

An optimized Eulerian–Lagrangian method for two-phase flow with coarse particles: Implementation in open-source field operation and manipulation, verification, and validation

Cite as: Phys. Fluids **33**, 113307 (2021); <https://doi.org/10.1063/5.0067553>

Submitted: 17 August 2021 • Accepted: 05 October 2021 • Published Online: 03 November 2021

 Yan Zhang,  Xiao-Bing Lu and  Xu-Hui Zhang

COLLECTIONS

 This paper was selected as an Editor's Pick



View Online



Export Citation



CrossMark

ARTICLES YOU MAY BE INTERESTED IN

[Numerical simulation on transportation behavior of dense coarse particles in vertical pipe with an optimized Eulerian–Lagrangian method](#)

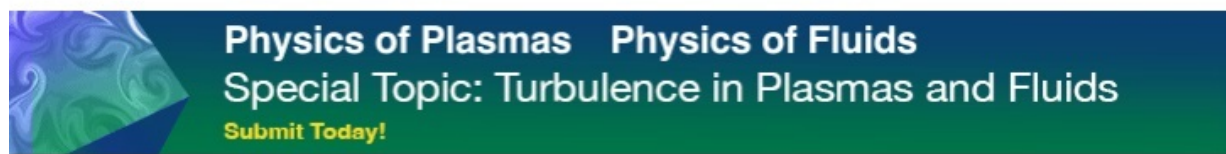
Physics of Fluids **34**, 033305 (2022); <https://doi.org/10.1063/5.0084263>

[Mechanisms of prompt and delayed ignition and combustion of explosively dispersed aluminum powder](#)

Physics of Fluids **33**, 113308 (2021); <https://doi.org/10.1063/5.0065312>

[Direct numerical simulation of the flow around a sphere immersed in a flat-plate turbulent boundary layer](#)

Physics of Fluids **33**, 115106 (2021); <https://doi.org/10.1063/5.0071878>



An optimized Eulerian-Lagrangian method for two-phase flow with coarse particles: Implementation in open-source field operation and manipulation, verification, and validation

Cite as: Phys. Fluids **33**, 113307 (2021); doi: 10.1063/5.0067553

Submitted: 17 August 2021 · Accepted: 5 October 2021 ·

Published Online: 3 November 2021



View Online



Export Citation



CrossMark

Yan Zhang,^{1,2}  Xiao-Bing Lu,^{1,2,3}  and Xu-Hui Zhang^{1,2,3,a)} 

AFFILIATIONS

¹Institute of Mechanics, Chinese Academy of Sciences, Beijing 100190, China

²School of Engineering Science, University of Chinese Academy of Sciences, Beijing 100049, China

³Southern Marine Science and Engineering Guangdong Laboratory (Guangzhou), Guangzhou 511458, China

^{a)}Author to whom correspondence should be addressed: zhangxuhui@imech.ac.cn

ABSTRACT

The solid-liquid two-phase flow with coarse particles is ubiquitous in natural phenomena and engineering practice, which is characterized by coarse particles, high particle concentration, and large particle size distribution. In this work, the numerical models describing two-phase flows are reviewed, which given that the Eulerian-Lagrangian method is applicable in this work. Then, some modified models are proposed for the situation where the conventional Eulerian-Lagrangian method is not applicable to deal with coarse particles. The continuous phase equations of liquid are solved based on the finite volume method. The pressure implicit with splitting of operators algorithm for solving the Navier-Stokes (N-S) equations of the pseudo-single-phase flow, considering phase fraction and momentum exchange source term, is proposed. The discrete coarse particle is tracked in the Lagrangian method. A virtual mass distribution function is proposed for calculating coarse particle volume fraction. A weighted function method relating to the particle size is given for the interpolation between the Eulerian and Lagrangian fields. The barycentric coordinates are introduced into the particle localization. All the modified models are algorithmically implanted in the open-source field operation and manipulation (OpenFOAM) as a new solver named coarse discrete particle method FOAM (CoarseDPMFoam). Subsequently, the applicability of the numerical simulation method is verified by some typical test cases. The proposed numerical simulation method provides new ideas and methods for the mechanism investigation and engineering application of the two-phase flow with coarse particles.

Published under an exclusive license by AIP Publishing. <https://doi.org/10.1063/5.0067553>

I. INTRODUCTION

The solid-liquid two-phase flow with coarse particles is ubiquitous and in natural phenomena and engineering practice, such as the chemical, coal, and mining industries,^{1–7} the hydraulic fracturing in unconventional oil and gas;^{8–12} the gas hydrates exploitation;^{13–16} and debris flows,^{17,18} which is characterized by coarse particles, high particle concentration, and large particle size distribution. The movement of the two-phase flow with coarse particles involves multiple geometric scales such as pipe or fracture size, configuration, and particle size, and is influenced by a variety of factors. Also, there are interactions between the particles and the liquid and among particles. These factors lead to a complex motion state of the two-phase flow, and predicting

its motion characteristics is difficult. Numerical simulation is an important research method for this problem.

In general, continuous phase and discrete phase can be used to distinguish the phases in a two-phase system. For example, solid is described as discrete phase and liquid is described as continuous phase in a solid-liquid two-phase flow. The difference between discrete phase and continuous phase is mainly that the discrete phase has a certain attribute distribution.¹⁹ The motion of the discrete phase in a two-phase system has a great influence on the flow characteristics. Hence, the investigation of discrete phase is critical for understanding the flow behavior of the entire system. In the numerical simulation of the two-phase flows, the coupling between the discrete phase and continuous

phase is divided into three types: one-way coupling, two-way coupling, and four-way coupling, which are generally related to the volume fraction of discrete phase (α_d). Figure 1 is the classification map proposed by Elghobashi.²⁰ For highly diluted flow with $\alpha_d \leq 10^{-6}$, one-way coupling should be used in which the discrete phase has a negligible effect on the continuous phase. For the particle volume fraction of $10^{-6} < \alpha_d \leq 10^{-3}$, the influence of the discrete phase on the continuous phase needs to be considered. The degree of influence depends on τ_p/τ_K , where $\tau_p = \rho_d d^2 / (18 \rho_c \nu_c)$ is the particle reaction time, $\tau_K = (\nu_c/\varepsilon)^{1/2}$ is the Kolmogorov time scale, ρ_d and ρ_c are the density of the discrete phase and continuous phase, respectively, d is the particle diameter, ν_c is the kinematic viscosity of the continuous phase, and ε is the turbulence dissipation rate. Small values of τ_p increase turbulence dissipation rate, while large values of τ_p enhance turbulence production. When $\alpha_d > 10^{-3}$, the interaction between particles such as particle collision and bonding must be considered due to the large number of particles per unit volume, which are referred to as four-way coupling.

First, a review of the numerical models describing multiphase flows is given. The models describing multiphase flows can be divided into microscopic model, mesoscopic model, and macroscopic model.²¹ Simulation based on microscopic model is also called direct numerical simulation (DNS) of the multiphase flow. DNS of multiphase flow mainly focuses on spatial resolution rather than solution accuracy. The forces on particles are obtained by integrating the surface of the particles, which is the true forces on the particles, rather than using a force model such as WenYu drag model.²² The microscopic model is suitable for micro-scale problems such as the movement of dozens of particles or bubbles. The macroscopic model focuses on the macroscopic motion behavior of the multiphase flow. It is an Eulerian-Eulerian method where variables of both the discrete phase and continuous phase are obtained by solving locally averaged Eulerian transport equation. Two fluid model (TFM)²³ is the most popular macroscopic model in the field of two-phase flows with the advantages of high solution efficiency and wide application range. However, it cannot be used to predict polydispersity effects and is not applicable when the Knudsen number (Kn) of the discrete phase is

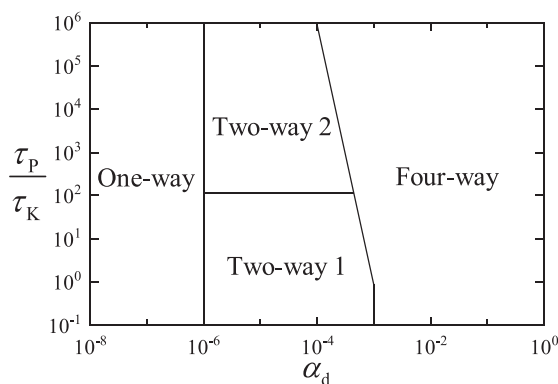


FIG. 1. Classification of coupling form between the discrete phase and continuous phase,²⁰ where two-way coupling 1 represents enhancing turbulence dissipation and two-way coupling 2 represents enhancing turbulence production. [Reproduced with permission from Elghobashi, "On predicting particle-laden turbulent flows," *Appl. Sci. Res.* **52**, 309–329 (1994). Copyright 2021 Springer Nature.]

larger than 0.1.²⁴ The mesoscopic model is proposed to overcome the shortcomings that the macroscopic model can only simulate the particle field distribution and the microscopic model has a large amount of computation. It is an Eulerian-Lagrangian method where the locally averaged equations of motion for the continuous phase are solved in an Eulerian framework and the discrete phase is tracked in a Lagrangian method by solving Newton's equations of motion. Models for the interaction force and effective stress in the continuous phase need to be defined. The computational fluid dynamics-discrete element method (CFD-DEM)^{25–27} and the multiphase particle-in-cell (MPPIC) method^{28,29} are all mesoscopic model. The difference between the two methods lies in the way for handling particle collisions. A particle stress model is proposed in MPPIC to reduce the calculation cost, which can be expressed as

$$\tau_{\text{MPPIC}} = \frac{P_d \alpha_d^\beta}{\max[\alpha_{d,\max} - \alpha_d, \gamma(1 - \alpha_d)]}, \quad (1)$$

where τ_{MPPIC} is the particle stress, P_d is a constant with pressure dimension, $\alpha_{d,\max}$ is the maximum volume fraction of particles, and β and γ are constants. In addition, the coupling of the CFD and the population balance equation (PBE) is also a mesoscopic model, which can be used to describe the evolution of a population of particles by the particle probability distribution function (PDF)

$$\frac{\partial \phi}{\partial t} + \nabla \cdot (\phi \mathbf{U}_d) + \nabla_{\mathbf{U}_d} \cdot (\phi \mathbf{a}) = 0, \quad (2)$$

where $\phi(\mathbf{x}, \mathbf{U}_d, \rho_d, V_d, t)$ is the PDF, \mathbf{x} is the position vector of the particle, \mathbf{U}_d is the particle velocity, V_d is the particle volume, and \mathbf{a} is the particle acceleration. The PBE is usually solved by method of moments (MOM)³⁰ due to its complex characteristics. As a result, the coupling of CFD and PBE is also called Eulerian-MOM method. The Eulerian transport equations for locally averaged variables of the discrete phase can also be obtained from Eq. (2). The current numerical simulation methods of multiphase flows are summarized in Fig. 2. The capabilities of various numerical simulation methods are summarized in Table I by referring Refs. 21 and 24.

To describe the solid-liquid two-phase flow in this paper, which is characterized by coarse particles, high particle concentration, and large particle size distribution, the Eulerian-Lagrangian method is used in this paper based on the above analysis. In contrast to other methods, it can handle a wide range of particle size distributions in both diluted and dense flows and capture nonlinear, multiscale interactions as well as non-equilibrium effects.³¹ The coarse particles may be eight times larger than the fine particles. In conventional Eulerian-Lagrangian method, the cell size for the continuous phase is chosen as five times of the largest particles to achieve smoothed particle volume fraction field. There is no sufficient scale resolution to capture the properties of the flow field properly if the cell size is set based on the size of the coarse particles.^{32,33} In addition, the conventional Eulerian-Lagrangian method is inappropriate due to inapplicability of the interpolation method between the Eulerian and Lagrangian field with coarse particles. Hence, the conventional Eulerian-Lagrangian method needs to be optimized.

The particle centroid method (PCM) is widely used to calculate the continuous quantities of the particles such as the particle volume fraction in conventional Eulerian-Lagrangian method by dividing the

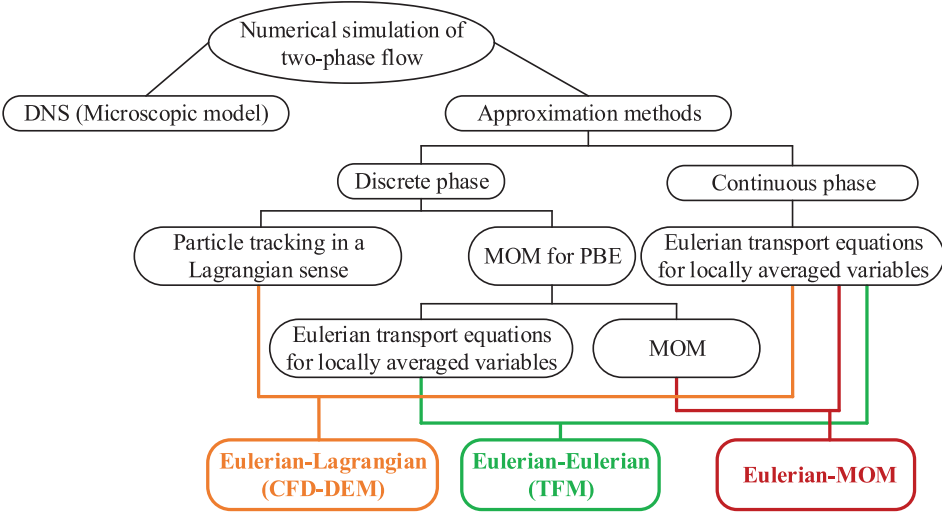


FIG. 2. Algorithms for multiphase flow.

added particle volume by the cell volume ($\alpha_d = \sum_{i=1}^n V_{di}/V_{\text{cell}}$, where V_{cell} is the cell volume). Deen *et al.*³⁴ found that the PCM can guarantee accuracy when the ratio of particle size to cell size is less than 0.2. The error can be up to 50 % when the particle centroid is located near the cell boundaries.³⁵ It means that if several such particles locate near the cell face and the particles size is close to or larger than the cell size, nonphysically conditions occur in this cell. For example, the particle volume fraction is larger than one and the fluid volume fraction is smaller than zero. It will make the governing equations no sense and result in non-smooth particle volume fraction with large gradients. Except for the shortcomings mentioned above, the PCM, which is seemingly unsophisticated, works well, especially when the particle size is much smaller than the cell size. Due to the simplicity and easy code implementation, the PCM is one of the most widely used procedures. To deal with coarse particles, Goniva *et al.*³⁶ proposed an indicator point method by arranging a certain number of points inside each particle, and then, the particle volume fraction is calculated by determining the position of the indicator point. However, this method

requires a large amount of computation. The fictitious particle method proposed by Tsuji, Higashida, and Okuyama³⁷ is similar to the indicator point method. The two-grid method³³ can also be used to deal with coarse particles by using coarse grids for discrete phase and fine grids for continuous phase. It is relatively easy to be implemented in structured grids, but difficult in unstructured grids. The statistical kernel method (SKM)^{38–40} is also a recognized method to deal with coarse particles by diffusing the coarse particles' effects to the surrounding grids. For example, the Gaussian kernel³⁸ is expressed as

$$h(\mathbf{x} - \mathbf{x}_i) = \frac{1}{(b^2\pi)^{\frac{3}{2}}} \exp \left[-\frac{(\mathbf{x} - \mathbf{x}_i)^T(\mathbf{x} - \mathbf{x}_i)}{b^2} \right], \quad (3)$$

where \mathbf{x} is the position vector, \mathbf{x}_i is the particle centroid, and b is the bandwidth of the Gaussian kernel. The SKM is easy to implement and fast in calculation, but special processing is required at the boundary. Based on the idea of SKM, we propose a virtual mass distribution function (VMDF) method to deal with coarse particles in the Euler–Lagrangian method.

TABLE I. Capabilities of the Eulerian-Eulerian (E-E) method, Eulerian-Lagrangian (E-L) method, and Eulerian-MOM (E-M) method.

Methods	Can do	Cannot do
E-E	Applicable to dense flow; Can simulate laboratory- and pilot-scale processes; Similar equations for two phases;	Cannot predict polydispersity effects; Cannot predict trajectory crossing; Inapplicable to highly diluted flow; Difficult to account for interparticle forces;
E-L	Tracking particle position; Particle size distribution included; Interparticle forces included; Easy to implementation;	Requires constitutive model for fluid–particle interaction force; Assumed soft particle for reducing calculation cost; Impractical for commercial-scale processes;
E-M	Fast processing of coalescence and breakage; Can simulate laboratory-scale and commercial-scale systems; Particle size distribution included;	Complex and difficult to converge; Cannot yet handle interparticle forces;

The continuous phase field, such as continuous phase velocity and pressure, at the particle centroid needs to be obtained during particle tracking. However, the continuous phase field is usually stored at cell center, and the particle centroid does not coincide with the cell center. Therefore, the field at the cell center needs to be interpolated to the particle centroid. There are usually two interpolation methods. The first one is to make the cell center value equal to the particle centroid value. The second one is to interpolate cell node value to the particle centroid based on the distance between the particle centroid and the cell nodes. However, these two methods are not suitable for coarse particles. When coarse particles occupy multiple cells, the continuous phase field at the particle centroid should not only be related to the cell where it is localized, but also to all the cells it occupies. A new interpolation method will be given in this paper for coarse particles.

The computational domain is discretized using cells to obtain the numerical solution, so the particles have to be localized and the cell where a particle is has to be determined. Haselbacher, Najjar, and Ferry⁴¹ proposed a simple and robust particle localization algorithm by determining for each cell face if the particle is on the right side of the face. The outward unit normal of each cell face (\mathbf{n}_i) is computed and then checks whether

$$(\mathbf{f}_i - \mathbf{p}) \cdot \mathbf{n}_i \geq 0, \quad (4)$$

where \mathbf{f}_i is the centroid of the i th cell face, and \mathbf{p} is the particle position. The particle is considered inside the cell if the test is passed for all cell faces. The disadvantage of this method is that the calculation speed will decrease significantly as the number of cells increases. The interaction method^{42–44} is another popular particle localization method in which the faces where the particle is passing through are searched. The method takes the advantage of the previous particle's cell location to deduce the next cell where the particle is localized. It is very efficient as it requests less calculation cost than the traditional localization method. There is a problem with this method that the relative floating point error related to the calculation of the location of the intersection can be very large. The particle may be lost if the error means the intersection cannot be found. To avoid this problem, the barycentric coordinates are used.

