)$$

where  $h_1(\kappa)$  and  $g_1(\kappa)$  are

$$h_1(\kappa) = A_0 - (1 + 0.5\kappa^2 + A_4\kappa^4 + A_6\kappa^6 + A_8\kappa^8) \ln(\kappa),$$

$$g_1(\kappa) = B_2\kappa^2 + B_4\kappa^4 + B_6\kappa^6 + B_8\kappa^8,$$

$$A_0 = -0.9156892732, \quad B_2 = 1.26653975,$$

$$A_4 = 0.05464866, \quad B_4 = -0.9180433,$$

$$A_6 = -0.26462967, \quad B_6 = 1.8771010,$$

$$A_8 = 0.792986, \quad B_8 = -4.66549.$$

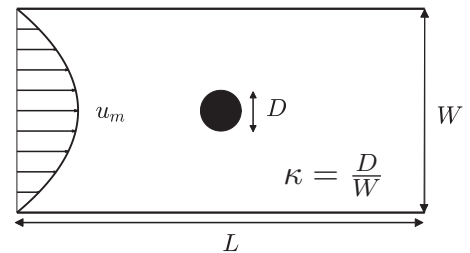


FIG. 4. Computational domain for flows around a cylinder.

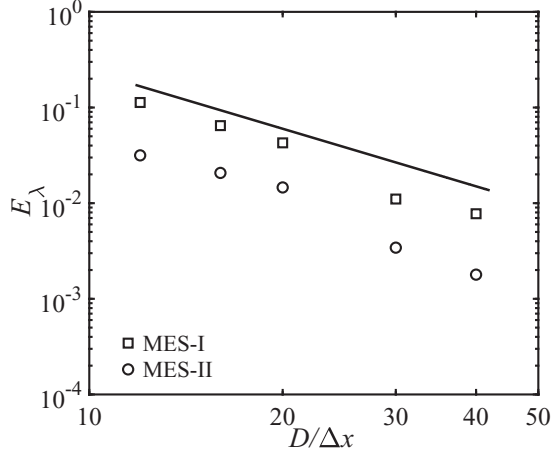


FIG. 5. Accuracy of the present LBM simulations with MES-I and MES-II for a channel flow past a stationary cylinder. For convenience, a guide line of slope  $-2$  is added.

As a benchmark case, we considered a channel of width  $2.5D$ , i.e.,  $\kappa = 0.4$ . A preliminary study shows that the length of  $40D$  is sufficient for the development of a fully resolved fluid flow, since an increase of 50% of the channel length led to a 0.001% increase of the correction factor.

A parabolic velocity profile with a maximum speed of  $u_m = 10^{-6}$  in lattice units is prescribed at both the inlet and outlet to simulate the Poiseuille flow, see Fig. 4. The fluid density, viscosity, and particle diameter are set to be 1, 0.005, and 1, respectively, in the lattice unit, which leads to a Reynolds number,  $\rho u_m D / \mu$ , of  $2 \times 10^{-4}$ . The no-slip boundary condition is implemented at the top and bottom wall using the equilibrium diffuse reflection (EDR) boundary scheme [50]. Simulations are run with a few different grids, i.e.,  $D/\Delta x = [12, 16, 20, 30, 40]$ , and the scheme accuracy is measured by comparing the calculated wall correction factor with  $\lambda_F$  via

$$E_\lambda = \left| \frac{\lambda - \lambda_F}{\lambda_F} \right|. \quad (33)$$

As has been shown in Fig. 5, simulations with both MES-I and MES-II demonstrate accuracy higher than second order (2.37 for MES-I and 2.55 for MES-II), which is consistent with their theoretical accuracy. However, MES-I induces significantly higher errors in comparison to MES-II for the same grid.