In this work, conventional Eulerian-Lagrangian method for two-phase flow is optimized to deal with coarse particles. Some modified models are proposed. For example, the VMDF method is proposed to calculate the volume fraction field of the coarse particles, the weight function method (WFM) relating to the particle size is proposed for the data exchange between the Eulerian and Lagrangian framework, and the barycentric coordinates particle tracking method (PTM) is applied to the particle localization. The pressure implicit with splitting of operators (PISO) algorithm is improved to make it suitable for solving the Navier–Stokes (N–S) equations for pseudo-single-phase flow with volume fraction field and momentum exchange source term. The algorithm is implanted in the open-source field operation and manipulation (OpenFOAM), and a new solver named coarse discrete particle method FOAM (CoarseDPMFoam) is developed.

The remainder of the paper is as follows: in Sec. II, the governing equations, the modified models, and the numerical discretization method are discussed. The solver development and implementation are also described. In Sec. III, the solver is validated against three test cases. Finally, conclusions are drawn in Sec. IV.

II. METHODS

A. Model equations

1. Discrete phase equations

The motion state of a particle in a continuous phase is translation and rotation. The particle will collide with other particles as well as walls and interact with the continuous phase during its movement. In fact, the particle is not only affected by the surrounding fluid and particles, but also the far fluid and particles due to the proliferation of disturbance waves.⁴⁵ However, in the numerical simulation, if a numerical time step is set less than a critical value, the diffusion range of the disturbance is about the same magnitude as the surrounding particle and fluid (continuous phase) range in a time step.⁴⁶ Hence, the force on the particle can be directly regarded as the forces from the surrounding fluid and particles at all times. The movement of each single particle is governed by Newton's second law of motion.⁴⁷ The particle motion equation is

$$m_d \frac{d\mathbf{U}_d}{dt} = m_d \mathbf{g} + \mathbf{F}_{dc} + \mathbf{F}_{col}, \quad (5)$$

$$I_d \frac{d\boldsymbol{\omega}_d}{dt} = \mathbf{M}, \quad (6)$$

where \mathbf{F}_{dc} is the interaction force between the particle and the fluid (continuous phase), \mathbf{F}_{col} is the collision force between particles and the collision force between the particle and the wall, m_d is the mass of the particle, \mathbf{g} is the gravity acceleration, $\boldsymbol{\omega}_d$ is the angular velocity of the particle, $I_d = m_d d^2/10$ is the moment of inertia, d is the diameter of the particle, and \mathbf{M} is the collision torque between particles and the collision torque between the particle and the wall.

Particles in solid–liquid two phase flow are subjected to several different forces,^{48,49} such as the drag force, the gravity force, the buoyancy force, the pressure gradient force, the virtual mass force, the Basset force, the Magnus force, and the Saffman force. The drag force is written as (in one-dimensional form)

$$F_d = K_d (U_c - U_d), \quad (7)$$

where F_d is the drag force, K_d is the coefficient of the drag force, U_c is the velocity of fluid, and U_d is the velocity of particle. For simplicity, K_d is written as²²

$$K_d = V_d \frac{3}{4} C_d \frac{\rho_c |U_c - U_d|}{d} \alpha_c^{-2.65}, \quad (8)$$

where V_d is the particle volume and C_d is drag coefficient. The pressure gradient force is written as

$$F_p = V_d \frac{\partial P}{\partial x}, \quad (9)$$

where F_p is the pressure gradient force and P is pressure. The virtual mass force is written as (for sphere particles)

$$F_v = \frac{1}{2} \pi \rho_c d^3 \left(\frac{dU_c}{dt} - \frac{dU_d}{dt} \right), \quad (10)$$

where F_v is the virtual mass force. The Basset force is written as

$$F_B = \frac{3}{2} d^2 \sqrt{\pi \rho_c \mu_c} \int_{t_0}^t \frac{\frac{dU_c(\tau)}{dt} - \frac{dU_d(\tau)}{dt}}{\sqrt{t - \tau}} d\tau, \quad (11)$$

where μ_c is the fluid viscosity. The Magnus force can be written as

$$F_M = \frac{\pi}{8} \rho_c d^3 \omega (U_c - U_d), \quad (12)$$

where F_M is the Magnus force, and ω is the angular velocity. The Saffman force can be written as

$$F_S = 1.62 d^2 \sqrt{\rho_c \mu_c} (U_c - U_d) \sqrt{\frac{dU_c}{dy}}, \quad (13)$$

where F_S is the Saffman force. The ratio of Magnus force to the drag force is

$$\begin{aligned} \frac{F_M}{F_D} &= \frac{\frac{\pi}{8} \rho_c d^3 \omega (U_c - U_d)}{V_d \frac{3}{4} C_d \frac{\rho_c |U_c - U_d|}{d} \alpha_c^{-2.65} (U_c - U_d)} \\ &= \frac{d\omega}{C_D |U_c - U_d| \alpha_c^{-2.65}} \approx 0.01. \end{aligned} \quad (14)$$

The ratio of Saffman force to the drag force is

$$\begin{aligned} \frac{F_S}{F_D} &= \frac{1.62 d^2 \sqrt{\rho_c \mu_c} (U_c - U_d) \sqrt{\frac{dU_c}{dy}}}{V_d \frac{3}{4} C_d \frac{\rho_c |U_c - U_d|}{d} \alpha_c^{-2.65} (U_c - U_d)} \\ &= \frac{1.62 d^2 \sqrt{\rho_c \mu_c} \sqrt{\frac{dU_c}{dy}}}{\frac{\pi}{8} C_D \rho_c |U_c - U_d| \alpha_c^{-2.65}} \approx 0.005. \end{aligned} \quad (15)$$

Assuming that the relative acceleration of the particles is constant and approximately expressed in the differential form as follows:

$$\frac{dU_c(\tau)}{dt} - \frac{dU_d(\tau)}{dt} \approx \frac{U_c - U_d}{t - t_0} = \text{const.} \quad (16)$$

Then, the ratio of basset force to the drag force is

$$\begin{aligned} \frac{F_B}{F_D} &= \frac{\frac{3}{2} d^2 \sqrt{\pi \rho_c \mu_c} \int_{t_0}^t \frac{\frac{dU_c(\tau)}{dt} - \frac{dU_d(\tau)}{dt}}{\sqrt{t - \tau}} d\tau}{V_d \frac{3}{4} C_d \frac{\rho_c |U_c - U_d|}{d} \alpha_c^{-2.65} (U_c - U_d)} \\ &\approx \frac{24 d^3 \sqrt{\rho_c \mu_c}}{\sqrt{\pi} C_d \rho_c \sqrt{t - t_0} |U_c - U_d| \alpha_c^{-2.65}} \approx O(10^{-6}). \end{aligned} \quad (17)$$

As a result, the Magnus force, the Saffman force, and the basset force are neglected here. The ratio of pressure gradient force to the drag force is

$$\frac{F_p}{F_D} = \frac{V_d \frac{\partial P}{\partial x}}{V_d \frac{3}{4} C_d \frac{\rho_c |U_c - U_d|}{d} \alpha_c^{-2.65} (U_c - U_d)} \approx O(1). \quad (18)$$

The ratio of virtual mass force to the drag force is

$$\frac{F_V}{F_D} = \frac{\frac{1}{2} \frac{\pi}{6} \rho_c d^3 \left(\frac{dU_c}{dt} - \frac{dU_d}{dt} \right)}{V_d \frac{3}{4} C_d \frac{\rho_c |U_c - U_d|}{d} \alpha_c^{-2.65} (U_c - U_d)}. \quad (19)$$

For steady particle transportation in the engineering practice, such as the hydraulic lifting in the chemical, coal and mining industries, the hydraulic fracturing in unconventional oil and gas, and $d(U_c - U_d)/dt \rightarrow 0$, $F_V/F_D \rightarrow 0$. In the case of $d(U_c - U_d)/dt$ do not tend to 0, the virtual mass force need to be considered. For purposes of the current effort, only the effect of drag force and pressure gradient force, which refers to force arising from a pressure gradient (both static and non-static) are considered since these contributions are the most important for the systems considered in this work. Hernandez⁵⁰ proposed two different forms of decomposition of the fluid-particle interaction force (\mathbf{F}_{dc})

$$\mathbf{F}_{dc} = K_d (\mathbf{U}_c - \mathbf{U}_d) - V_d \nabla P, \quad (20)$$

$$\mathbf{F}_{dc} = \frac{K_d (\mathbf{U}_c - \mathbf{U}_d)}{\alpha_c} - \rho_c V_d \left(\mathbf{g} - \frac{D\mathbf{U}_c}{Dt} \right), \quad (21)$$

where \mathbf{U}_c is the velocity of the fluid (continuous phase) and α_c is the volume fraction of the fluid. The pressure gradient force in Eq. (20) includes the force due to the pressure gradient in a static fluid as well as the effect of relative motion on local pressure gradient. However, the pressure gradient force $\rho_c V_d (\mathbf{g} - D\mathbf{U}_c/Dt)$ in Eq. (21) only includes the effect of the static pressure gradient. As a result, the drag force in Eq. (21) is divided by the fluid volume fraction α_c .⁵⁰

The ErgunWenYu drag model⁵¹ is used in this work. To cover the whole range of the fluid volume fraction, Gidaspow⁵¹ combined the WenYu drag model²² and the Ergun drag model.⁵² For a fluid volume fraction greater than 0.8, the WenYu drag model is used. For a fluid volume fraction less than 0.8, the Ergun drag model is used. The ErgunWenYu drag model is expressed by

$$K_d = \begin{cases} \alpha_c < 0.8, & V_d \left(150 \frac{(1 - \alpha_c) \mu_c}{\alpha_c d^2} + 1.75 \frac{\rho_c |U_c - U_d|}{d} \right), \\ \alpha_c \geq 0.8, & V_d \frac{3}{4} C_d \frac{\rho_c |U_c - U_d|}{d} \alpha_c^{-2.65}, \end{cases} \quad (22)$$

$$C_d = \begin{cases} Re < 1000, & \frac{24(1 + 0.15 Re^{0.687})}{Re}, \\ Re \geq 1000, & 0.424, \end{cases} \quad (23)$$

$$Re = \frac{\rho_c |U_c - U_d| d}{\mu_c}, \quad (24)$$

where Re is the relative Reynolds number.

In general, the contact between two particles during colliding each other is a finite area because of the deformation of the particles. In the numerical simulation, if the two particles overlap; that is, the distance between the centers of the two particles is less than the sum of the radius of the two particles, the two particles are considered colliding each other, and the collision force is subsequent calculated from the overlap ("deformation").⁵³ The particle collision force is calculated based on the "soft model."^{46,54} More details about the soft model in OpenFOAM can be found in Refs. 55 and 45.

Substituting Eq. (21) into Eq. (5), the particle motion equation can be rewritten as

$$m_d \frac{d\mathbf{U}_d}{dt} = m_d \mathbf{g} \left(1 - \frac{\rho_c}{\rho_d} \right) + \frac{K_d (\mathbf{U}_c - \mathbf{U}_d)}{\alpha_c} + \rho_c \frac{m_d D\mathbf{U}_c}{\rho_d Dt} + \mathbf{F}_{col}. \quad (25)$$

2. Continuous phase equations

The motion of the fluid phase is solved in terms of the locally averaged variables over the computational mesh by solving the continuity and momentum equation of an incompressible fluid. The continuity and the momentum conservation equation of the fluid phase are⁵⁶

$$\begin{aligned} \frac{\partial}{\partial t}(\alpha_c \rho_c) + \nabla \cdot (\alpha_c \rho_c \mathbf{U}_c) &= 0, \\ \frac{\partial}{\partial t}(\alpha_c \rho_c \mathbf{U}_c) + \nabla \cdot (\alpha_c \rho_c \mathbf{U}_c \mathbf{U}_c) - \nabla \cdot (\alpha_c \rho_c \boldsymbol{\tau}_c) &= -\nabla P + \rho_c \mathbf{g} - \mathbf{f}_{dc}, \end{aligned} \quad (26)$$

where $\boldsymbol{\tau}_c$ is the stress tensor of the fluid and \mathbf{f}_{dc} is the momentum exchange term between the particle and the fluid. The current work focuses on the numerical treatment of the Eulerian-Lagrangian method in CFD simulations for solid particle motions. Hence, the heat transfer and mass transfer are neglected. The governing equations for the continuous phase in the Eulerian-Lagrangian method are similar to those in the Eulerian-Eulerian method. It is assumed that the pressure drop only exists in the continuous phase equations because the solid particle is tracked under Lagrangian framework. As a result, the first and second term of the right-hand side of Eq. (27) is not multiplied by the fluid volume fraction. The stress tensor of the fluid is expressed as

$$\boldsymbol{\tau}_c = \nu_c (\nabla \mathbf{U}_c + \nabla \mathbf{U}_c^T) - \frac{2}{3} \nu_c \nabla \cdot \mathbf{U}_c \mathbf{I}, \quad (28)$$

where \mathbf{I} is the identity matrix. The forces exerted by the fluid on the particles are calculated on a per-particle basis and an opposite force is applied to the fluid. Based on Eq. (20), the force per unit volume exerted by the fluid on the particles in a cell is given by

$$\begin{aligned} \mathbf{f}_{dc} &= \frac{1}{V_{\text{cell}}} \sum_{i=1}^n \mathbf{F}_{dc} = \frac{1}{V_{\text{cell}}} \sum_{i=1}^n K_d (\mathbf{U}_c - \mathbf{U}_d) - \frac{1}{V_{\text{cell}}} \sum_{i=1}^n V_d \nabla P \\ &= \mathbf{f}_d - \alpha_d \nabla P, \end{aligned} \quad (29)$$

$$\mathbf{f}_d = \frac{1}{V_{\text{cell}}} \sum_{i=1}^n K_d (\mathbf{U}_c - \mathbf{U}_d), \quad (30)$$

where \mathbf{f}_d is the momentum exchange term of the drag force, $\alpha_d = \sum_{i=1}^n V_d / V_{\text{cell}}$ is the volume fraction of the particle, n is the number of particles in a cell, and i is the particle index.

The fluid phase pressure P consists of a static fluid pressure $\rho \mathbf{g} \cdot \mathbf{h}$ and relative motion pressure P_{rgh} . The fluid phase pressure gradient in Eq. (27) can be change to the bellow form to make the definition of boundary conditions easier as well as reduce the false velocity caused by hydrostatic pressure under non-orthogonal grids⁵⁷

$$\nabla P = \nabla P_{\text{rgh}} + \rho \mathbf{g} + \mathbf{g} \cdot \mathbf{h} \nabla \rho, \quad (31)$$

where \mathbf{h} is the position vector of the cell center, and $\rho = \alpha_c \rho_c + \alpha_d \rho_d$ is the mixture density of the fluid and the particle, and α_c and α_d satisfy the relationship $\alpha_c + \alpha_d = 1$. Substituting Eqs. (29) and (31) into Eq. (27) and extracting the density item from Eqs. (26) and (27), the continuity and momentum conservation equation of the fluid can be rewritten as

$$\begin{aligned} \frac{\partial \alpha_c}{\partial t} + \nabla \cdot (\alpha_c \mathbf{U}_c) &= 0, \\ \frac{\partial}{\partial t}(\alpha_c \mathbf{U}_c) + \nabla \cdot (\alpha_c \mathbf{U}_c \mathbf{U}_c) - \mathbf{U}_c \left(\frac{\partial \alpha_c}{\partial t} + \nabla \cdot (\alpha_c \mathbf{U}_c) \right) \\ - \nabla \cdot (\alpha_c \boldsymbol{\tau}_c) &= -\alpha_c \nabla \frac{P_{\text{rgh}}}{\rho_c} - \left(\alpha_c \frac{\rho}{\rho_c} - 1 \right) \mathbf{g} - \frac{\alpha_c}{\rho_c} \mathbf{g} \cdot \mathbf{h} \nabla \rho - \frac{\mathbf{f}_d}{\rho_c}. \end{aligned} \quad (32)$$

Mathematically, the third term of the left-hand side of Eq. (33) is zero. However, the equation is discretized based on the finite volume method (FVM) in order to be solved numerically. Equation (33) might actually be different without the third term on the left-hand side, as the discretized continuity equation might not equal to zero. Hence, the third term of left-hand side of Eq. (33) is retained because it helps to maintain the boundedness of the solution variable and promotes better convergence.^{58,59} In addition, the first term of the right-hand side of Eq. (33) is multiplied by the fluid volume fraction due to pressure gradient force added to the pressure gradient term.