We believe that the performance drop of MES-I compared to MES-II is related to the term,  $\Omega_{II}$ , i.e., if the corresponding terms can properly enforce the solid point velocity,  $\mathbf{V}_P$ , at partially saturated nodes. In MES-II the formulation of  $\Omega_{II}$  directly enforces  $\mathbf{V}_P$  at both partially saturated nodes  $\varphi < 1$  and solid (virtual fluid) nodes. On the contrary, the terms that MES-I utilizes only satisfies the same condition at solid nodes,  $\varphi = 1$ . For partially saturated nodes, the enforced velocity is also dependant on the fluid phase velocity.

A further assessment of the two schemes reveals that their optimal range of normalized relaxation time,  $\hat{\tau}$ , is different. Specifically, simulations using MES-II shows better performance for normalized relaxation times in the range

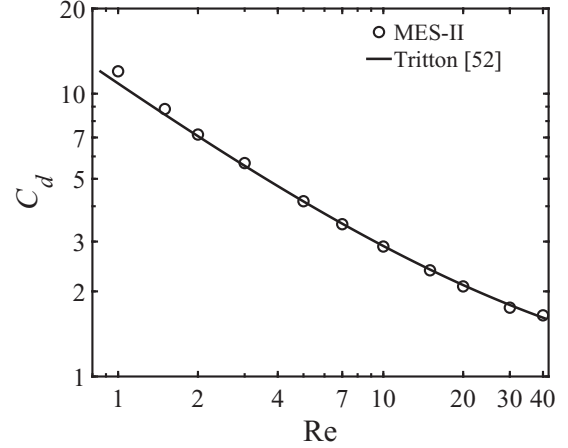


FIG. 6. Drag coefficient  $C_d$  predicted by MES-II for unconfined flows past a cylinder at various Reynolds numbers. The experimental results in Ref. [52] are included for comparison.

of  $[0.1, 0.55]$  while the first scheme performs better in the range of  $[0.6, 0.9]$ . Due to its better performance, we will employ MES-II by default in the following validations unless otherwise specified.

We also consider unconfined flows past an infinite cylinder. For this case, we set the domain length and width to be  $50D$  which is considered as adequate since an increase of 50% of the domain size leads to roughly 2% increase of the drag coefficient at  $Re = 40$ . A constant velocity  $u_1$  is imposed at the inlet using the EDR scheme, while the pressure is specified as unity in lattice units at the outlet using the implementation discussed in Ref. [51]. The top and bottom boundaries are implemented as a fixed velocity (equal to the inlet velocity) boundary using the EDR scheme. At  $Re = 2$ , we also tested the zero-gradient boundary condition for the top and bottom boundary and found negligible difference in drag force calculation. The viscosity,  $\mu_f$ , and particle diameter,  $D$ , are set to 0.001 and 1 in all simulations and the inlet velocity,  $u_1$ , is adjusted to simulate flows with Reynolds number,  $\rho u_1 D / \mu_f$ , ranging from 0 to 40. Based on the results in Fig. 5, we choose the lattice space  $\Delta x$  to be  $D/30$ . For steady flows,  $Re \leq 40$ , the simulations were carried out until converged (i.e., the velocity residual is less than  $10^{-7}$ ) and the drag coefficient is computed as

$$C_d = \frac{2F}{\rho u_1^2 D}. \quad (34)$$

The calculated drag coefficient is compared with the experimental results of Tritton [52] in Fig. 6. For Reynolds numbers above 1, the calculated drag coefficients show good agreement with the experimental data. For  $Re = 1$ , a better agreement was found with the analytical solution  $8\pi/Re(2.002 - \ln Re)$ , Eq. (9.19) in Ref. [53], where the relative error is 4.4% compared to 10.3% with the experimental values.

## B. Settling of a particle in a channel under gravity

To verify the accuracy for fluid flows driven by a body force, we considered the case of an infinite cylindrical particle settling in a channel under gravity,  $g$ . Initially, the particle

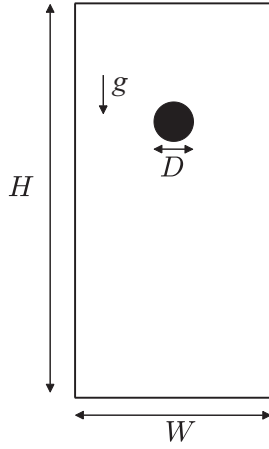


FIG. 7. Computational domain for case of a particle settling inside a channel under gravity.

is at rest and a drag force is developed as it starts moving under gravity. At some point, the drag force will become large enough so that gravity is balanced. From this point, the particle moves with a constant velocity,  $V_c$ , which is called the terminal velocity. For small Reynolds number,  $\text{Re} < 0.001$ , the terminal velocity for a particle of density  $\rho_s$  and diameter  $D$  can be estimated as [54]

$$V_c = \frac{\pi g D^2 (\rho_s - \rho)}{4 \mu_f \lambda_F(\kappa)}, \quad (35)$$

where  $\mu_f$  is the fluid viscosity. Faxén's correction factor,  $\lambda_F$ , for a cylindrical particle moving between two parallel plane walls [49] is equal to

$$\lambda_F = \frac{4\pi}{A_0 - \ln(\kappa) + A_2 \kappa^2 + A_4 \kappa^4 + A_6 \kappa^6 + A_8 \kappa^8}, \quad (36)$$

where

$$\begin{aligned} A_0 &= -0.9156892732, & A_2 &= 1.7243844, \\ A_4 &= -1.730194, & A_6 &= 2.405644, \\ A_8 &= -4.59131, \end{aligned}$$

and  $\kappa$  is the aspect ratio of the channel defined as  $\kappa = D/W$  (Fig. 7). The formulation, Eq. (35), is believed to be valid for  $\kappa \leq 0.2$ . To compare with Eq. (35), we consider a channel of width  $5D$  which leads to an aspect ratio  $\kappa$  of 0.2. To minimize the influence of the channel height, it was set to  $80D$ . No-slip boundary conditions are prescribed at the boundaries of the channel. The particle density is set to 2.6 in lattice units and the fluid viscosity is 0.005. The diameter of the infinite cylinder is set to 1. The magnitude of the gravitational acceleration is specified such that the Reynolds number,  $\rho V_c D / \mu_f$ , is equal to  $5 \times 10^{-4}$ . The accuracy of the proposed scheme is evaluated by running simulations with five different lattices, i.e.,  $D/\Delta x = [12, 16, 20, 30, 40]$ , and measuring the error,

$$E_v = \left| \frac{V_c - V_c}{V_c} \right|, \quad (37)$$

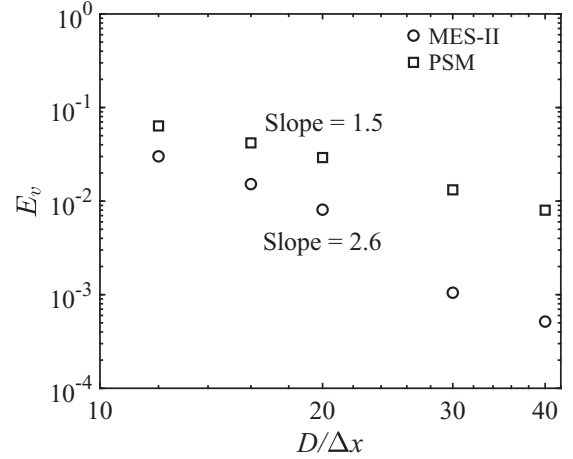


FIG. 8. Accuracy of MES-II and PSM for a cylindrical particle settling in a channel under gravity.

between the calculated terminal velocity,  $V_c$ , and the one given by Eq. (35).

The results are shown in Fig. 8. For convenience, we also include the predictions of the original PSM where the force term is treated in the way described in Eq. (28). While both schemes converge to Eq. (35), the present LBM model with MES-II shows better performance by demonstrating an accuracy order of 2.6 and much smaller error. In contrast, the PSM shows an accuracy order of 1.5, which is lower than the one for cases without body force [17,36]. As discussed in Sec. IID, this deficiency can be attributed to the inappropriate treatment of the body-force term.