B. Code implementation of governing equations

1. Continuous phase

It can be seen that Eq. (33) has no diffusion flux for implicitly discretizing. However, the discretization of equations based on the FVM generates a numerical diffusion.⁵⁸ Therefore, the stress tensor of the fluid in Eq. (33) is decomposed into two terms for numerical considerations^{60,61}

$$\begin{aligned} \boldsymbol{\tau}_c &= \boldsymbol{\tau}_c - \nu_c \nabla \mathbf{U}_c + \nu_c \nabla \mathbf{U}_c \\ &= \nu_c (\nabla \mathbf{U}_c + \nabla \mathbf{U}_c^T) - \frac{2}{3} \nu_c \nabla \cdot \mathbf{U}_c \mathbf{I} - \nu_c \nabla \mathbf{U}_c + \nu_c \nabla \mathbf{U}_c \\ &= \left(\nu_c \nabla \mathbf{U}_c^T - \frac{2}{3} \nu_c \nabla \cdot \mathbf{U}_c \mathbf{I} \right) + \nu_c \nabla \mathbf{U}_c \\ &= \boldsymbol{\tau}'_c + \nu_c \nabla \mathbf{U}_c. \end{aligned} \quad (34)$$

The diffusion term concerning \mathbf{U}_c in Eq. (34) is artificially derived, which can be implicitly discretized to decrease the numerical diffusion. Substituting Eq. (34) into Eq. (33), the discretized momentum equation of the fluid can be written as

$$\begin{aligned} \frac{\partial}{\partial t}(\alpha_c \langle \mathbf{U}_c \rangle) + \nabla \cdot (\alpha_c \mathbf{U}_c \langle \mathbf{U}_c \rangle) - \langle \mathbf{U}_c \rangle \left(\frac{\partial \alpha_c}{\partial t} + \nabla \cdot (\alpha_c \mathbf{U}_c) \right) \\ - \nabla \cdot (\alpha_c \boldsymbol{\tau}'_c) - \nabla \cdot (\alpha_c \nu_c \nabla \langle \mathbf{U}_c \rangle) \\ = -\alpha_c \nabla \frac{P_{\text{rgh}}}{\rho_c} - \left(\alpha_c \frac{\rho}{\rho_c} - 1 \right) \mathbf{g} - \frac{\alpha_c}{\rho_c} \mathbf{g} \cdot \mathbf{h} \nabla \rho - \frac{\mathbf{f}_d}{\rho_c}, \end{aligned} \quad (35)$$

where $\langle \rangle$ means implicit discretization. Equation (35) only gives a velocity prediction of the fluid, which will be corrected after the pressure is updated because the pressure gradient term, the gravity force term, and the momentum exchange term of drag force are explicitly discretized. The discretized algebraic equation of Eq. (35) can be written as

$$\begin{aligned} A_P \mathbf{U}_{c,P} + \sum A_N \mathbf{U}_{c,N} - S_P \\ = -(\alpha_c \nabla P_{\text{rgh}})_P - \left[\left(\alpha_c \frac{\rho}{\rho_c} - 1 \right) \mathbf{g} - \frac{\alpha_c}{\rho_c} \mathbf{g} \cdot \mathbf{h} \nabla \rho - \frac{\mathbf{f}_d}{\rho_c} \right]_P, \end{aligned} \quad (36)$$

where A_p and A_N are the diagonal and off-diagonal elements of the coefficients matrix, respectively, $\mathbf{U}_{c,p}$ is the fluid velocity of the cell P , $\mathbf{U}_{c,N}$ is the fluid velocity of the adjacent cell, and S_p is the source of the algebraic equation. The discretized form of the pressure gradient term is not given in Eq. (36) for preventing the pressure oscillations in the spirit of the Rhie–Chow interpolation.⁶² Zhang, Zhao, and Bayyuk⁶³ proposed that the gravity force term is better included in the pressure Poisson equation. Hence, the gravity force term is not merged into S_p . There are many different methods to link the momentum exchange term of drag force to the velocity of the fluid after the fluid velocity is predicted based on Eq. (35) such as the fully explicit method, the partially implicit method, the fully implicit method, and the partial elimination method.^{64,65} Miller and Miller⁶⁶ studied the difference of these methods. In the present paper, the fully explicit method is used because the particle is larger than the cell. The fully explicit method treats the momentum exchange term of drag force as a constant term. Equation (36) can be written as

$$\mathbf{U}_{c,p} = \mathbf{HbyA}_p - \frac{\alpha_{c,p}}{A_p} (\nabla P_{\text{rgh}})_p - \frac{1}{A_p} \left[\left(\alpha_c \frac{\rho}{\rho_c} - 1 \right) \mathbf{g} - \frac{\alpha_c}{\rho_c} \mathbf{g} \cdot \mathbf{h} \nabla \rho - \frac{\mathbf{f}_d}{\rho_c} \right], \quad (37)$$

where \mathbf{HbyA}_p is the finite volume representation of the spatial convective and diffusive fluxes of the phase momentum^{59,67} and is expressed as

$$\mathbf{HbyA}_p = \frac{1}{A_p} \left(- \sum A_N \mathbf{U}_{c,N} + S_p \right). \quad (38)$$

The velocity field, $\mathbf{U}_{c,p}$, does not satisfy the continuity constraint. Hence, a pressure Poisson equation needs to be given to correct the velocity field. Substituting Eq. (37) into Eq. (32), the pressure Poisson equation can be constructed as follows:

$$\nabla \cdot \left(\frac{\alpha_c^2}{A_p} \nabla P_{\text{rgh}} \right) = \frac{\partial \alpha_c}{\partial t} + \nabla \cdot \left(\alpha_c \mathbf{HbyA} - \frac{\alpha_c}{A_p} \left(\alpha_c \frac{\rho}{\rho_c} - 1 \right) \mathbf{g} - \frac{\alpha_c^2}{A_p \rho_c} \mathbf{g} \cdot \mathbf{h} \nabla \rho - \frac{\alpha_c \mathbf{f}_d}{A_p \rho_c} \right). \quad (39)$$

Equation (39) is different from the basic pressure Poisson equation, which is motivated by numerical stability considerations. In OpenFOAM, except for the boundary surface, all variables are stored at the cell centroid. The variables at the boundary are stored at the centroid of the boundary face. The Rhie–Chow interpolation⁶² is adopted to avoid pressure checker-boarding. The pressure Poisson equation is multiplied by α_c^2 instead of α_c .

The continuous phase equations are solved as following steps:

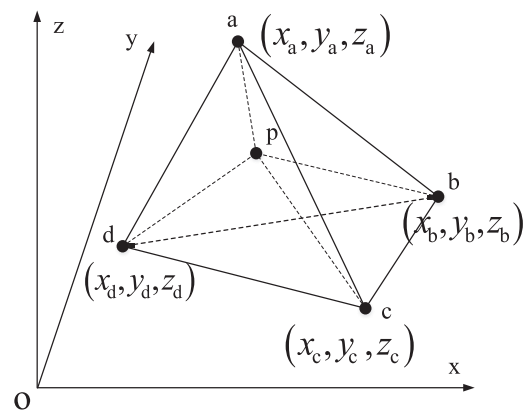
1. Solving Eq. (35) to obtain the predicted velocity field;
2. Constructing the \mathbf{HbyA} based on Eq. (38);
3. Solving Eq. (39) to obtain the pressure field;
4. Updating the velocity field based on Eq. (37);
5. Looping 2–4 steps to get the convergence results.

2. Discrete phase

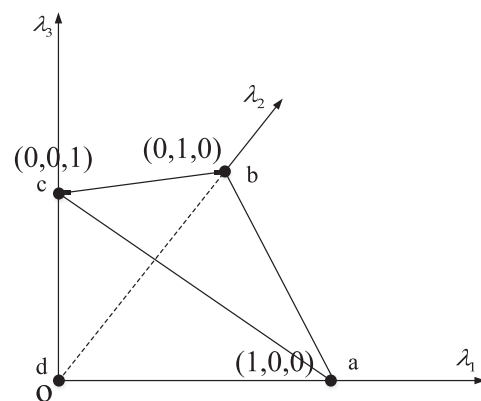
Particle tracking is a process of following a particle as it progresses through a fluid cell. The PTM tracks the particle through a

polyhedral mesh. The polyhedral mesh (tetrahedra, pyramids, prisms, and hexahedra) is decomposed into tetrahedra in OpenFOAM, and the particle is at any time associated with the tetrahedron containing it. In this paper, the barycentric coordinates, which are local to the tetrahedron related to the particle, are used to store the particle position instead of the globe coordinates. The concept of barycentric coordinates is well established and widely applied to computer graphics. In addition, some researchers have taken the coordinates for particle tracking in Eulerian-Lagrangian simulation.^{68,69} A short recapitulation below aims to summarize the usage of the barycentric coordinates of this paper in the particle tracking.

First, the particle position with respect to a tetrahedron needs to be defined. Considering a tetrahedral element, which is defined by its vertices **a**, **b**, **c**, and **d** [Fig. 3(a)], with an arbitrary point **p** within it. The global coordinates of the element vertices and point **p** (x_p, y_p, z_p) are given. The arbitrary point **p** can be expressed using a set of local coordinates ($\lambda_1, \lambda_2, \lambda_3, \lambda_4$) associated with the element vertices. The local coordinates are defined as barycentric coordinates with respect to the element if they exhibit the following properties:



(a)



(b)

FIG. 3. The barycentric coordinates, (a) point **p** in tetrahedral element \widetilde{abcd} , and (b) the tetrahedral element after transformation.

- (1) $\lambda_i \geq 0, i = 1, 2, 3, 4$ for all points $\mathbf{p} \in$ element;
- (2) $\lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 = 1$ for all points \mathbf{p} ;
- (3) $\mathbf{p} = \lambda_1 \mathbf{a} + \lambda_2 \mathbf{b} + \lambda_3 \mathbf{c} + \lambda_4 \mathbf{d}$.

The point \mathbf{p} in the element divides the tetrahedron into four sub-tetrahedrons: V_{pbcd} , V_{pacd} , V_{pabd} , and V_{pabc} . Then, the barycentric coordinates of the arbitrary point \mathbf{p} are given by

$$\lambda_1 = \frac{V_{pbcd}}{V_{abcd}} = 1 - \lambda_2 - \lambda_3 - \lambda_4, \quad (40)$$

$$\lambda_2 = \frac{V_{pacd}}{V_{abcd}} = \frac{6V_{pacd}}{6V_{abcd}} = \frac{\mathbf{ap} \cdot (\mathbf{ac} \times \mathbf{ad})}{\mathbf{ab} \cdot (\mathbf{ac} \times \mathbf{ad})}, \quad (41)$$

$$\lambda_3 = \frac{V_{pabd}}{V_{abcd}} = \frac{6V_{pabd}}{6V_{abcd}} = \frac{\mathbf{ap} \cdot (\mathbf{ad} \times \mathbf{ab})}{\mathbf{ab} \cdot (\mathbf{ac} \times \mathbf{ad})}, \quad (42)$$

$$\lambda_4 = \frac{V_{pabc}}{V_{abcd}} = \frac{6V_{pabc}}{6V_{abcd}} = \frac{\mathbf{ap} \cdot (\mathbf{ab} \times \mathbf{ac})}{\mathbf{ab} \cdot (\mathbf{ac} \times \mathbf{ad})}, \quad (43)$$

where V_{pbcd} , V_{pacd} , V_{pabd} , and V_{pabc} are the volume of the four sub-tetrahedrons, $\mathbf{ap} = \mathbf{p} - \mathbf{a}$, $\mathbf{ab} = \mathbf{b} - \mathbf{a}$, $\mathbf{ac} = \mathbf{c} - \mathbf{a}$, $\mathbf{ad} = \mathbf{d} - \mathbf{a}$. For example, if the barycentric coordinates are (1, 0, 0, 0), the point \mathbf{p} is on the position of vertex \mathbf{a} . In addition, if one of the coordinates is zero, the point \mathbf{p} is on the element face corresponding to the coordinate, such as the barycentric coordinates (0.5, 0, 0.2, 0.3) representing the

point \mathbf{p} on the face \widetilde{acd} . For point outside the element, some of the barycentric coordinates can be negative or greater than one. For an arbitrary point \mathbf{p} in the tetrahedron element $abcd$, the barycentric coordinates satisfy the relationship

$$0 \leq \lambda_i(\mathbf{p}) \leq 1, \quad i = 1, 2, 3, 4. \quad (44)$$

The transformation from the barycentric coordinates to the global coordinates is based on the properties, three mentioned above by the following matrix transformation:

$$\mathbf{p} = \mathbf{A}\boldsymbol{\lambda}(\mathbf{p}) \rightarrow \begin{bmatrix} x_p \\ y_p \\ z_p \end{bmatrix} = \begin{bmatrix} x_a & x_b & x_c & x_d \\ y_a & y_b & y_c & y_d \\ z_a & z_b & z_c & z_d \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \\ \lambda_4 \end{bmatrix}. \quad (45)$$

The matrix \mathbf{A} is called the transform matrix, which is composed of the global position of the element vertices. Similar to the transform matrix, a reverse transform matrix \mathbf{T} needs to be defined for the transformation from the global coordinates to the barycentric coordinates. The reverse transform matrix \mathbf{T} is expressed as

$$\mathbf{T} = [\mathbf{bd} \times \mathbf{bc} \quad \mathbf{ac} \times \mathbf{ad} \quad \mathbf{ad} \times \mathbf{ab} \quad \mathbf{ab} \times \mathbf{ac}]. \quad (46)$$

The barycentric displacement from point \mathbf{a} to point \mathbf{p} is calculated by

$$\lambda_{\mathbf{a} \rightarrow \mathbf{p}} = \lambda(\mathbf{p}) - \lambda(\mathbf{a}) = \frac{\mathbf{ap} \cdot \mathbf{T}}{\det \mathbf{A}} = \frac{[\mathbf{ap} \cdot (\mathbf{bd} \times \mathbf{bc}) \quad \mathbf{ap} \cdot (\mathbf{ac} \times \mathbf{ad}) \quad \mathbf{ap} \cdot (\mathbf{ad} \times \mathbf{ab}) \quad \mathbf{ap} \cdot (\mathbf{ab} \times \mathbf{ac})]}{\mathbf{ab} \cdot (\mathbf{ac} \times \mathbf{ad})}, \quad (47)$$

where $\det \mathbf{A}$ is the determinant of the matrix \mathbf{A} and is written as $\mathbf{ab} \cdot (\mathbf{ac} \times \mathbf{ad})$. Then, the barycentric coordinates of the point \mathbf{p} are expressed as

$$\begin{aligned} \lambda(\mathbf{p}) &= \lambda(\mathbf{a}) + \frac{\mathbf{ap} \cdot \mathbf{T}}{\det \mathbf{B}} = (1, 0, 0, 0) + \frac{[\mathbf{ap} \cdot (\mathbf{bd} \times \mathbf{bc}) \quad \mathbf{ap} \cdot (\mathbf{ac} \times \mathbf{ad}) \quad \mathbf{ap} \cdot (\mathbf{ad} \times \mathbf{ab}) \quad \mathbf{ap} \cdot (\mathbf{ab} \times \mathbf{ac})]}{\mathbf{ab} \cdot (\mathbf{ac} \times \mathbf{ad})} \\ &= \left(1 + \frac{\mathbf{ap} \cdot (\mathbf{bd} \times \mathbf{bc})}{\mathbf{ab} \cdot (\mathbf{ac} \times \mathbf{ad})}, \frac{\mathbf{ap} \cdot (\mathbf{ac} \times \mathbf{ad})}{\mathbf{ab} \cdot (\mathbf{ac} \times \mathbf{ad})}, \frac{\mathbf{ap} \cdot (\mathbf{ad} \times \mathbf{ab})}{\mathbf{ab} \cdot (\mathbf{ac} \times \mathbf{ad})}, \frac{\mathbf{ap} \cdot (\mathbf{ab} \times \mathbf{ac})}{\mathbf{ab} \cdot (\mathbf{ac} \times \mathbf{ad})} \right). \end{aligned} \quad (48)$$

The reverse transform matrix \mathbf{T} can be written as

$$\mathbf{T} = \begin{bmatrix} (y_d - y_b)(z_c - z_b) - (z_d - z_b)(y_c - y_b) & (y_c - y_a)(z_d - z_a) - (z_c - z_a)(y_d - y_a) \\ (z_d - z_b)(x_c - x_b) - (x_d - x_b)(z_c - z_b) & (z_c - z_a)(x_d - x_a) - (x_c - x_a)(z_d - z_a) \\ (x_d - x_b)(y_c - y_b) - (y_d - y_b)(x_c - x_b) & (x_c - x_a)(y_d - y_a) - (y_c - y_a)(x_d - x_a) \\ (y_d - y_a)(z_b - z_a) - (z_d - z_a)(y_b - y_a) & (y_b - y_a)(z_c - z_a) - (z_b - z_a)(y_c - y_a) \\ (z_d - z_a)(x_b - x_a) - (x_d - x_a)(z_b - z_a) & (z_b - z_a)(x_c - x_a) - (x_b - x_a)(z_c - z_a) \\ (x_d - x_a)(y_b - y_a) - (y_d - y_a)(x_b - x_a) & (x_b - x_a)(y_c - y_a) - (y_b - y_a)(x_c - x_a) \end{bmatrix}. \quad (49)$$

The determinant of the matrix \mathbf{A} is a constant for a given element and is expressed as

$$\begin{aligned} \det \mathbf{A} &= \mathbf{ab} \cdot (\mathbf{ac} \times \mathbf{ad}) = [x_b - x_a \quad y_b - y_a \quad z_b - z_a] \begin{bmatrix} (y_c - y_a)(z_d - z_a) - (z_c - z_a)(y_d - y_a) \\ (z_c - z_a)(x_d - x_a) - (x_c - x_a)(z_d - z_a) \\ (x_c - x_a)(y_d - y_a) - (y_c - y_a)(x_d - x_a) \end{bmatrix} \\ &= (x_b - x_a)(y_c - y_a)(z_d - z_a) - (x_b - x_a)(z_c - z_a)(y_d - y_a) + (y_b - y_a)(z_c - z_a)(x_d - x_a) \\ &\quad - (y_b - y_a)(x_c - x_a)(z_d - z_a) + (z_b - z_a)(x_c - x_a)(y_d - y_a) - (z_b - z_a)(y_c - y_a)(x_d - x_a). \end{aligned} \quad (50)$$

For a point \mathbf{p} moving through point \mathbf{m} to point \mathbf{n} with intersection point \mathbf{s} between the vector \mathbf{pn} and the face \widetilde{bcd} as shown in Fig. 4(a), where the point \mathbf{p} and \mathbf{m} are inside the tetrahedron \widetilde{abcd} and the point \mathbf{n} is outside the tetrahedron \widetilde{abcd} , the barycentric displacement related to the tetrahedron \widetilde{abcd} from point \mathbf{p} to point \mathbf{n} is calculated by

$$\lambda_{\mathbf{p} \rightarrow \mathbf{n}} = \lambda(\mathbf{n}) - \lambda(\mathbf{p}) = \frac{\mathbf{an} \cdot \mathbf{T}}{\det \mathbf{A}} - \frac{\mathbf{ap} \cdot \mathbf{T}}{\det \mathbf{A}} = \frac{\mathbf{pn} \cdot \mathbf{T}}{\det \mathbf{A}} \\ = \frac{[\mathbf{pn} \cdot (\mathbf{bd} \times \mathbf{bc}) \quad \mathbf{pn} \cdot (\mathbf{ac} \times \mathbf{ad}) \quad \mathbf{pn} \cdot (\mathbf{ad} \times \mathbf{ab}) \quad \mathbf{pn} \cdot (\mathbf{ab} \times \mathbf{ac})]}{\mathbf{ab} \cdot (\mathbf{ac} \times \mathbf{ad})}. \quad (51)$$

The face \widetilde{abc} , \widetilde{adc} , and \widetilde{abd} are parallel to the face $\widetilde{pb_1c_1}$, $\widetilde{pd_1c_1}$, and $\widetilde{pb_1d_1}$ in Fig. 4(a). Similarly, the barycentric displacement from the point \mathbf{p} to point \mathbf{m} is expressed as $\lambda_{\mathbf{p} \rightarrow \mathbf{m}} = \mathbf{pm} \cdot \mathbf{T} / \det \mathbf{A}$. The sign of the four elements of barycentric displacement $\lambda_{\mathbf{p} \rightarrow \mathbf{n}}$ can be different with the intersection point \mathbf{s} on the different area of the face \widetilde{bcd} as shown in Fig. 4(b).