### C. Flow around two impacting cylinders

Complex flows are generated when the fluid interacts with particles that are in contact or near contact. The simplest case of such flows is the flow around two impacting cylinders, where the fluid is initially stationary [55–58]. This case is used to assess the performance of Eq. (14) for more complex flows as well as the performance of the multiple-particle scheme discussed in Sec. IIC2. Although a multiple-particle type scheme has been discussed in the context of LBM and PSM by Noble and Torczynski [17], as far as we know, this is the first time we can fully assess the performance of multiple-particle schemes.

Two infinite cylinders with an identical diameter,  $D$ , were positioned in a square box of width  $50D$  with an initial gap,  $G$ , of  $2D$  (Fig. 9). The fluid viscosity is set to be 0.0015 in lattice

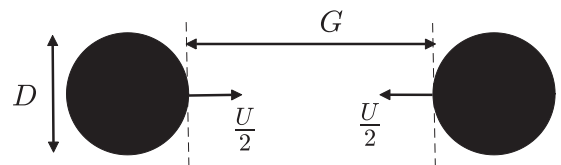


FIG. 9. The setup of two infinite cylinders of diameter  $D$  moving toward each other with an initial velocity of  $U/2$ . The initial gap of the two cylinders is  $2D$ .



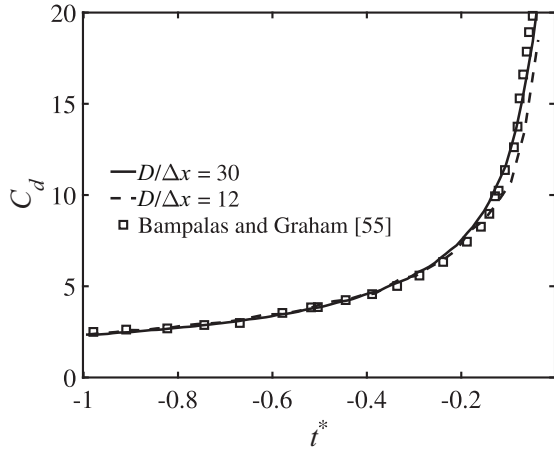


FIG. 10. Evolution of the drag coefficient for flow past two impacting cylinders against the dimensionless time  $t^*$  at  $Re = 50$  using two sets of  $D/\Delta x$ . The results of Bampalas and Graham [55] are included for comparison.

units. The cylinders are moving toward each other with the same speed,  $U/2$ , that leads to a Reynolds number,  $\rho U D/2\mu_f$ , of 50. The square box is enclosed by four stationary walls and the no-slip boundary condition is imposed using the EDR scheme. Two grids with a ratio  $D/\Delta x$  equal to 12 and 30 are utilized to discretize the computational domain. The multiple-particle scheme is applied to model the momentum exchange between the fluid and particle.

The evolution of the drag coefficient is presented in Fig. 10 where the time is normalized as  $t^* = U(t - t_{\text{impact}})/D$  by using the relative speed  $U$  of the cylinders, and the time of cylinder collision,  $t_{\text{impact}} = 2D/U$ . For comparison, the finite-element results of Bampalas and Graham [55] are also depicted in Fig. 10. For the finer grid, the multiple-particle scheme is equivalent to the single-particle scheme when the gap,  $G$ , between the cylinders is larger than  $0.033D$  ( $t^* = -0.033$ ), while for the coarser grid, this critical gap becomes  $0.083D$  ( $t^* = -0.083$ ). For  $t^*$  less than  $-0.3$ , the LBM predictions agrees well with the finite-element predictions of Bampalas and Graham [55] independent of grid size  $D/\Delta x$ . However, the finer grid appears to be necessary for capturing the force exerted from the fluid to an individual cylinder when the gap between the two cylinders is smaller than  $0.3D$  ( $t^* = -0.3$ ) as the performance using the coarser grid is less satisfactory. For the coarser grid, Fig. 10, the transition from the single-particle scheme to the multiple-particle scheme is smooth at around  $t^* = -0.083$ , and the transition does not affect the accuracy of the proposed scheme since the divergence from the finite-element results occurs earlier (i.e., at  $t^* \approx 0.25$ ) which we believe to be associated with the coarser grid. In general, the multiple-particle scheme condition allows the calculation of the drag force of an individual particle, which cannot be achieved with the single-particle scheme since it cannot distinguish two or more different particles in the same computational cell and Eq. (25) becomes invalid. Moreover, better performance is expected if particles move with different velocities.