Figure 4(b) is the top view of the face \widetilde{bcd} , which is divided into seven different regions. For example, if the intersection point \mathbf{s} is in the region $b_1c_1d_1$, the first element of $\lambda_{\mathbf{p} \rightarrow \mathbf{n}}$ is negative and the vector \mathbf{pn} only intersect with the face \widetilde{bcd} . What's more, if the intersection point is in the region $b_1b_3c_2c_1$, the vector \mathbf{pn} intersects with the face \widetilde{bcd} in point \mathbf{s} first and then intersects with the face \widetilde{abc} in point \mathbf{t} [Fig. 4(c)] with negative value of the first and fourth element of $\lambda_{\mathbf{p} \rightarrow \mathbf{n}}$. As a result, the vector \mathbf{pn} will intersect with the relating element face when the barycentric displacement is negative. This method is also applicable to the vector \mathbf{pm} with whose extension will intersect the face \widetilde{bcd} . The barycentric coordinates of the intersection point \mathbf{s} relating to the tetrahedral element \widetilde{abcd} can be calculated by adding the $\lambda(\mathbf{p})$ and the barycentric displacement $\lambda_{\mathbf{p} \rightarrow \mathbf{s}}$

$$\lambda(\mathbf{s}) = \lambda(\mathbf{p}) + \lambda_{\mathbf{p} \rightarrow \mathbf{s}} = \lambda(\mathbf{p}) + \mathbf{ps} \cdot \mathbf{T} / \det \mathbf{A} = \lambda(\mathbf{p}) + \mu_1 \lambda_{\mathbf{p} \rightarrow \mathbf{n}}, \quad (52)$$

where μ_1 is the ratio of the magnitude of the vector \mathbf{ps} and \mathbf{pn}

$$\mu_1 = \frac{|\mathbf{ps}|}{|\mathbf{pn}|} = \frac{|\mathbf{pe}|}{|\mathbf{pf}|} = \frac{\mathbf{bp} \cdot (\mathbf{bd} \times \mathbf{bc})}{-\mathbf{pn} \cdot (\mathbf{bd} \times \mathbf{bc})} = -\frac{\lambda_1(\mathbf{p})}{\lambda_{1\mathbf{p} \rightarrow \mathbf{n}}}, \quad (53)$$

where $\lambda_1(\mathbf{p})$ and $\lambda_{1\mathbf{p} \rightarrow \mathbf{n}}$ are the first element of the barycentric coordinates $\lambda(\mathbf{p})$ and barycentric displacement $\lambda_{\mathbf{p} \rightarrow \mathbf{n}}$.

However, if the vector \mathbf{pn} intersects the face \widetilde{abc} in point \mathbf{s} first and then intersects the face \widetilde{bcd} in point \mathbf{t} as shown in Fig. 4(d), the sign of the barycentric displacement $\lambda_{\mathbf{p} \rightarrow \mathbf{n}}$ is $(- + + -)$, which is the same sign as Fig. 4(c). The barycentric coordinates of the point \mathbf{p} and \mathbf{t} in Fig. 4(d) are expressed as

$$\lambda(\mathbf{s}) = \lambda(\mathbf{p}) + \mu_4 \lambda_{\mathbf{p} \rightarrow \mathbf{n}}, \quad (54)$$

$$\lambda(\mathbf{t}) = \lambda(\mathbf{p}) + \mu_1 \lambda_{\mathbf{p} \rightarrow \mathbf{n}}, \quad (55)$$

where $\mu_4 = \lambda_4(\mathbf{p}) / \lambda_{4\mathbf{p} \rightarrow \mathbf{n}}$. Equation (44) can be used to determine whether the point \mathbf{s} and \mathbf{t} are in the tetrahedron \widetilde{abcd} . The point \mathbf{s} is in the tetrahedron \widetilde{abcd} and satisfies the relationship of Eq. (44). It can be concluded that the vector \mathbf{pn} intersects the face \widetilde{abc} first. This method can be used to determine which face the vector \mathbf{pn} intersects with first considering an arbitrary point \mathbf{n} outside the tetrahedron \widetilde{abcd} .

The position, velocity, cell containing information, tetrahedron containing information, and step fraction of a particle are stored

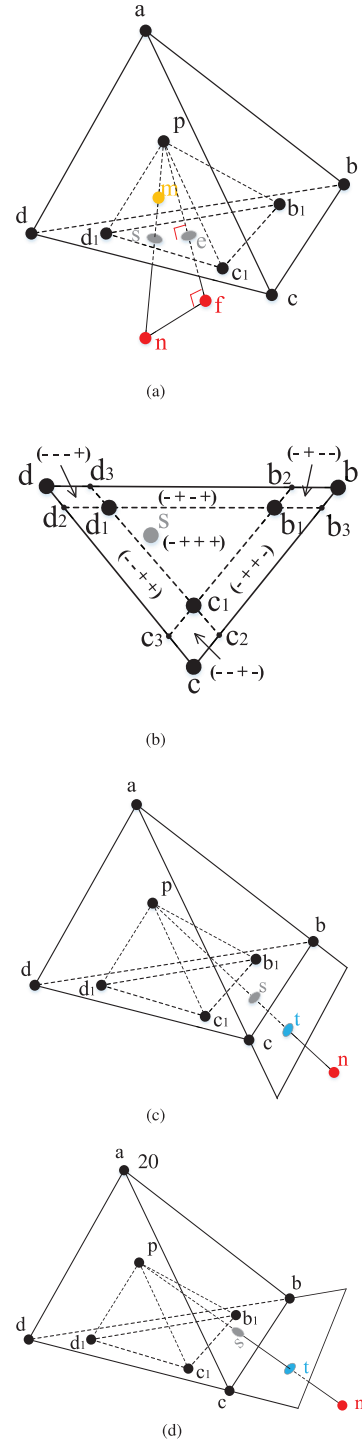


FIG. 4. Diagram for judging the intersection of particle trajectory and cell face. (a) Diagram of the intersection of particle trajectory \mathbf{pn} and tetrahedral element face \widetilde{bcd} . The vector \mathbf{pf} is perpendicular to the face \widetilde{bcd} . (b) Sign of the four elements of barycentric displacement of different intersection point \mathbf{s} on face \widetilde{bcd} . (c) Particle trajectory \mathbf{pn} intersects with the face \widetilde{bcd} (first) and \widetilde{abc} (second). (d) Particle trajectory \mathbf{pn} intersects with the face \widetilde{abc} (first) and \widetilde{bcd} (second).

during the particle tracking. The step fraction is the ratio of the current displacement to the total displacement of the particle in a fluid time step. For example, considering the particle moving from point **a** to point **b** in a fluid time step, the step fraction of the particle in point **r** is $|\mathbf{ar}|/|\mathbf{ab}|$ (Fig. 5). The particle tracking is complete in the fluid time step when step fraction = 1. The particle tracking is divided into many steps according to the intersection of the particle trajectory and the tetrahedron face in a fluid time step. Figure 5 shows a particle moving through a hexahedral cell, which is divided into twelve tetrahedra with the same vertex **c** (cell center). The point **p**, **q**, **r**, **s**, and **t** are the intersection point. Then, the particle tracking in this cell is divided into four steps **p** → **q** → **r** → **s** → **t**. The cell and tetrahedron containing information are the label of the cell and tetrahedron where the particle is localized.

A briefly description is given bellow to describe the particle tracking process in a cell shown in Fig. 5. It is assumed that the step fraction when particle moves through point **p** is SF_1 . The cell and tetrahedron containing information need to be changed after the particle passing point **p** because the particle hit a cell face $\widetilde{278}$ and enter from one cell to another cell. Then, the particle is tracked from point **p** to point **b** with the intersection point **q** on face $\widetilde{c87}$. The barycentric coordinates of the point **p** and barycentric displacement of the vector **pb** relating to the tetrahedron $\widetilde{c728}$ are calculated based on Eqs. (47) and (51). Based on the method of determining the intersection point and face between a vector and a tetrahedron given above [Eqs. (51)–(55)], the intersection face $\widetilde{c78}$ can be determined and the barycentric coordinates of the intersection point **q** relating to the tetrahedron $\widetilde{c728}$ can be calculated by Eq. (52). The global coordinates of the point **q** are achieved from Eq. (45). After the particle moving to the point **q**, the step fraction needs to be updated by

$$\text{stepfraction} = SF_1 + (1 - SF_1)\mu_i, \quad (56)$$

$$\mu_i = -\frac{\lambda_i(\mathbf{p})}{\lambda_{ip \rightarrow b}}, \quad (57)$$

where μ_i is $|\mathbf{pq}|/|\mathbf{pb}|$, which is similar to Eq. (53), and i is the i th (intersection) face of the tetrahedron $\widetilde{c728}$ in Fig. 5. The tetrahedron containing information is changed from the tetrahedron $\widetilde{c728}$ to $\widetilde{c678}$, while the cell containing information does not need to be changed due to the same cell where the particle is localized. Then, the particle is tracking from **q** to **r**, **r** to **s**, and **s** to **t** with the same process of **p** to **q**. It should be noted that both the cell and tetrahedron containing information need to be changed after the particle passing the point **t**

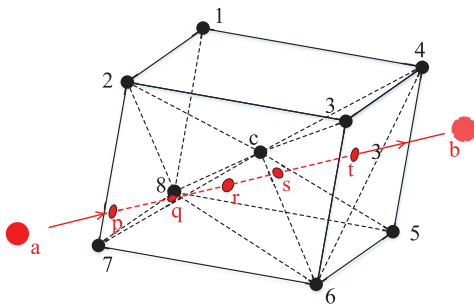


FIG. 5. Particle moving through a cell.

because the particle hit the cell face $\widetilde{3654}$. The particle collisions are processed at the end of each time step. During calculating the velocity and position of the particle in the next time step, the velocity changing caused by the particle collision is separated from the velocity changing resulting from other forces such as the gravity force, the pressure gradient force, and so on because the collision of particles is instantaneous. The leapfrog integration algorithm (LIA)⁷⁰ is used to calculate the velocity of the particle due to the particle collisions. The equations for updating position and velocity of the particle are

$$\mathbf{U}_{\text{dcol}}^{t+\frac{\Delta t}{2}} = \mathbf{U}_{\text{dcol}}^t + \frac{\mathbf{F}_{\text{col}}^t \Delta t}{m_d}, \quad (58)$$

$$\mathbf{x}_{\text{dcol}}^{t+\Delta t} = \mathbf{x}_{\text{dcol}}^t + \mathbf{U}_{\text{dcol}}^{t+\frac{\Delta t}{2}} \Delta t, \quad (59)$$

$$\mathbf{U}_{\text{dcol}}^{t+\Delta t} = \mathbf{U}_{\text{dcol}}^{t+\frac{\Delta t}{2}} + \frac{\mathbf{F}_{\text{col}}^{t+\frac{\Delta t}{2}} \Delta t}{m_d}, \quad (60)$$

where \mathbf{U}_{dcol} and \mathbf{x}_{dcol} are the particle velocity and displacement caused by the particle collisions, respectively. The process of LIA can be summarized as: (a) applying half $\Delta \mathbf{U}_{\text{dcol}}$ with the particle collision force stored in the previous time step, (b) moving the particle with the new particle velocity and getting the new particle position based on the PTM, (c) calculating the particle collision force in the new particle position, and (d) applying half $\Delta \mathbf{U}_{\text{dcol}}$ with the new particle collision force, where $\Delta \mathbf{U}_{\text{dcol}}$ is the velocity changing caused by the particle collision.

The particle velocity resulting from other forces is calculated based on the implicit Euler scheme. Removing the particle collision force from Eq. (25)

$$m_d \frac{d\mathbf{U}_{\text{dcol}}}{dt} = m_d \mathbf{g} \left(1 - \frac{\rho_c}{\rho_d} \right) + \frac{K_d (\mathbf{U}_c - \mathbf{U}_{\text{dcol}})}{\alpha_c} + \rho_c \frac{m_d}{\rho_d} \frac{D\mathbf{U}_c}{Dt}, \quad (61)$$

where \mathbf{U}_{dcol} is the particle velocity resulting from other forces. Applying the implicit Euler discretization to Eq. (61)

$$\frac{\mathbf{U}_{\text{dcol}}^{t+\Delta t} - \mathbf{U}_{\text{dcol}}^t}{\Delta t} = \left(1 - \frac{\rho_c}{\rho_d} \right) \mathbf{g} + \frac{K_d (\mathbf{U}_c^t - \mathbf{U}_{\text{dcol}}^{t+\Delta t})}{m_d \alpha_c} + \rho_c \frac{1}{\rho_d} \frac{D\mathbf{U}_c^t}{Dt}. \quad (62)$$

The superscript t and $t + \Delta t$ represent the value of the variable at this time step. Then, the velocity of the particle at the next time step ($t + \Delta t$) relating to other forces is expressed by

$$\mathbf{U}_{\text{dcol}}^{t+\Delta t} = \frac{\mathbf{U}_{\text{dcol}}^t + \left[\left(1 - \frac{\rho_c}{\rho_d} \right) \mathbf{g} + \frac{K_d}{m_d \alpha_c} \mathbf{U}_c^t + \rho_c \frac{1}{\rho_d} \frac{D\mathbf{U}_c^t}{Dt} \right] \Delta t}{1 + \frac{K_d}{m_d \alpha_c} \Delta t}. \quad (63)$$

The velocity of the particle causing by the particle collisions and other forces is calculated based on Eqs. (60) and (63), respectively. As a result, the particle velocity and position at the next time step are calculated by

$$\mathbf{U}_d^{t+\Delta t} = \mathbf{U}_{\text{dcol}}^{t+\Delta t} + \mathbf{U}_{\text{dcol}}^{t+\Delta t}, \quad (64)$$

$$\mathbf{x}_d^{t+\Delta t} = \mathbf{x}_d^t + \left(\mathbf{U}_{\text{dcol}}^t + \mathbf{U}_{\text{dcol}}^{t+\frac{\Delta t}{2}} \right) \Delta t. \quad (65)$$

C. Aggregation and interpolation

The Eulerian framework, specific locations in a space through which the fluid flows are focused on as time passes. An individual

point is followed as it moves through space and time in the Lagrangian framework. The flow field of the continuous phase is solved in the Eulerian framework, while the motion of the discrete phase is solved in the Lagrangian framework in this work. Hence, the bridging between the continuous phase and the discrete phase needs to be illustrated, such as the momentum exchange term of the drag force in Eq. (33) and the continuous phase velocity in Eq. (25). Here, aggregation means the mapping from discrete phase to continuous phase and interpolation means the mapping from continuous phase to discrete phase. Sometimes, the aggregation and interpolation are also referred to as “forward” and “backward” interpolations, respectively.

1. Aggregation

The PCM is used to calculate the continuous quantities of the particles such as the particle volume fraction (α_d) and the momentum exchange term of the drag force (\mathbf{f}_d) in OpenFOAM. The PCM only corresponds to the cell where the particles' centroid locate, which may be suitable for fine particles where the particle size is smaller than the cell size ($d < 0.2 - 0.4\Delta x$, where $\Delta x = \sqrt[3]{V_{\text{cell}}}$ is the cell size).³⁴ To overcome the drawback of the PCM for large particles ($d/\Delta x \geq 1$), a VMDF is given in this paper for calculating the volume fraction of large particles. In general, the true mass distribution function (TMDF) for a particle is

$$M_T(r) = \int_{|\gamma|=0}^r m_T(\gamma) d\gamma, \quad (66)$$

$$m_T(\gamma) = \begin{cases} |\gamma| \leq R, & \rho_d, \\ |\gamma| > R, & 0, \end{cases} \quad (67)$$

where $\gamma = \mathbf{x} - \mathbf{p}$, \mathbf{x} is the position vector of an arbitrary point in the flow field, \mathbf{p} is the position vector of the particle center, R is the particle radius, and $m_T(\gamma)$ is the true density distribution function (TDDF). In the numerical simulation, if the cell size is larger than the particle diameter ($d/\Delta x < 1$), the particle volume fraction is computed as

$$\alpha_d = \frac{1}{V_{\text{cell}}} \sum_{i=1}^n \int_{|\gamma| \in V_{\text{cell}}} \frac{1}{\rho_d} m_{T,i}(\gamma) d\gamma. \quad (68)$$

Notice that the cell volume is larger than the particle volume and a packing limit or maximum particle volume fraction is set as 0.9 before for preventing the particle volume fraction from too large in a cell. Therefore, it may result in a rigorous relation of $0 \leq \alpha_d < 1$. However, if the cell volume is smaller than the particle volume, all the particle volume fraction in the containing cell is 0.9 due to the packing limit, which will cause calculation divergence or inaccurate calculation results. A coarse mesh can be used to achieve practical particle volume fraction field. Yet, this method may also lead to inadequate simulation results of the continuous phase. Therefore, the VMDF is proposed to overcome the drawback. As shown in Fig. 6, a large particle occupies several cells, and the VMDF spreads the influence of particle from the containing cell into the surrounding cell. An appropriate VMDF should follow the criteria as follow:³⁹

1. The integral of the virtual density distribution function (VDDF) needs to be converging and finite, that is,

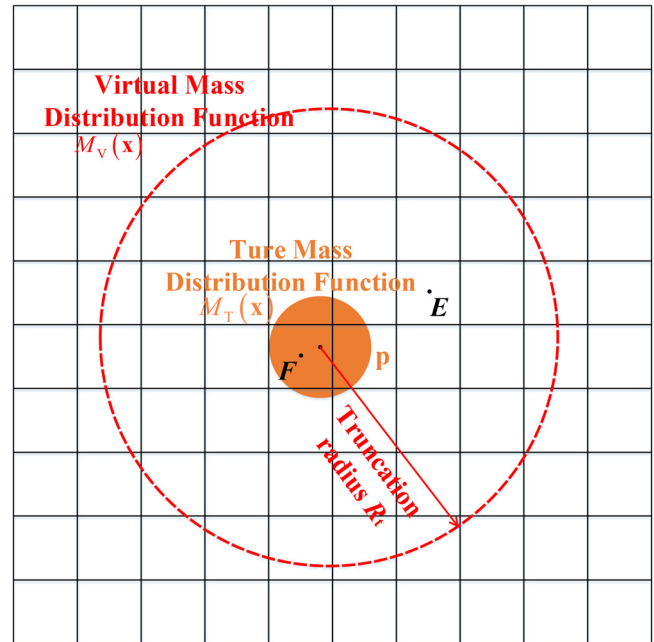


FIG. 6. The virtual mass distribution function.

$$\lim_{r \rightarrow +\infty} M_V(r) = \int_{|\gamma|=0}^r m_V(\gamma) d\gamma, \quad (69)$$

where $M_V(r)$ is the VMDF, and $m_V(\mathbf{r})$ is the VDDF.

2. The particle mass should be conserved when using the VMDF; that is, the mass computed from the VMDF should be the same as the true particle. The conservation requirement can be expressed as

$$M_V(\infty) = \int_{|\gamma|=0}^{\infty} m_V(\gamma) d\gamma = \int_{|\gamma|=0}^{\infty} m_T(\gamma) d\gamma = M_T(\infty). \quad (70)$$

3. There should be a truncation radius R_t that satisfies

$$M_V(\infty) = \int_{|\gamma|=0}^{\infty} m_V(\gamma) d\gamma \approx \int_{|\gamma|=0}^{R_t} m_V(\gamma) d\gamma = M_V(R_t). \quad (71)$$

4. The VMDF should be a smooth function to prevent numerical instability caused by discontinuities on the particle surface.

To guarantee the conservation requirement of the VDDF, we notice the linear partial differential equations

$$\begin{cases} \frac{\partial \varphi(\boldsymbol{\beta}, \tau)}{\partial \tau} = \nabla^2 \varphi(\boldsymbol{\beta}, \tau), \\ \varphi(\boldsymbol{\beta}, \tau)|_{\tau=0} = \delta(\boldsymbol{\beta}) \end{cases} \quad (72)$$

has the analytical solution of

$$\varphi(\boldsymbol{\beta}, \tau) = \frac{1}{(4\pi\tau)^{\frac{3}{2}}} \exp\left(-\frac{\boldsymbol{\beta}^2}{4\tau}\right), \quad (73)$$

where τ is a time parameter, and $\delta(\boldsymbol{\beta})$ is the Dirac function. Figure 7 shows the change of $\varphi(\boldsymbol{\beta}, \tau)$ with $|\boldsymbol{\beta}|$ at different time parameter τ . It

can be seen that $\lim_{|\beta| \rightarrow +\infty} \varphi(\beta) = 0$, and $\int_{|\beta|=0}^{\infty} \varphi(\beta) d\beta = 1$. Hence, the VDDF can be defined as

$$m_V(\gamma) = \frac{4}{3} \pi R^3 \rho_d \varphi(\gamma, \tau). \quad (74)$$

Then, $M_V(\infty) = \frac{4}{3} \pi R^3 \rho_d = M_T(\infty)$. Furthermore, the VDDF given in Eq. (74) has a truncation radius of about $R_t = 2.5R$ and the VMDF is a smooth function. The VMDF defined by Eq. (74) meets all the criteria given above.

It is assumed that there is only a large particle \mathbf{p} locating at the cell F in the flow field, as shown in Fig. 6. Applying the VMDF to calculating the particle volume fraction, the particle volume fraction field at cell E relating to the particle \mathbf{p} locating at cell F with any mesh scale can be written as

$$\begin{aligned} \alpha_d(\mathbf{x}_E) &= \frac{1}{V_{\text{cell}}(F)} \int_{|\gamma| \in V_{\text{cell}}(E)} \frac{1}{\rho_d} m_V(\gamma) d\gamma \\ &= \int_{|\gamma| \in V_{\text{cell}}(E)} \frac{4}{3} \frac{\pi R^3}{V_{\text{cell}}(F)} \frac{1}{(4\pi\tau)^{\frac{3}{2}}} \exp\left(-\frac{\gamma^2}{4\tau}\right) d\gamma \\ &= \int_{|\gamma| \in V_{\text{cell}}(E)} \frac{\alpha_{d,\text{PCM}}(\mathbf{x}_F)}{(4\pi\tau)^{\frac{3}{2}}} \exp\left(-\frac{\gamma^2}{4\tau}\right) d\gamma, \end{aligned} \quad (75)$$

where \mathbf{x}_E and \mathbf{x}_F are the cell center of the cell E and F , respectively; $V_{\text{cell}}(E)$ and $V_{\text{cell}}(F)$ are the volume of the cell E and F , respectively; and $\alpha_{d,\text{PCM}}(\mathbf{x}_F)$ is particle volume fraction at cell F calculated based on the PCM and can be regarded as a square integrable function because the particle volume fraction in a cell is the same. The cell E can be an arbitrary cell in the flow field. Based on the Green's function, Eq. (75) is the solution of the following equation:

$$\begin{cases} \frac{\partial \varphi(\beta, \tau)}{\partial \tau} = \nabla^2 \varphi(\beta, \tau), \\ \varphi(\beta, \tau)|_{\tau=0} = \alpha_{d,\text{PCM}}(\mathbf{x}_F) \delta(\mathbf{x}_F). \end{cases} \quad (76)$$

To conclude, in order to get the particle volume fraction field relating to the particle \mathbf{p} (Fig. 6), Eq. (76) can be solved with the boundary condition of zero gradient to meet the conservation requirement.⁷¹ If there are many large particles in the flow field, the particle volume fraction

field can be added together. The concept of VMDF is for computation of the particle volume fraction field only.

Similar procedure can be applied to the momentum exchange term of the drag force (\mathbf{f}_d) in Eq. (27). The momentum exchange term of the drag force \mathbf{f}_d is expressed by Eq. (30), which is a source term in the continuous phase momentum equation. The source term must be distributed into surrounding cells because a large particle occupies several cells. To ensure conservation, the distribution function $\varphi(\gamma, \tau)$ is still used to distribute the momentum exchange term of the drag force as follows:

$$\mathbf{f}_d(\xi) = \mathbf{f}_{d,\text{PCM}}(\mathbf{x}) \varphi(\mathbf{x} - \xi, \tau), \quad (77)$$

where $\mathbf{f}_{d,\text{PCM}}$ is the momentum exchange term of the drag force calculated based on the PCM. Equation (77) can also be achieved by solving the following equation:

$$\begin{cases} \frac{\partial \mathbf{f}_{d,\text{PCM}}}{\partial \tau} = \nabla^2 \mathbf{f}_{d,\text{PCM}}, \\ \mathbf{f}_{d,\text{PCM}}|_{\tau=0} = \mathbf{f}_{d,\text{PCM}}(\mathbf{x}) \delta(\mathbf{x}), \end{cases} \quad (78)$$

where \mathbf{x} is center of the containing cell of the particle. The VMDF given above can be used to deal with coarse particles.

2. Interpolation

The mapping from continuous phase to discrete phase, which is called interpolation, can also be different for fine particles and coarse particles. The field relating to the continuous phase such as the velocity field, the volume fraction field, and the pressure field is stored in the center of the cell or cell face. However, if the interaction force between the particle and the fluid is calculated from Eqs. (21)–(24), the continuous phase velocity \mathbf{U}_c at the particle centroid is unknown and need to be achieved from the stored velocity in the cell first because the particle centroid is neither in the center of the cell F nor in the center of the cell face. The barycentric interpolation method (BIM) can be used for fine particles. For example, a small particle \mathbf{p} locates in a tetrahedral cell $abcd$ as shown in Fig. 3(a). The velocity of the continuous phase (\mathbf{U}_c) at the cell vertex is known, and then, \mathbf{U}_c at the particle centroid can be calculated based on the barycentric coordinates of \mathbf{p} relating to the tetrahedron $abcd$

$$\begin{aligned} \mathbf{U}_c(\mathbf{x}_p) &= \lambda_1(\mathbf{p}) \mathbf{U}_c(\mathbf{x}_a) + \lambda_2(\mathbf{p}) \mathbf{U}_c(\mathbf{x}_b) + \lambda_3(\mathbf{p}) \mathbf{U}_c(\mathbf{x}_c) \\ &\quad + \lambda_4(\mathbf{p}) \mathbf{U}_c(\mathbf{x}_d), \end{aligned} \quad (79)$$

where \mathbf{x}_p , \mathbf{x}_a , \mathbf{x}_b , \mathbf{x}_c , and \mathbf{x}_d are position vector. When considering large particles, the barycentric interpolation is not applicable. Another interpolation method called weighted function method (WFM) is proposed for large particles in this paper. The quantities of the continuous phase are collected among the surrounding cells according to the distance between the surrounding cell centers and the particle centroid, which is used to construct the weighted function. Figure 8 shows a large particle in the flow field with surrounding nine cells. The weight functions w_i allocating into CFD cells are calculated by

$$w_i = \frac{1/|\mathbf{x}_p - \mathbf{x}_i|}{\sum_{i=1}^{N_s} 1/|\mathbf{x}_p - \mathbf{x}_i|}, \quad (80)$$

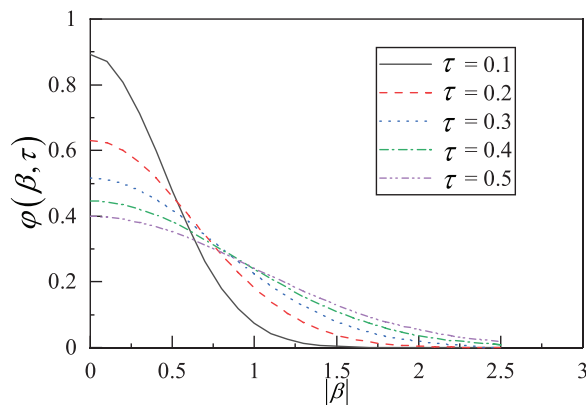


FIG. 7. The change of $\varphi(\beta, \tau)$ with $|\beta|$ at different time parameter τ .

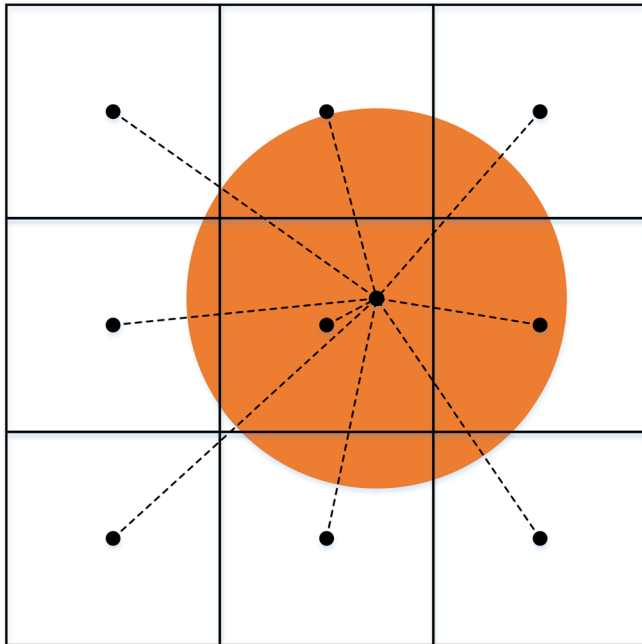


FIG. 8. The diagram of the weighted function method (WFM).

where \mathbf{x}_i is the position vector of the center of the surrounding cells, and N_s is the number of the surrounding cells. Then, the continuous phase velocity at the particle center is written as

$$\mathbf{U}_c(\mathbf{x}_p) = \sum_{i=1}^{N_s} w_i \mathbf{U}_c(\mathbf{x}_i). \quad (81)$$

D. Numerical platform

The numerical models given in Secs. II A–II C, which are released as a new solver named CoarseDPMFoam for modeling gas–solid or fluid–solid two-phase flow, are conducted by means of OpenFOAM. OpenFOAM is a C++ tool box for the development of customized numerical solvers, and pre-processing and post-processing utilities for the solution. The proposed CoarseDPMFoam is built upon through the FVM for the continuous phase equations and the centroid method for the discrete phase equations. The motion of the continuous phase is described by the macroscopic continuum approach, which is set by the continuity and momentum conservation equations. The discrete phase particles are tracked individually based on Newton’s second law of motion.⁴⁷ Thus, the distributions of the continuous phase, and the velocities and trajectories of the discrete phase are obtained in the cells. Figure 9 shows the flow chart of the CoarseDPMFoam. The Eulerian–Lagrangian CFD simulation requires processes both at particle scale such as the collision force, pressure gradient force, drag force, and relating velocity to be solved in the Lagrangian framework and at macro-scale including phase velocity and volume fraction in the Eulerian framework.²¹ In each time step, the discrete phase calculation is performed first, and then, the continuous phase calculation followed

because the volume fraction of the continuous phase depends on the position of the discrete phase particles.

III. NUMERICAL SIMULATIONS AND MODEL VALIDATION

For accessing the accuracy of the numerical models (CoarseDPMFoam), some test cases are presented. The first one is the particle distribution case, the second one is a gas–solid fluidized bed case, the third one is a liquid–solid particle size segregation case, and the last one is a granular column collapse case. The equations governing gas–solid and liquid–solid flows are the same in the incompressible region, and the difference lies in the physical properties of the continuous phase (gas or liquid). The previous equation derivation does not involve separate equations for gas–solid and liquid–solid flows. Therefore, the gas–solid and liquid–solid flow cases are given to prove that the solver is suitable for both gas–solid and liquid–solid flows.

A. Distribution of particle volume fraction

To verify the accuracy of the VMDF method for calculating the particle volume fraction field, two distributions of particles located in the middle and near boundaries of a two-dimensional domain with size of $160d \times 160d$ in length (x -direction) and height (y -direction), respectively, are used. Simulations are performed on meshes with only one layer of cell in z -direction. The thickness of the cell in z -direction is the same as the particle diameter. All particles locate on the same plane normal to z -direction. The test case is only used to verify the particle volume fraction distribution. As a result, both particles and fluid remain stationary, which means no CFD or DEM simulations are performed in this test case. 5000 spherical particles are random distributed in the computational domain (Fig. 10). In addition to the results obtained from the VMDF method, the particle volume distribution based on the two-grid method and the PCM (in OpenFOAM) is also given. The coarse mesh of the two-grid method obtains 4×4 cells in this paper. Figure 11 shows the comparison of the particle volume fraction along the red dashed line in Fig. 10 between the VMDF method, the two-grid method, and the PCM. It can be seen that the particle volume fraction obtained by using VMDF method and that by using the two-grid method agree well in both the middle and boundary part. Due to the lower resolution of the coarse mesh for two-grid method, the results obtained from the VMDF method are smoother. However, as is evident from the comparison, extreme high or low values of the particle volume fraction are most frequent in the PCM results with largest value of about 0.6, which is close to the maximum possible value for close-packed spherical particles in two-dimensional domain. It is attributed to the model defects of the PCM. In contrast, no such phenomenon is present in the results obtained with the two-grid method and that from the VMDF method. The VMDF proposed in this paper gives accurate calculation results with clear physical meaning. The important thing is that this method is easy to implement and overcome the shortcomings of the PCM in OpenFOAM. It has a wider range of applications than the two-grid method because the two-grid method is difficult to be implemented in unstructured grids.

B. Gas–solid fluidized bed

A gas–solid fluidized bed, where both motion of the continuous phase gas and the discrete phase particles are resolved in a pseudo-2D

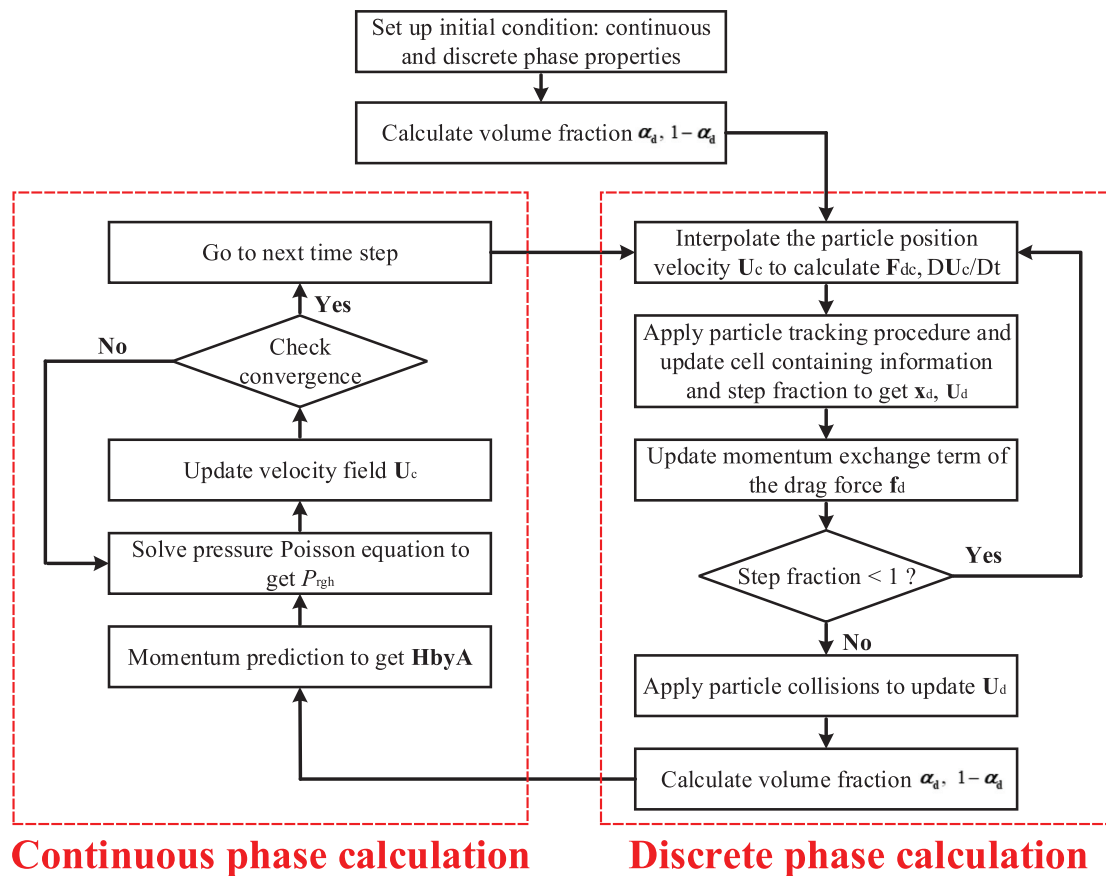


FIG. 9. The flow chart of the CoarseDPMFoam.

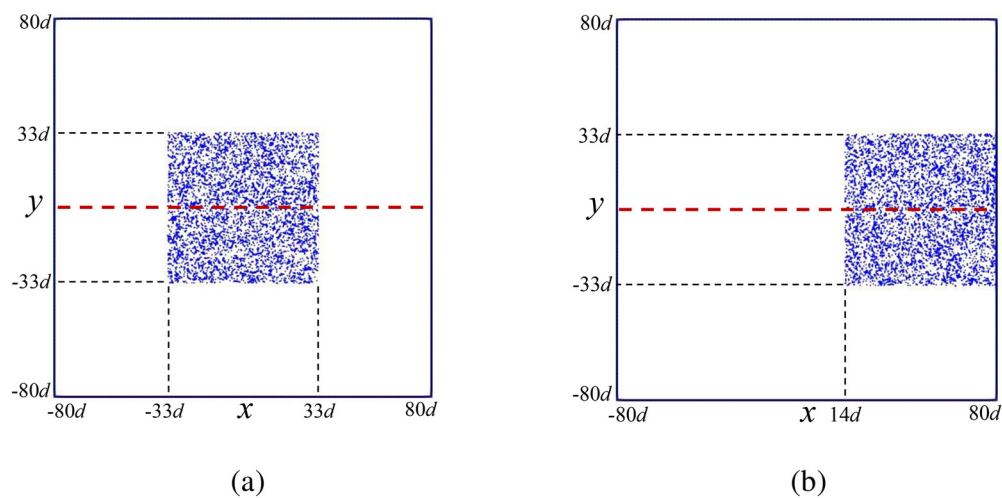


FIG. 10. Diagram of the particle distribution in (a) the middle of the domain and (b) the boundary of the domain.