#### IV. CONCLUSIONS

We derived a lattice Boltzmann model for fluid-particle interaction by considering the fluid-particle system as a two-phase mixture. A virtual fluid, which has the same density as the actual fluid and an infinite viscosity, was introduced to transfer the momentum from the solid phase (i.e., particle) to the two-phase mixture. Based on this concept, a Boltzmann-BGK equation that is consistent with existing macroscopic mixture equations (via a Chapman-Enskog expansion) was devised. By employing the time and spatial discretization of He *et al.* [46], we derived a scheme that is second-order accurate in both time and space including the force term. In contrast to the popular partially saturated method proposed in Ref. [17], the proposed scheme is mathematically rigorous and does not require the introduction of a weight parameter,  $B$ , that nonphysically depends on fluid viscosity to achieve second-order accuracy. The present model introduces a second-order treatment of the forcing term, a condition that is not fulfilled by the PSM. Moreover, we devise a multiple-particle scheme, which is suitable for cases where two or more particles intersect a single computational cell, typically occurring for particles in contact or close to contact. The proposed scheme was validated against three typical problems, namely the flow past a stationary cylinder, a cylindrical particle settling under gravity, and the flow around two impacting cylinders. Good performance was observed in all cases. In particular, the case of a particle settling under gravity confirms the appropriateness of the body-force treatment and the ability of the model to achieve second-order accuracy in the presence of body forces, a condition that was not achieved by the original PSM.

#### ACKNOWLEDGMENTS

We thank the Engineering and Physical Science Research Council for their support under projects EP/N033841/1 (C.T., J.M., and D.E.), EP/P022243/1 (J.M. and D.E.), EP/R029598/1 (J.M. and D.E.), and EP/N016602/1 (D.E.). We also thank Dr. Qin Yang from College of Mathematics Physics and Information Engineering, Jiaying University, China for her efforts in performing a few simulations in Sec. III A.

#### APPENDIX A

A Chapman-Enskog expansion [28] is performed to find the macroscopic equations recovered by the Eq. (1) at the continuum limit. The solution of the distribution function is assumed to be an asymptotic series,

$$f = f^0 + \epsilon f^{(1)} + \epsilon^2 f^{(2)} + \dots, \quad (\text{A1})$$

where the symbol  $\epsilon$  plays the role of the Knudsen number. It is further assumed that these terms satisfy the following conditions:

$$\int f^{(k)} d\xi = 0, \quad \int \xi f^{(k)} d\xi = 0, \quad \int \xi \cdot \xi f^{(k)} d^3 \xi = 0 \quad k > 0. \quad (\text{A2})$$

Furthermore, the time and spatial derivatives as well as the forcing term are expanded as follows:

$$\frac{\partial}{\partial t} = \epsilon \frac{\partial^{(1)}}{\partial t} + \epsilon^2 \frac{\partial^{(2)}}{\partial t} + \dots, \quad (\text{A3a})$$

$$\frac{\partial}{\partial x_i} = \epsilon \frac{\partial^{(1)}}{\partial x_i}, \quad (\text{A3b})$$

$$F_i = \epsilon F_i^{(1)}. \quad (\text{A3c})$$

Substituting Eq. (A1), the derivatives and the force expansion Eq. (A3) into Eq. (1), we obtain

$$O(\epsilon^{(0)}) : f^0 = f^{\text{eq}}, \quad (\text{A4a})$$

$$O(\epsilon^{(1)}) : \frac{\partial^{(1)}}{\partial t} f^{\text{eq}} + \xi_i \frac{\partial^{(1)}}{\partial x_i} f^{\text{eq}} + \epsilon \frac{F_{b,i}^{(1)}}{\rho} \frac{\partial}{\partial \xi_i} f^{\text{eq}} = -\frac{\epsilon}{\tau} f^{(1)}, \quad (\text{A4b})$$