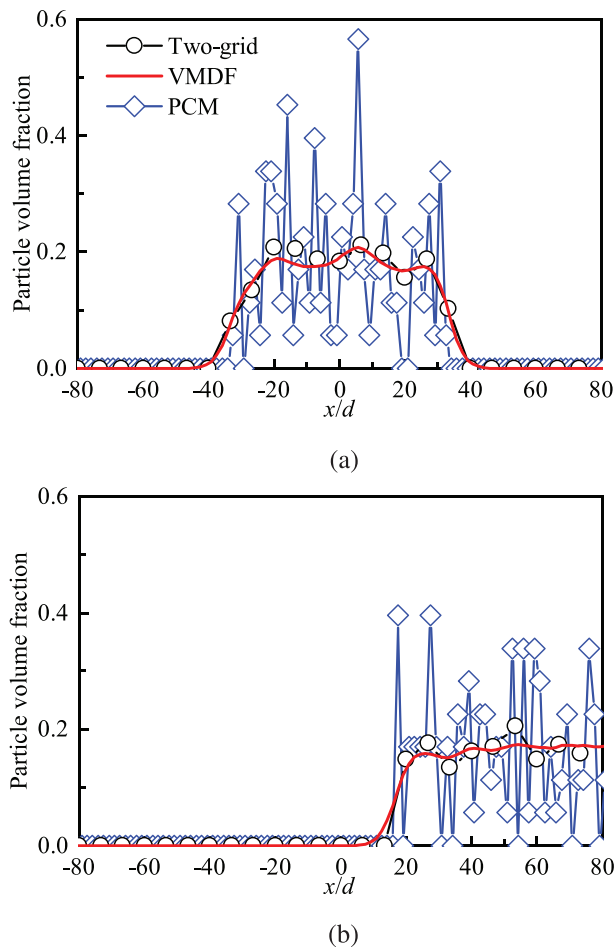


FIG. 11. Comparison of the particle volume fraction between the VMDF method, the two-grid method, and the PCM, (a) the particle volume fraction along the red dashed line showed in Figs. 10(a), and (b) the particle volume fraction along the red dashed line showed in Fig. 10(b).

geometry, is simulated. The numerical test case references the fluidized bed experiments of Muller.^{72,73} The geometrical and flow conditions are reproduced in the numerical test case. In the experiments, the scales of the fluidized bed are width \times length \times height = $10 \times 44 \times 120 \text{ mm}^3$ (aligned with the x , y , and z axes, respectively, in the numerical test case coordinate system). Poppy seeds with diameter 1.2 mm and density 1000 kg/m^3 are used as fluidized bed particles, which have a kindly-like shape. Initially, the fluidized bed height is 30 mm, consisting of 9240 poppy seed particles, and these seeds are fluidized with the air. Figure 12 gives the geometry along with the coordinate system used in the numerical simulations of the gas-solid fluidized bed. Magnetic resonance is used for the gas volume fraction measurement⁷³ and the particles velocity detection.⁷² The mesh resolution in the numerical simulation is $N_x \times N_y \times N_z = 8 \times 36 \times 100$, where N_x , N_y , and N_z are the number of cells in the width, length, and height of the fluidized bed, respectively. The y -direction faces are set to be walls with no-slip boundary conditions, x -direction faces are set to symmetry, the top face (upper z -direction) is set to be zero value for

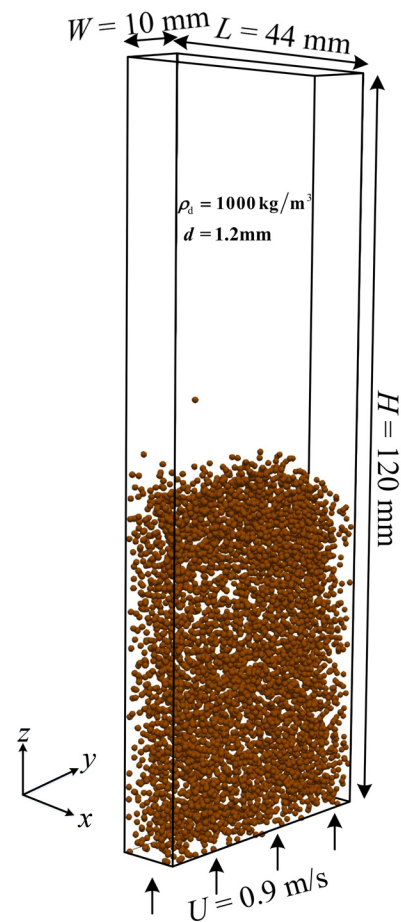


FIG. 12. Geometry representation of the simulated fluidized bed.

the relative motion pressure (P_{rgh}) divided by the gas density, and the bottom face (lower z -direction) is set to a fixed velocity for the gas phase. For the bottom face, the velocity, which is called interstitial inlet velocity, is calculated by dividing the specified gas velocity with the local solid volume fraction at the bottom face. Due to the changing of the particle volume fraction resulting from the particle movement near the bottom face, the actual average velocity may be a little different to keep the constant flow rate along time.

The parameters used in the numerical simulation of the fluidized bed are listed in Table II. A fixed time step of $1 \times 10^{-5} \text{ s}$ is used in the simulation. The experiment data measured by magnetic resonance can only obtain time-averaged measurements. As a result, the instantaneous particle velocity and gas volume fraction have to be time-averaged to compare the results of the simulation and the experiment. The simulations are averaged for 15 s, which is long enough to obtain statistically time-averaged results. Sections III B 1–III B 4 show the comparison between the present simulation results with the experiment results and other numerical simulations such as the CFD-DEM,^{72,73} TFM-kinetic theory of granular flow (KTGF), and DPMFoam results. The TFM-KTGF model is an Eulerian–Eulerian

TABLE II. Parameters used in the numerical simulation of the fluidized bed.

Parameters	Values
Geometry of the fluidized bed	
Width, length, height	10, 44, 120 mm
Particle properties	
Density ρ_d	1000 kg/m ³
Diameter d	1.2 mm
Number	9240
Young's modulus E	1.2×10^5 Pa
Poisson's ratio ν	0.33
Coefficient of restitution for particle-particle/wall α/α_w ^a	0.02/0.01
Coefficient of friction for particle-particle/wall μ/μ_w ^b	0.10/0.09
Gas properties	
Density ρ_c	1.2 kg/m ³
Kinetic viscosity ν_c	1.5×10^{-5} m ² /s
Inlet velocity U_z	0.9 m/s
Cells in x, y, and z direction	8, 36, 100
N_x , N_y , and N_z	

^aParameters in soft model (Sec. II A 1).

^bParameters in soft model (Sec. II A 1).

method combining the TFM with the kinetic theory of granular flow (KTGF), which is used to model the particle collisions. The details about the TFM-KTGF can be found in Refs. 74 and 75. This model has been implanted in OpenFOAM with a solver named twophaseEulerFoam.⁷⁶ The DPMFoam⁴⁵ is a Eulerian-Lagrangian method solver in OpenFOAM similar to the proposed solver CoarseDPMFoam. The DPMFoam solver uses the PCM method to calculate the particle volume fraction and uses the cell interpolation method for the mapping from continuous phase to discrete phase. The grid size for the DPMFoam solver is set as $N_x \times N_y \times N_z = 3 \times 12 \times 33$, with the ratio of the cell size to the particle size is $\Delta x/d = 3$.

1. Comparison of the time-averaged fields

The gas volume fraction α_c is measured at two different heights, $z = 16.4$ and 31.2 mm (the middle and the top of the fluidized bed) in the fluidized bed experiment.⁷³ Hence, the gas volume fraction at the two heights is also extracted in the numerical simulation. The comparison with the experimental results, the CFD-DEM simulations,⁷³ the TFM-KTGF simulation results, and the DPMFoam results is given in Fig. 13. The gas volume fraction calculated from the present CoarseDPMFoam solver agrees well with the experimental results. It should be noted that the gas volume fraction obtained from the numerical simulation (CoarseDPMFoam and CFD-DEM) is greater than the experimental results near the wall. This phenomenon is especially obvious when $z = 31.2$ mm. Martin, Loth, and Lankford⁴⁴ attributed this phenomenon to the over-prediction of the bubbles width in the CFD-DEM solvers, which will lead to large gas volume fraction near the wall. In addition, the gas volume fraction profiles obtained from the TFM-KTGF model are very different from the experimental

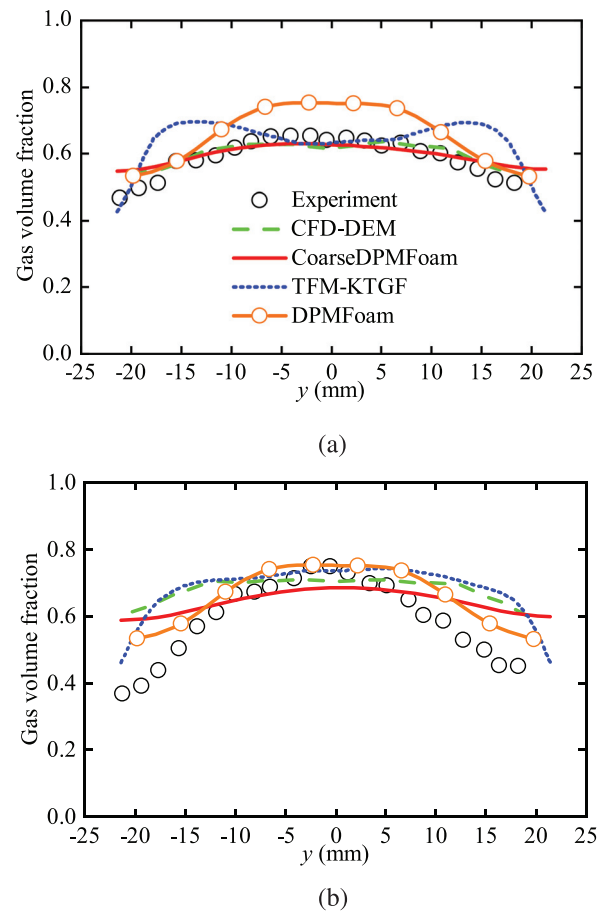


FIG. 13. Gas volume fraction comparison between the CoarseDPMFoam with the experiment results, the CFD-DEM results,⁷³ the TFM-KTGF simulation results, and the DPMFoam results. The results are extracted at two different heights of the fluidized bed, (a) $z = 16.4$ mm, and (b) $z = 31.2$ mm. [Reproduced with permission from Müller, "Validation of a discrete element model using magnetic resonance measurements," *Particuology* 7, 297–306 (2009). Copyright 2009 Chinese Society of Particuology and Institute of Process Engineering, Chinese Academy of Sciences.]

value. A possible explanation is that the KTGF not applicable when the particles are larger than the cells. The error of solid viscosity and the solid pressure may be relatively large. The DPMFoam results are larger than the experimental results in the whole region, which may result from the inaccurate flow field simulation based on the coarse mesh.

The time-averaged vertical velocities of the poppy seed particles obtained from the experiments and different numerical simulations (CoarseDPMFoam, CFD-DEM, TFM-KTGF, and DPMFoam) at two different heights, $z = 15$ and 25 mm, are shown in Fig. 14. The numerical simulation results from the CoarseDPMFoam, the CFD-DEM, and the DPMFoam are all give a good prediction of the particle time-averaged vertical velocities. However, the results of the optimized CoarseDPMFoam method are slightly better than the CFD-DEM and DPMFoam solver in OpenFOAM. In addition, it can be found that the time-averaged particle vertical velocity calculated by the CFD-DEM

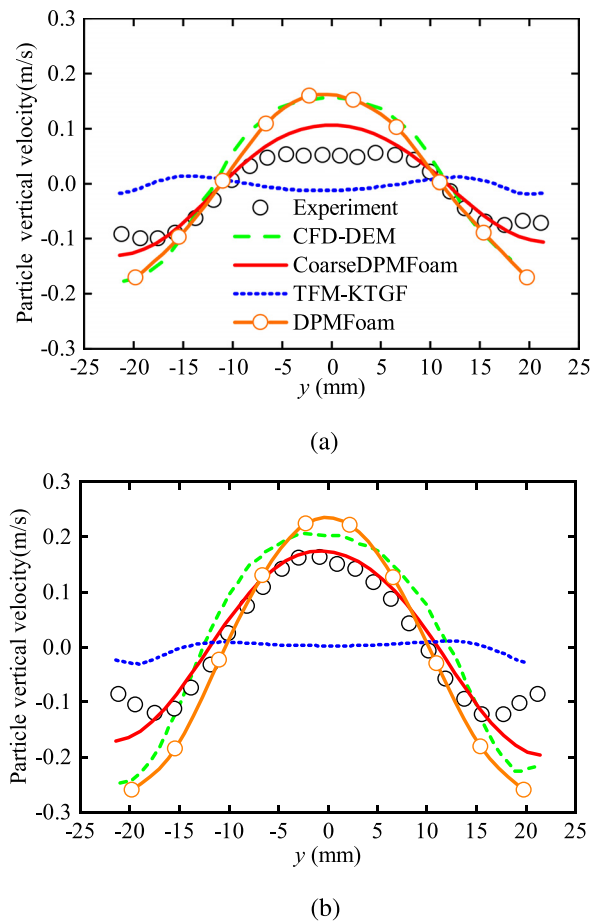


FIG. 14. Particle vertical velocity comparison between the CoarseDPMFoam with the experiment results, the CFD-DEM results,⁷² the TFM-KTGF simulation results, and the DPMFoam results. The results are extracted at two different heights of the fluidized bed, (a) $z = 15$ mm, and (b) $z = 25$ mm. [Reproduced with permission from Müller *et al.*, "Granular temperature: Comparison of magnetic resonance measurements with discrete element model simulations," *Powder Technol.* **184**, 241–253 (2008). Copyright 2007 Elsevier B.V.]

method and the DPMFoam solver is greater than the experiment value over the entire section ($z = 15$ and 25 mm). This is because the particle volume fraction field calculated based on the conventional CFD-DEM method is not smooth, and the particle volume fraction in a certain cell may be large. The drag force between the particle and the gas increases dramatically with the increase in the particle volume fraction, which lead to large particle vertical velocity, although it can be seen from Fig. 13 that the time-averaged gas volume fraction is basically the same. The drag force between the particle and gas is calculated based on the instantaneous volume fraction. The instantaneous particle volume fraction in a certain cell can be large when the CFD-DEM method and DPMFoam solver are used. Consequently, it results in large drag force between the particle and gas, and thus large time-averaged particle vertical velocity. The time-averaged particle vertical velocity obtained from the TFM-KTGF model tends to zero in the whole region. The first reason may be the inapplicability of KTGF for

coarse particles. The second reason may be the boundary conditions for the zero value of the particle velocity at the side walls, which may be different from the experiment. However, the results based on the CoarseDPMFoam seem to give prediction closer to the experiment except for a minor over-prediction at the center of the fluidized bed. In general, the numerical simulation results given by the CoarseDPMFoam solver are in good agreement with the experiment results and the previous CFD-DEM simulation of the same case.

2. Comparison of the Ergun pressure drop

When the gas passes through the poppy seed particles, the pressure will loss in the fluidized bed due to the frictional resistance. As the gas velocity increases, the pressure drop increases accordingly. When the upward drag force of the gas on the poppy seed particles is equal to the buoyant weight of the particles in the fluidized bed, the bed reaches the fluidization state because the particles are lifted by the upflow gas and the separation of the particles increase. As the inlet velocity of the gas increases, bed expansion continues accordingly. For spherical particles, the fluidization is found to begin when the gas volume fraction exceeds a value of about 0.46, which is nearly that for the void fraction in cubical arrangement of spherical particles (loosest stable form).⁷⁷ This phenomenon means that the fluidization does not start before the bed reaches the equivalent of the loosest stable form. Ergun and Oring⁷⁷ investigated the fluid flow through packed and fluidized bed and proposed a general equation, which related the pressure drop to the gas flow for fixed beds

$$\frac{\Delta P}{L} = 150 \frac{(1 - \alpha_c)^2 \mu_c U_z}{\alpha_c^3 d^2} + 1.75 \frac{1 - \alpha_c \rho_c U_z^2}{\alpha_c^3 d}, \quad (82)$$

where L is the bed height, and μ_c is the gasdynamic viscosity. According to Eq. (82), the ratio of the pressure drop to the gas inlet velocity is a linear function of the gas mass flow rate. The coefficients of the linear function depend on the gas volume fraction, the particle diameter, and the gasdynamic viscosity. The fixed bed will expand after pressure drop equals the buoyant weight of the particles per unit area of the bed, which is called the minimum fluidization. The general equation for the fixed bed can also be applied to the expanding bed or fluidized bed. The pressure drop stays constant at the fluidization state. In addition, the inlet velocity at the minimum fluidization ($U_{z,mf}$) can be predicted by the correlation

$$Ar = a * Re_{mf} + b * Re_{mf}^2, \quad (83)$$

where Ar is the Archimedes number and is expressed by $Ar = d^3 \rho_c (\rho_d - \rho_c) g / \mu_c^2$, $Re_{mf} = \rho_c U_{z,mf} d / \mu_c$ is the Reynolds number, and a and b are constants. Many researchers have studied the constants a and b , which is summarized by Subramani, Mothivel Balaiyya, and Miranda.⁷⁸ The pressure drop obtained from the CoarseDPMFoam solver is compared with that calculated based on the analytical formula given by Ergun and Oring⁷⁷ (as shown in Fig. 15). It can be seen that the numerical simulation results correctly predicts the pressure drop of the fluidized bed. The pressure drop obtained from the CoarseDPMFoam solver is slightly different from the analytical curve. The numerical simulation predictions of the pressure drop are over-estimated at the turning point of the analytical curve. However, with the increase in the inlet velocity of the gas, the

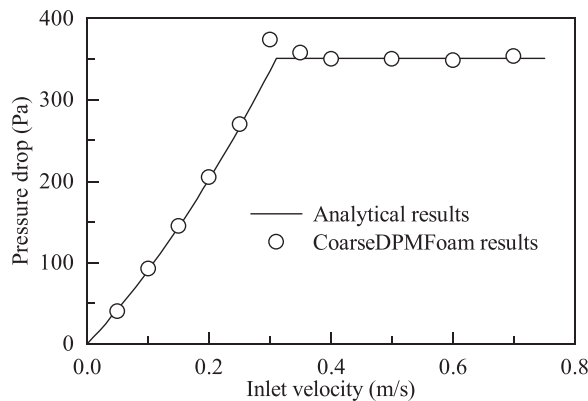


FIG. 15. Comparison of the pressure drop between the CoarseDPMFoam results and the analytical results obtained from Ergun equation [Eq. (82)].

numerical pressure drop approaches the constant value of the analytical pressure drop.

3. Mesh independence

As discussed above, the PCM may be suitable for fine particles where the particle size is smaller than the cell size or to be more precise $d < 0.2 - 0.4\Delta x$. Here, the numerical simulation results based on the CoarseDPMFoam solver at different cell size need to be studied. As a result, the gas volume fraction and the particle vertical velocity at the same location are compared by using five different cell size $\Delta x/d = 5, 3, 2, 1$, and 0.5 , arranged with increasing the mesh resolution. The details of the cell size are given in Table III.

The gas volume fraction at two cross sections where $z = 31.2$ mm (horizontal direction) and $y = 10$ mm (vertical direction) is exacted for comparison. Although the experimental data at the cross sections $y = 10$ mm are unknown, it does not impair the investigation of the effect of the mesh size on the simulation results because the comparison of the simulation results obtained from different mesh size is mainly focused on in this section. The particle vertical velocity is also obtained at two cross sections $z = 25$ mm and $y = 10$ mm. Figure 16 shows the time-averaged gas volume fraction and the particle vertical velocity at two cross sections obtained from the five different meshes (M1-M5). It can be seen that the same simulation results are achieved in the prediction of the gas volume fraction and particle vertical velocity at all the five meshes. However, some minor discrepancies are

TABLE III. The details of the mesh sizes for investigating the effect of the mesh size on the numerical simulation results.

Mesh	N_x	N_y	N_z	$\Delta x/d$
M1	2	7	20	5
M2	3	12	33	3
M3	4	18	50	2
M4	8	36	100	1
M5	16	72	200	0.5

found in the simulation results of mesh M1 ($\Delta x/d = 5$). For example, the gas volume fraction in cross section $z = 31.2$ mm obtained from mesh M1 is slightly larger than other simulation results, and the particle vertical velocity at $z = 25$ mm near the middle domain of the fluidized bed [e.g., between $y = -10$ mm and $y = 10$ mm in Fig. 16(c)] is smaller than other results. The minor discrepancies may result from inadequate mesh resolution. The mesh independence investigation the CoarseDPMFoam solver shows that this solver can be used for different mesh size and $\Delta x/d$ is better less than 5.

4. Interpolation method

The WFM relating to the particle size is given for the interpolation between the Eulerian and Lagrangian field for coarse particles. The influence of different interpolation method on the simulation results is investigated. Three interpolation methods are used: barycentric interpolation, WFM, and cell interpolation, which makes the cell center value equal to the particle centroid value. Figure 17 gives the comparison of the particle vertical velocity between the WFM, barycentric interpolation method, the cell interpolation method, and the experiment results. The ratio of the cell size to the particle size is $\Delta x/d = 0.5$. Other parameters are same to those in Table II. The barycentric interpolation and cell interpolation method are only applied for fine particles. The comparison shows that the WFM proposed gives more accurate prediction results than the other two interpolation methods.

C. Particle size segregation

Granular materials composed of particles with different densities, sizes, or shapes will experience particle segregation during flow.^{79,80} For example, as shown in Fig. 18, a mixture of particles with different sizes moves to the right and particle segregation occurs due to different drag forces exerted on the particles. Two test cases are taken to investigate the particle size segregation in this chapter.

1. One-way coupling

First, a one-way coupling test case is given. The movement of particles with different sizes along the horizontal direction as reported in Fig. 18 is simulated. Initially, a mixture of particles with three different sizes ($d = 0.2, 1.2$, and 2 mm) are at a certain location and move to the right horizontal direction with an equal velocity of 0.5 m/s. The continuous phase remains stationary, whose velocity is set as 0 m/s. The continuous phase is not affected by the discrete phase due to the one-way coupling, which means that the velocity of the continuous phase is 0 m/s all the time. The density of the continuous phase and the particles are 100 and 1000 kg/m³, respectively, and the drag coefficient C_D is set as 0.424 for simplicity. The width of the particle mixture is 0.15 m. Only the drag force is considered in this test case. The particles will stop at a certain time due to the drag force. It is worth noting that the particle with different size decelerates differently because the drag force is dependent on the particle size.

Because only the drag force is considered in the test case, the analytical solution of Eq. (25) can be obtained. Equation (25) is simplified to

$$m_d \frac{dU_d}{dt} = -K_d U_d. \quad (84)$$

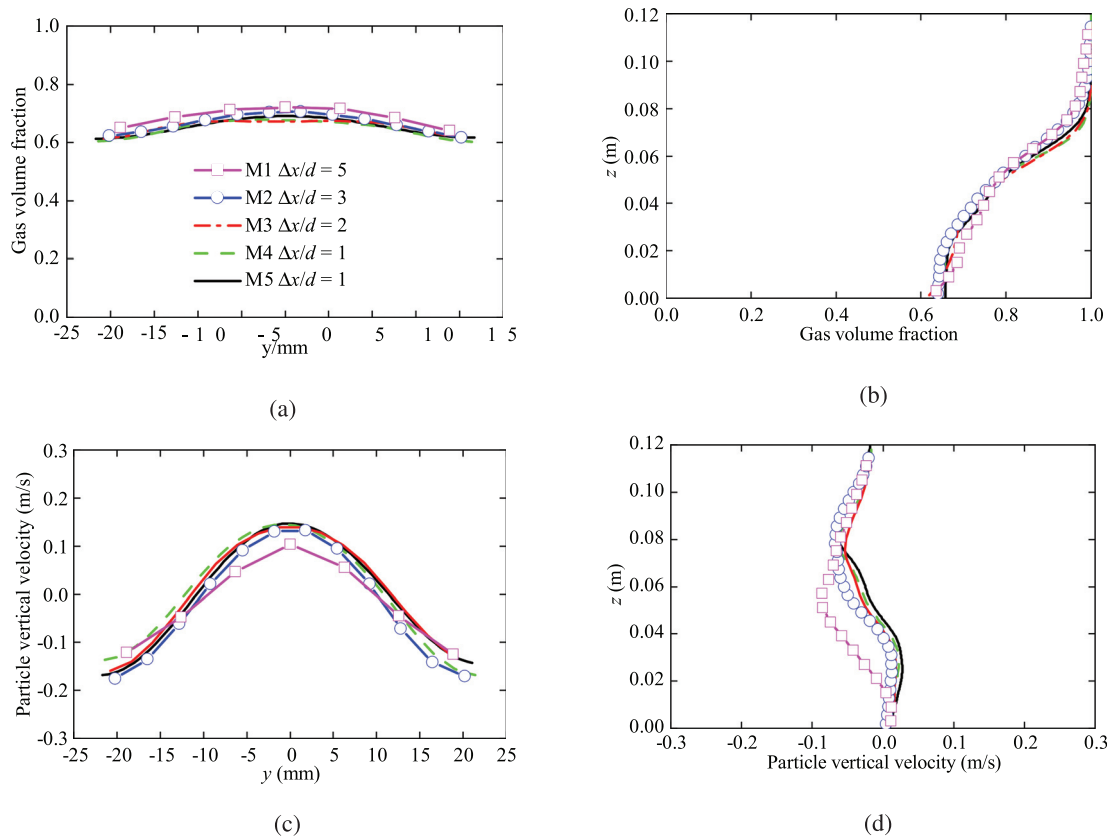


FIG. 16. Mesh independence investigation of the CoarseDPMFoam solver, showing the gas volume fraction α_c at two cross sections located at (a) $z = 31.2$ mm, and (b) $y = 10$ mm, respectively, and giving the particle vertical velocity at two cross sections located at (c) $z = 25$ mm, and (d) $y = 10$ mm, respectively. The simulation results obtained from five different meshes are compared.

The analytical solution of Eq. (84) is

$$U_d = \frac{U_{d0}}{aU_{d0}t + 1}, \quad (85)$$

where U_{d0} is the initial velocity of the particles and $a = \frac{3}{4} \frac{C_D}{d} \frac{\rho_c}{\rho_d}$ is a constant. Then, the displacement of the particles is calculated by $dx/dt = U_d$ and is written as

$$x(t) = \frac{\ln(aU_{d0}t + 1)}{a}. \quad (86)$$

From the known parameters, the displacement of the particles with different sizes can be obtained. The comparison of the numerical simulation results of the particle displacement ($d = 0.2, 1.2$, and 2 mm) with the analytical solution is reported in Fig. 19. It can be seen that both the numerical simulation results agree well with the analytical solution, which confirms the suitability of the CoarseDPMFoam solver in simulating the particle size segregation and verifies its implementation. The particle with diameter $d = 2$ mm moves farther than the other particles due to large inertial force. As time increases, the particle mixtures gradually segregate each other.

2. Four-way coupling

The above test case is a one-way coupling simulation of the particle size segregation. Then, a four-way coupling test case is presented here to show the capabilities of the CoarseDPMFoam solver. There are

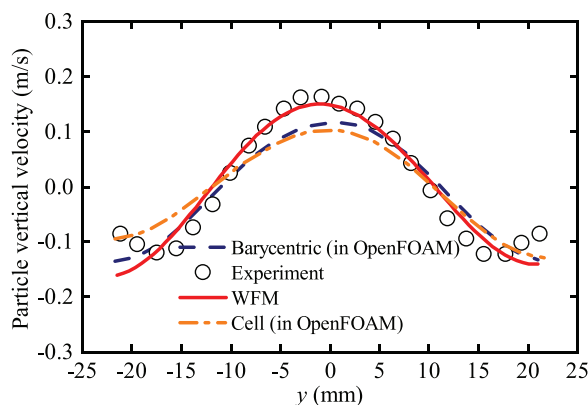


FIG. 17. Comparison of the particle vertical velocity between the WFM, barycentric interpolation method, the cell interpolation method, and the experiment results.

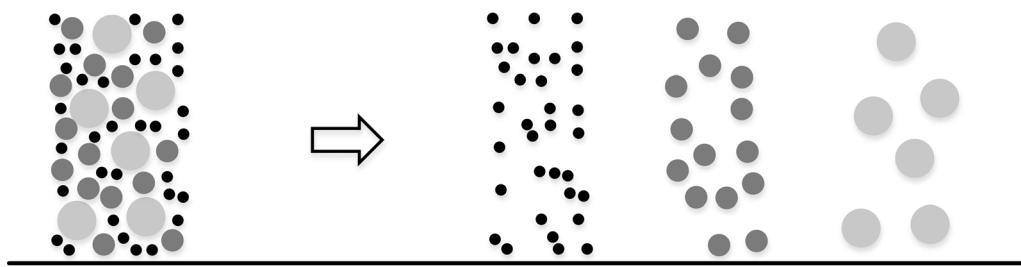


FIG. 18. Particle size segregation resulting from different drag forces exerted on particle with different sizes.

a few experimental data for solid–liquid particle size segregation. An experimental apparatus is designed in this paper (as shown in Fig. 20), which mainly includes four parts: vertical pipe, circulating water system, control system, and image acquisition device. The vertical pipe has an inner diameter of 50 mm and a height of 3 m. An electromagnetic flowmeter is installed in the middle of the pipe to measure the flow rate. 700 particles of diameter 6 (600) and 13 mm (100) are placed at the bottom of the pipe initially. A steel filter is installed at the bottom to prevent particles from falling. Then, the pump is turned on to form a delivery water flow. The particles are glass beads with density 2600 kg/m^3 and flat surface, which can be used to study the particle size segregation under ideal conditions. The water density and viscosity are 1000 kg/m^3 and 0.001 Pa s . The particle movement is recorded as the flow rate increasing.

As the flow rate increases slowly from $0 \text{ m}^3/\text{h}$, the mixed particles remain static. When the flow rate reaches $0.12 \text{ m}^3/\text{h}$, few particles start to move on the upper part of the particle bed. When the flow rate reaches $0.28 \text{ m}^3/\text{h}$, some of the particles of diameter 6 mm have moved to the upper part of the bed. Then, the particles of diameter 6 mm are completely separated from the particle bed with obvious segregation when the flow rate reaches $0.35 \text{ m}^3/\text{h}$. The particles of diameter 6 mm are in a fluidized state, while the particles of diameter 13 mm have not reached the fluidized state. When the flow rate reaches $0.7 \text{ m}^3/\text{h}$, the expansion height of the particle bed increases and the particles of diameter 13 mm reach the fluidized state. However, the coarse and fine particles still have an obvious interface.

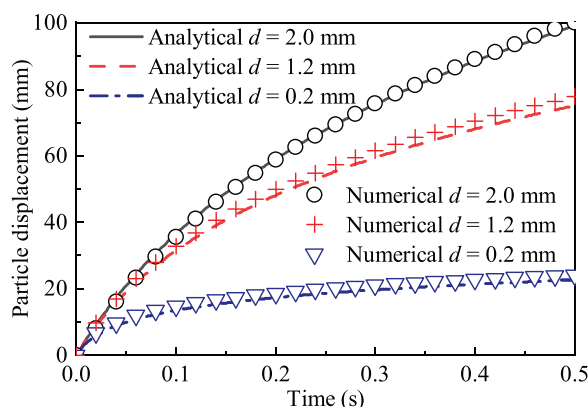


FIG. 19. Comparison of the numerical simulation results of the particle displacement ($d = 0.2$, 1.2 , and 2 mm) with the analytical solution.

The same test case is simulated based on CoarseDPMFoam. The comparison between the experimental results and the numerical simulation results at different flow rates is given in Fig. 21. The numerical simulation results are basically consistent with the experimental results.

It is found that the overall motion of the particles has obvious pulsation characteristics in both numerical simulations and experiments. The motion of the particles is similar to a spring, which shows a state of up and down [Fig. 21(c)]. The velocity of the water in the pipe will increase due to the particle blocking. The particles move upward under the water lifting effect. As the particles move upward, the gaps between the particles become larger. Then, the flow area of the water gradually increases, and the fluid velocity decreases. When the downward forces of the particles are larger than the lifting force, the particles will fall. Similarly, as the particles fall, the gap becomes smaller and the water velocity increases. When the upward force of the particles is greater than the downward force, the particles begin to move upward. In addition, the pulsation amplitude of the particles of diameter 6 mm is much smaller than that of 13 mm. This may be because of larger gap between the coarse particles. The gap change of coarse particles is larger than that of the fine particles, so the velocity of the water also changes greatly, resulting in stronger pulsation.

D. Granular column collapse process

The granular column collapse is a typical process of the debris flow or landslide. Lee, Huang, and Yu,⁸¹ Lee, and Huang⁸² studied the

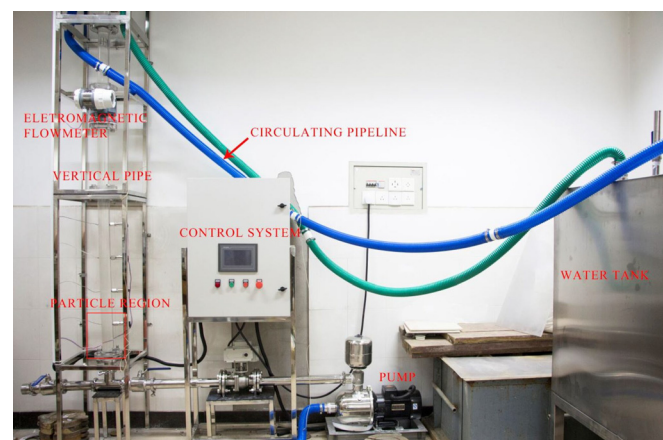


FIG. 20. The vertical pipe apparatus.

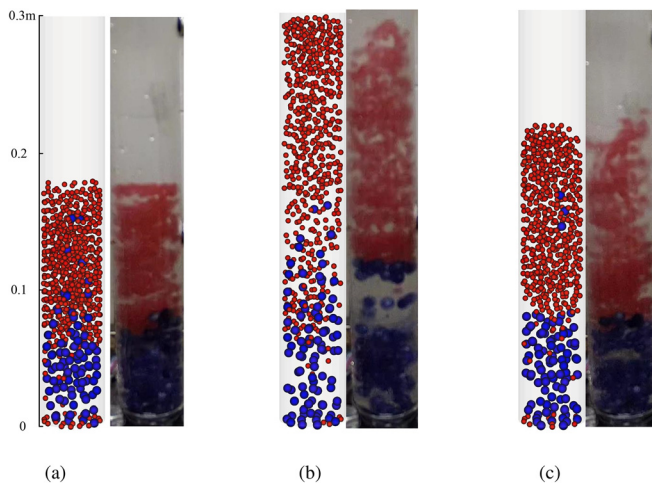


FIG. 21. Comparison of the solid-liquid particle segregation between CoarseDPMFoam results and the experiment data at different water volume flow rate, (a) $0.35 \text{ m}^3/\text{h}$, (b) $0.70 \text{ m}^3/\text{h}$, and (c) $1.05 \text{ m}^3/\text{h}$.

collapse of saturated granular columns immersed in water. Zhang *et al.*⁸³ paid attention to granular column collapse in air. The experimental results extracted from Zhang *et al.*⁸³ are used in this section. Figure 22 is a schematic diagram of the granular column collapse in this study. The height and length of the initial particles are $H_0 \times L_0 = 120 \times 40 \text{ mm}^2$. The granular column is on one side of the cuboid, and free collapse of the granular column is studied. The granular materials are glass beads with a density of 2650 kg/m^3 and a uniform size of 2.5 mm . The density of the air is 1.2 kg/m^3 , and the viscosity is $1.8 \times 10^{-5} \text{ Pa s}$. The initial volume fraction of particles is set to 0.55 with particles numbers of about 1000. A same case is simulated based on the CoarseDPMFoam solver and DPMFoam solver. The mesh size is $1.0 \times 1.0 \times 1.0 \text{ mm}^3$ for CoarseDPMFoam and $4 \times 4 \times 4 \text{ mm}^3$ for DPMFoam. The simulation results are shown in Fig. 23. The results obtained from CoarseDPMFoam are obviously better in line with the experimental results than the DPMFoam results.