$$O(\epsilon^{(2)}) : \frac{\partial^{(2)}}{\partial t} f^{\text{eq}} + \left[ \frac{\partial^{(1)}}{\partial t} + \xi_i \frac{\partial^{(1)}}{\partial x_i} \right] f^{(1)} + \epsilon \frac{F_{b,i}^{(1)}}{\rho} \frac{\partial}{\partial \xi_i} f^{(1)} = -\frac{\epsilon}{\tau} f^{(2)}. \quad (\text{A4c})$$

Taking the zeroth to second moment of Eq. (A4), the  $O(\epsilon)$ -order macroscopic equations are as follows:

$$\frac{\partial^{(1)}}{\partial t} \rho + \frac{\partial^{(1)}}{\partial x_i} (\rho u_i) = 0, \quad (\text{A5a})$$

$$\frac{\partial^{(1)}}{\partial t} (\rho u_i) + \frac{\partial^{(1)}}{\partial x_j} \Pi_{ij}^{\text{eq}} - \epsilon F_{b,i}^{(1)} = 0, \quad (\text{A5b})$$

$$\frac{\partial^{(1)}}{\partial t} \Pi_{ij}^{\text{eq}} + \frac{\partial^{(1)}}{\partial x_k} \Pi_{ijk}^{\text{eq}} = -\frac{\epsilon}{\tau} \Pi_{ij}^{(1)}. \quad (\text{A5c})$$

The terms  $\Pi_{ij}^{\text{eq}}$ ,  $\Pi_{ijk}^{\text{eq}}$ , and  $\Pi_{ij}^{(1)}$  are defined as

$$\Pi_{ij}^{\text{eq}} = \int \xi_i \xi_j f^{\text{eq}} d\xi = \rho u_i u_j + \rho c_s^2 \delta_{ij}, \quad (\text{A6a})$$

$$\begin{aligned} \Pi_{ijk}^{\text{eq}} &= \int \xi_i \xi_j \xi_k f^{\text{eq}} d\xi = \rho c_s^2 (u_i \delta_{jk} + u_j \delta_{ik} + u_k \delta_{ij}) \\ &\quad + \rho u_i u_j u_k, \end{aligned} \quad (\text{A6b})$$

$$\Pi_{ij}^{(1)} = \int \xi_i \xi_j f^{(1)} d\xi. \quad (\text{A6c})$$

In a similar manner, the zeroth and first moments at  $O(\epsilon^2)$  are

$$\frac{\partial^{(2)}}{\partial t} \rho = 0, \quad (\text{A7a})$$

$$\frac{\partial^{(2)}}{\partial t} (\rho u_i) + \frac{\partial^{(1)}}{\partial x_j} \Pi_{ij}^{(1)} = 0. \quad (\text{A7b})$$

Collecting the  $O(\epsilon)$  and  $O(\epsilon^2)$  terms, we can obtain the following equations:

$$\frac{\partial \rho}{\partial t} + \frac{\partial (\rho u_i)}{\partial x_i} = 0, \quad (\text{A8a})$$

$$\frac{\partial (\rho u_i)}{\partial t} + \frac{\partial \Pi_{ij}^{\text{eq}}}{\partial x_j} - \epsilon F_i = -\frac{\partial \Pi_{ij}^{(1)}}{\partial x_j}, \quad (\text{A8b})$$

where Eq. (A8) is the continuity equation. To find an explicit form for Eq. (A8), we need some more algebraic manipulations based on Eq. (A5c) (for more details see Ref. [59]) and arrive at

$$\Pi_{ij}^{(1)} = -\frac{\rho c_s^2 \tau}{\epsilon} \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \quad (\text{A9})$$

and

$$\begin{aligned} &\frac{\partial (\rho u_i)}{\partial t} + \frac{\partial (\rho u_j u_i)}{\partial x_j} \\ &= -\frac{\partial p}{\partial x_i} + \frac{\partial}{\partial x_j} \left[ \mu_e \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \right] + \epsilon F_i, \end{aligned} \quad (\text{A10})$$

where the pressure is  $p = \rho c_s^2$  and the mixture viscosity,  $\mu_e$ , becomes

$$\mu_e = \frac{p\tau}{\epsilon}. \quad (\text{A11})$$

- 
- [1] E. C. Eckstein, A. W. Tilles, and F. J. Millero III, *Microvasc. Res.* **36**, 31 (1988).
- [2] K. Müller, D. A. Fedosov, and G. Gompper, *Sci. Rep.* **4**, 4871 (2014).
- [3] A. Mulder, A. A. Van de Graaf, L. A. Robertson, and J. G. Kuenen, *FEMS Microbiol. Ecol.* **16**, 177 (1995).
- [4] C. Wen and Y. Yu, *AICHE J.* **12**, 610 (1966).
- [5] K. Ishihara, *Geotechnique* **43**, 351 (1993).
- [6] D. Sterpi, *Int. J. Geomech.* **3**, 111 (2003).
- [7] A. A. Johnson and T. E. Tezduyar, *Comput. Methods Appl. Mech. Eng.* **119**, 73 (1994).
- [8] I. Demirdžić and M. Perić, *Int. J. Numer. Methods Fluids* **10**, 771 (1990).
- [9] H. S. Udaykumar, R. Mittal, P. Rampunggoon, and A. Khanna, *J. Comput. Phys.* **174**, 345 (2001).
- [10] C. S. Peskin, *Acta Numer.* **11**, 479 (2002).
- [11] R. Mittal and G. Iaccarino, *Annu. Rev. Fluid Mech.* **37**, 239 (2005).
- [12] J. Kim, D. Kim, and H. Choi, *J. Comput. Phys.* **171**, 132 (2001).
- [13] H. Tanaka and T. Araki, *Phys. Rev. Lett.* **85**, 1338 (2000).
- [14] S. Chen and G. D. Doolen, *Annu. Rev. Fluid Mech.* **30**, 329 (1998).
- [15] A. J. C. Ladd and R. Verberg, *J. Stat. Phys.* **104**, 1191 (2001).
- [16] Z. G. Feng and E. Michaelides, *J. Comput. Phys.* **195**, 602 (2004).
- [17] D. R. Noble and J. R. Torczynski, *Int. J. Mod. Phys. C* **9**, 1189 (1998).
- [18] M. Bouzidi, M. Firdaouss, and P. Lallemand, *Phys. Fluids* **13**, 3452 (2001).
- [19] B. Chun and A. J. C. Ladd, *Phys. Rev. E* **75**, 066705 (2007).
- [20] O. Filippova and D. Hänel, *Comput. Fluids* **26**, 697 (1997).
- [21] C. S. Peskin, *J. Comput. Phys.* **25**, 220 (1977).

- [22] E. A. Fadlun, R. Verzicco, P. Orlandi, and J. Mohd-Yusof, *J. Comput. Phys.* **161**, 35 (2000).
- [23] Z. G. Feng and E. Michaelides, *J. Comput. Phys.* **202**, 20 (2005).
- [24] X. D. Niu, Y. T. Chew, and Y. Peng, *Phys. Lett. A* **354**, 173 (2006).
- [25] M. A. Gallivan, D. R. Noble, J. G. Georgiadis, and R. O. Buckius, *Int. J. Numer. Methods Fluids* **25**, 249 (1997).
- [26] I. Ginzburg and D. d’Humières, *Phys. Rev. E* **68**, 066614 (2003).
- [27] C. Pan, L. S. Luo, and C. T. Miller, *Comput. Fluids* **35**, 898 (2006).
- [28] T. Krüger, H. Kusumaatmaja, A. Kuzmin, O. Shardt, G. Silva, and E. M. Viggien, *The Lattice Boltzmann Method* (Springer, Berlin, 2017).
- [29] B. K. Cook, D. R. Noble, and J. R. Williams, *Computat. Comput.* **21**, 151 (2004).
- [30] Y. T. Feng, K. Han, and D. R. J. Owen, *Int. J. Numer. Methods Eng.* **72**, 1111 (2007).
- [31] D. R. J. Owen, C. R. Leonardi, and Y. T. Feng, *Int. J. Numer. Methods Eng.* **87**, 66 (2011).
- [32] S. A. Galindo-Torres, *Comput. Methods Appl. Mech. Eng.* **265**, 107 (2013).
- [33] G. Zhou, L. Wang, X. Wang, and W. Ge, *Phys. Rev. E* **84**, 066701 (2011).
- [34] L. Chen, Y. Yu, and G. Hou, *Phys. Rev. E* **87**, 053306 (2013).
- [35] D. Wang, C. R. Leonardi, and S. M. Aminossadati, *Comput. Math. Appl* **75**, 2593 (2018).
- [36] O. E. Strack and B. K. Cook, *Int. J. Numer. Methods Fluids* **55**, 103 (2007).
- [37] C. Tsigginos, J. Meng, X. J. Gu, and D. R. Emerson (unpublished).
- [38] X. Shan, X. F. Yuan, and H. Chen, *J. Fluid Mech.* **550**, 413 (2006).
- [39] X. He and L.-S. Luo, *Phys. Rev. E* **56**, 6811 (1997).
- [40] X. Shan and X. He, *Phys. Rev. Lett.* **80**, 65 (1998).
- [41] D. D. Joseph, T. S. Lundgren, R. Jackson, and D. A. Saville, *Int. J. Multiphase Flow* **16**, 35 (1990).
- [42] D. A. Nield and A. Bejan, *Convection in Porous Media* (Springer, Berlin, 2006), Vol. 3.
- [43] P. L. Bhatnagar, P. E. Gross, and M. Krook, *Phys. Rev.* **94**, 511 (1954).
- [44] M. Manninen, V. Taivassalo, S. Kallio, and A. Akademi, On the mixture model for multiphase flow, Tech. Rep. (Espoo, Finland, 1996).
- [45] S. Ji, *J. Geophys. Res.: Solid Earth* **109** (2004).
- [46] X. He, S. Chen, and G. D. Doolen, *J. Comput. Phys.* **146**, 282 (1998).
- [47] M. Wang, Y. T. Feng, D. Owen, and T. Qu, *Comput. Methods Appl. Mech. Eng.* **346**, 109 (2019).
- [48] Y. Han and P. A. Cundall, *Int. J. Numer. Methods Eng.* **67**, 314 (2011).
- [49] O. H. Faxén, *Proc. R. Swed. Acad. Eng. Sci.* **187**, 1 (1946).
- [50] J. Meng, X.-J. Gu, D. R. Emerson, Y. Peng, and J. Zhang, [arXiv:1803.00390](https://arxiv.org/abs/1803.00390) (2018).
- [51] F. Verhaeghe, S. Luo, and B. Blaspain, *J. Comput. Phys.* **228**, 147 (2009).
- [52] D. J. Tritton, *J. Fluid Mech.* **6**, 547 (1959).
- [53] D. J. Tritton, *Physical Fluid Dynamics*, 2nd ed. (Oxford Science Publications, Oxford, 1988).
- [54] S. Ghosh and J. M. Stockie, *Comm. Comput. Phys.* **18**, 380 (2015).
- [55] N. Bampalas and J. Graham, *J. Fluid Mech.* **616**, 205 (2008).
- [56] M. Gazzola, C. Mimeau, A. A. Tchieu, and P. Koumoutsakos, *Phys. Fluids* **24**, 043103 (2012).
- [57] R. Lagrange, X. Delaune, L. Borsoi, and J. Antunes, *J. Fluids Struct.* **77**, 102 (2018).
- [58] Q. X. Wang, *Phys. Fluids* **16**, 4412 (2004).
- [59] E. M. Viggien, The Lattice Boltzmann Method: Fundamentals and Acoustics, Ph.D. thesis, Norwegian University of Science and Technology, Trondheim, Norway (2014).