IV. CONCLUSION

In this work, an optimized Eulerian-Lagrangian method for two-phase flow with coarse particles is proposed and implemented in the open-source CFD code OpenFOAM as CoarseDPMFoam solver. It mainly includes five parts: derivation of governing equations, equation

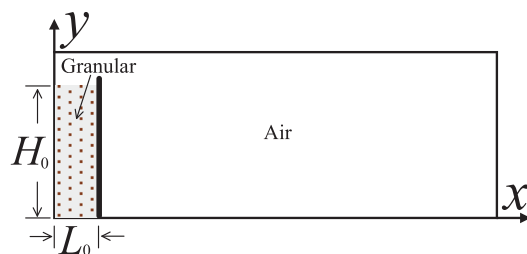
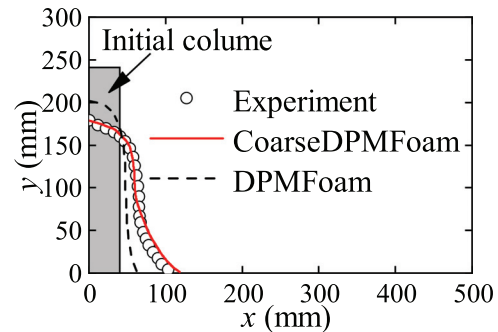
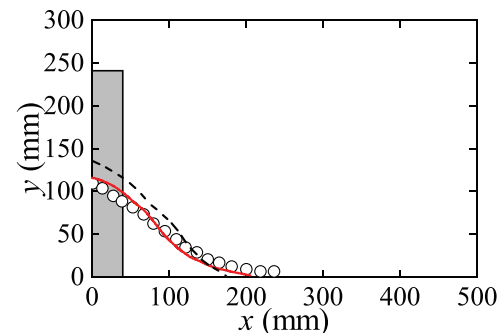


FIG. 22. Diagram of granular column in water.



(a)



(b)

FIG. 23. Simulation results of the granular column collapse process at different time, (a) $t = 0.15 \text{ s}$, and (b) $t = 0.45 \text{ s}$.

solving strategies, Eulerian-Lagrangian coupling, solver structure, and test cases.

A review of the numerical models describing multiphase flows is given in this paper to illustrate the capabilities of different models. Then, some modified models are proposed for conventional Eulerian-Lagrangian method. Considering the interphase interaction and the influence of the high hydrostatic pressure, the locally averaged equations of motion for the continuous phase, which is solved in an Eulerian framework, are obtained based on N-S equations of single-phase flow. Then, the PISO algorithm for solving the N-S equations of the continuous phase with phase fraction and momentum exchange source term is proposed. The discrete phase is tracked in a Lagrangian method by solving Newton's equations of motion, in which the implicit Euler scheme and the leapfrog integration algorithm are adopted to separate the velocity changing caused by the particle collisions from the velocity changing resulting from other forces. In dealing with Eulerian-Lagrangian coupling, the VMDF is proposed to calculate the volume fraction fields and the momentum exchange source term fields of the coarse particles based on the idea of SKM. The WFM relating to the particle size is given for mapping data from continuous phase to discrete phase. Our algorithm and implementation are validated against four test cases. One is the particle distribution case to verify the application of the VMDF method for coarse particles. The second is the gas-solid fluidized bed experiment, which verifies the time-averaged fields, pressure drop, and mesh independence of the fluidized bed, respectively. The third is the

verification of particle segregation based on the experimental results from a self-developed particle transportation apparatus. The last one is a granular column collapse case, which is based on experimental results in the literature. The proposed numerical simulation method provides new ideas and methods for the mechanism investigation and engineering application of the two-phase flow with coarse particles.

ACKNOWLEDGMENTS

This study is supported by the Strategic Priority Research Program of the Chinese Academy of Sciences (Grant No. XDA22000000), the Key Special Project for Introduced Talents Team of Southern Marine Science and Engineering Guangdong Laboratory (Guangzhou, No. GML2019ZD0307), and the Youth Innovation Promotion Association of Chinese Academy of Sciences (No. 2017027).

AUTHOR DECLARATIONS

Conflict of Interest

The authors have no conflicts to disclose.

DATA AVAILABILITY

The data that support the findings of this study are available from the corresponding author upon reasonable request.

REFERENCES

- T. Asim and R. Mishra, "Optimal design of hydraulic capsule pipelines transporting spherical capsules," *Can. J. Chem. Eng.* **94**, 966–979 (2016).
- J. Oh, J. Jung, H. Kim, S. Hong, K. Sung, and D. Bae, "Gap size effect on the tribological characteristics of the roller for deep-sea mining robot," *Mar. Georesour. Geotechnol.* **35**, 120–126 (2017).
- H. Zhang, T. Li, Z. Huang, S. Kuang, and A. Yu, "Investigation on vertical plug formation of coarse particles in a non-mechanical feeder by CFD-DEM coupling method," *Powder Technol.* **332**, 79–89 (2018).
- J.-W. Oh, J.-Y. Jung, and S. Hong, "On-board measurement methodology for the liquid-solid slurry production of deep-seabed mining," *Ocean Eng.* **149**, 170–182 (2018).
- M. Zhou, S. Wang, S. Kuang, K. Luo, J. Fan, and A. Yu, "CFD-DEM modelling of hydraulic conveying of solid particles in a vertical pipe," *Powder Technol.* **354**, 893–905 (2019).
- X. Ting, Z. Xinzhuo, S. A. Miedema, and C. Xiuhan, "Study of the characteristics of the flow regimes and dynamics of coarse particles in pipeline transportation," *Powder Technol.* **347**, 148–158 (2019).
- S. Kuang, M. Zhou, and A. Yu, "CFD-DEM modelling and simulation of pneumatic conveying: A review," *Powder Technol.* **365**, 186–207 (2020).
- J. Wang, D. Joseph, N. Patankar, M. Conway, and R. Barree, "Bi-power law correlations for sediment transport in pressure driven channel flows," *Int. J. Multiphase Flow* **29**, 475–494 (2003).
- S. Tong and K. K. Mohanty, "Proppant transport study in fractures with inter-sections," *Fuel* **181**, 463–477 (2016).
- M. W. McClure, H. Jung, D. D. Cramer, and M. M. Sharma, "The fracture-compliance method for picking closure pressure from diagnostic fracture-injection tests," *SPE J.* **21**, 1321–1339 (2016).
- J. Wang, D. Elsworth, and M. K. Denison, "Propagation, proppant transport and the evolution of transport properties of hydraulic fractures," *J. Fluid Mech.* **855**, 503–534 (2018).
- W. Shen, X. Li, T. Ma, J. Cai, X. Lu, and S. Zhou, "High-pressure methane adsorption behavior on deep shales: Experiments and modeling," *Phys. Fluids* **33**, 063103 (2021).
- R. Boswell and T. S. Collett, "Current perspectives on gas hydrate resources," *Energy Environ. Sci.* **4**, 1206–1215 (2011).
- X. Zhang, X. Lu, and L. Liu, "Advances in natural gas hydrate recovery methods," *Prog. Geophys.* **29**, 858–869 (2014).
- P. Li, X. Zhang, and X. Lu, "Three-dimensional Eulerian modeling of gas-liquid-solid flow with gas hydrate dissociation in a vertical pipe," *Chem. Eng. Sci.* **196**, 145–165 (2019).
- P. Li, X. Zhang, and X. Lu, "Dissociation equilibrium height and friction coefficient in pipeline transportation of gas hydrate-bearing sediment particles," *J. Nat. Gas Sci. Eng.* **81**, 103470 (2020).
- X. Li and J. Zhao, "A unified CFD-DEM approach for modeling of debris flow impacts on flexible barriers," *Int. J. Numer. Anal. Methods Geomech.* **42**, 1643–1670 (2018).
- K. F. E. Cui, G. D. Zhou, L. Jing, X. Chen, and D. Song, "Generalized friction and dilatancy laws for immersed granular flows consisting of large and small particles," *Phys. Fluids* **32**, 113312 (2020).
- D. L. Marchisio and R. O. Fox, *Computational Models for Polydisperse Particulate and Multiphase Systems*, Cambridge Series in Chemical Engineering (Cambridge University Press, 2013).
- S. Elghobashi, "On predicting particle-laden turbulent flows," *Appl. Sci. Res.* **52**, 309–329 (1994).
- S. Sundaresan, A. Ozel, and J. Kolehmainen, "Toward constitutive models for momentum, species, and energy transport in gas-particle flows," *Annu. Rev. Chem. Biomol. Eng.* **9**, 61–81 (2018).
- C. Y. Wen and Y. H. Yu, "A generalized method for predicting the minimum fluidization velocity," *AIChE J.* **12**, 610–612 (1966).
- D. A. Drew, "Mathematical modeling of two-phase flow," *Annu. Rev. Fluid Mech.* **15**, 261–291 (1983).
- D. Li, D. Marchisio, C. Hasse, and D. Lucas, "twoWayGPBEFoam: An open-source Eulerian QBMM solver for monokinetic bubbly flows," *Comput. Phys. Commun.* **250**, 107036 (2020).
- J. Wang, X. Chen, W. Bian, B. Zhao, and J. Wang, "Quantifying the non-equilibrium characteristics of heterogeneous gas-solid flow of smooth, inelastic spheres using a computational fluid dynamics-discrete element method," *J. Fluid Mech.* **866**, 776–790 (2019).
- Y. Gu, A. Ozel, J. Kolehmainen, and S. Sundaresan, "Computationally generated constitutive models for particle phase rheology in gas-fluidized suspensions," *J. Fluid Mech.* **860**, 318–349 (2019).
- P. Zhao, J. Xu, X. Liu, W. Ge, and J. Wang, "A computational fluid dynamics-discrete element-immersed boundary method for Cartesian grid simulation of heat transfer in compressible gas-solid flow with complex geometries," *Phys. Fluids* **32**, 103306 (2020).
- D. Snider, "An incompressible three-dimensional multiphase particle-in-cell model for dense particle flows," *J. Comput. Phys.* **170**, 523–549 (2001).
- V. Verma and J. T. Padding, "A novel approach to MP-PIC: Continuum particle model for dense particle flows in fluidized beds," *Chem. Eng. Sci.* **X 6**, 100053 (2020).
- R. O. Fox, F. Laurent, and A. Vié, "Conditional hyperbolic quadrature method of moments for kinetic equations," *J. Comput. Phys.* **365**, 269–293 (2018).
- S. Subramaniam, "Lagrangian-Eulerian methods for multiphase flows," *Prog. Energy Combust. Sci.* **39**, 215–245 (2013).
- M. Farzaneh, S. Sasic, A. Altmstedt, F. Johnsson, and D. Pallarès, "A novel multi-grid technique for Lagrangian modeling of fuel mixing in fluidized beds," *Chem. Eng. Sci.* **66**, 5628–5637 (2011).
- S. Deb and D. K. Tafti, "A novel two-grid formulation for fluid-particle systems using the discrete element method," *Powder Technol.* **246**, 601–616 (2013).
- N. Deen, M. Van Sint Annaland, M. Van der Hoef, and J. Kuipers, "Review of discrete particle modeling of fluidized beds," *Chem. Eng. Sci.* **62**, 28–44 (2007).
- Z. Peng, E. Doroodchi, C. Luo, and B. Moghtaderi, "Influence of void fraction calculation on fidelity of CFD-DEM simulation of gas-solid bubbling fluidized beds," *AIChE J.* **60**, 2000–2018 (2014).
- C. Goniva, C. Kloss, N. G. Deen, J. A. Kuipers, and S. Pirker, "Influence of roll-friction on single spout fluidized bed simulation," *Particuology* **10**, 582–591 (2012).
- T. Tsuji, K. Higashida, and Y. Okuyama, "Fictitious particle method: A numerical model for flows including dense solids with large size difference," *AIChE J.* **60**, 1606–1620 (2014).

- ³⁸B. J. Glasser and I. Goldhirsch, "Scale dependence, correlations, and fluctuations of stresses in rapid granular flows," *Phys. Fluids* **13**, 407–420 (2001).
- ³⁹N. Gui, X. Yang, J. Tu, and S. Jiang, "A fine LES-DEM coupled simulation of gas-large particle motion in spouted bed using a conservative virtual volume fraction method," *Powder Technol.* **330**, 174–189 (2018).
- ⁴⁰Y. Zhang, Y. Zhao, Z. Gao, C. Duan, J. Xu, L. Lu, J. Wang, and W. Ge, "Experimental and Eulerian-Lagrangian study of binary gas-solid flow containing particles of significantly different sizes," *Renewable Energy* **136**, 193–201 (2019).
- ⁴¹A. Haselbacher, F. Najjar, and J. Ferry, "An efficient and robust particle-localization algorithm for unstructured grids," *J. Comput. Phys.* **225**, 2198–2213 (2007).
- ⁴²R. Chordá, J. Blasco, and N. Fueyo, "An efficient particle-locating algorithm for application in arbitrary 2d and 3d grids," *Int. J. Multiphase Flow* **28**, 1565–1580 (2002).
- ⁴³M. Sani and M. Saidi, "A set of particle locating algorithms not requiring face belonging to cell connectivity data," *J. Comput. Phys.* **228**, 7357–7367 (2009).
- ⁴⁴G. Martin, E. Loth, and D. Lankford, "Particle host cell determination in unstructured grids," *Comput. Fluids* **38**, 101–110 (2009).
- ⁴⁵C. Fernandes, D. Semyonov, L. Ferrás, and J. Nobrega, "Validation of the CFD-DPM solver DPMFoam in OpenFOAM® through analytical, numerical and experimental comparisons," *Granular Matter* **20**, 64 (2018).
- ⁴⁶P. A. Cundall and O. D. L. Strack, "A discrete numerical model for granular assemblies," *Géotechnique* **29**, 47–65 (1979).
- ⁴⁷P. Sippola, J. Kolehmainen, A. Ozel, X. Liu, P. Saarenrinne, and S. Sundaresan, "Experimental and numerical study of wall layer development in a tribo-charged fluidized bed," *J. Fluid Mech.* **849**, 860–884 (2018).
- ⁴⁸H. Shalaby, K. Wozniak, and G. Wozniak, "Numerical calculation of particle-laden cyclone separator flow using LES," *Eng. Appl. Comput. Fluid Mech.* **2**, 382–392 (2008).
- ⁴⁹C. Song, B. Pei, M. Jiang, B. Wang, D. Xu, and Y. Chen, "Numerical analysis of forces exerted on particles in cyclone separators," *Powder Technol.* **294**, 437–448 (2016).
- ⁵⁰J. Hernandez, "Influence of drag laws on segregation and bubbling behavior in gas-fluidized beds," Ph.D. thesis (University of Colorado, 2008).
- ⁵¹D. Gidaspow, *Multiphase Flow and Fluidization: Continuum and Kinetic Theory Descriptions* (Academic Press, San Diego, 1994).
- ⁵²S. Ergun, "Fluid flow through packed columns," *Chem. Eng. Prog.* **48**, 89–94 (1952).
- ⁵³L. Yao, Z. Xiao, J. Liu, Q. Zhang, and M. Wang, "An optimized CFD-DEM method for fluid-particle coupling dynamics analysis," *Int. J. Mech. Sci.* **174**, 105503 (2020).
- ⁵⁴Y. Tsuji, T. Tanaka, and T. Ishida, "Lagrangian numerical simulation of plug flow of cohesionless particles in a horizontal pipe," *Powder Technol.* **71**, 239–250 (1992).
- ⁵⁵H. Kuo, P. Knight, D. Parker, Y. Tsuji, M. Adams, and J. Seville, "The influence of DEM simulation parameters on the particle behaviour in a v-mixer," *Chem. Eng. Sci.* **57**, 3621–3638 (2002).
- ⁵⁶Z. Y. Zhou, S. B. Kuang, K. W. Chu, and A. B. Yu, "Discrete particle simulation of particle-fluid flow: Model formulations and their applicability," *J. Fluid Mech.* **661**, 482–510 (2010).
- ⁵⁷E. Berberović, N. P. van Hinsberg, S. Jakirlić, I. V. Roisman, and C. Tropea, "Drop impact onto a liquid layer of finite thickness: Dynamics of the cavity evolution," *Phys. Rev. E* **79**, 036306 (2009).
- ⁵⁸D. Li and H. Christian, "Simulation of bubbly flows with special numerical treatments of the semi-conservative and fully conservative two-fluid model," *Chem. Eng. Sci.* **174**, 25–39 (2017).
- ⁵⁹H. Jasak, "Error analysis and estimation for the finite volume method with applications to fluid flows," Ph.D. thesis (Imperial College London, University of London, 1996).
- ⁶⁰L. Silva and P. Lage, "Development and implementation of a polydispersed multiphase flow model in OpenFOAM," *Comput. Chem. Eng.* **35**, 2653–2666 (2011).
- ⁶¹Z. Zhang, S. Ye, B. Yin, X. Song, Y. Wang, C. Huang, and Y. Chen, "A semi-implicit discrepancy model of Reynolds stress in a higher-order tensor basis framework for Reynolds-averaged Navier–Stokes simulations," *AIP Adv.* **11**, 045025 (2021).
- ⁶²C. M. Rhie and W. L. Chow, "Numerical study of the turbulent flow past an airfoil with trailing edge separation," *AIAA J.* **21**, 1525–1532 (1983).
- ⁶³S. Zhang, X. Zhao, and S. Bayyuk, "Generalized formulations for the Rhie–Chow interpolation," *J. Comput. Phys.* **258**, 880–914 (2014).
- ⁶⁴R. Issa and P. Oliveira, "On the numerical treatment of interphase forces in two-phase flow," in *Numerical Methods in Multiphase Flows* (ASME, 1994), Vol. FED185, p. 10.
- ⁶⁵H. Karema and S. Lo, "Efficiency of interphase coupling algorithms in fluidized bed conditions," *Comput. Fluids* **28**, 323–360 (1999).
- ⁶⁶T. F. Miller and D. J. Miller, "A Fourier analysis of the IPSA/PEA algorithms applied to multiphase flows with mass transfer," *Comput. Fluids* **32**, 197–221 (2003).
- ⁶⁷R. Issa, A. Gosman, and A. Watkins, "The computation of compressible and incompressible recirculating flows by a non-iterative implicit scheme," *J. Comput. Phys.* **62**, 66–82 (1986).
- ⁶⁸A. Sinigersky, R. Koch, and H.-J. Bauer, "A novel application of barycentric coordinates to Lagrange two-way coupled spray calculations," *Prog. Comput. Fluid Dyn.* **13**, 337–345 (2013).
- ⁶⁹G. Capodaglio and E. Aulisa, "A particle tracking algorithm for parallel finite element applications," *Comput. Fluids* **159**, 338–355 (2017).
- ⁷⁰D. C. Rapaport, *The Art of Molecular Dynamics Simulation*, 2nd ed. (Cambridge University Press (CUP), New York, USA, 2004).
- ⁷¹R. Sun and H. Xiao, "Diffusion-based coarse graining in hybrid continuum–discrete solvers: Theoretical formulation and a priori tests," *Int. J. Multiphase Flow* **77**, 142–157 (2015).
- ⁷²C. Müller, D. Holland, A. Sederman, S. Scott, J. Dennis, and L. Gladden, "Granular temperature: Comparison of magnetic resonance measurements with discrete element model simulations," *Powder Technol.* **184**, 241–253 (2008).
- ⁷³C. R. Müller, S. A. Scott, D. J. Holland, B. C. Clarke, A. J. Sederman, J. S. Dennis, and L. F. Gladden, "Validation of a discrete element model using magnetic resonance measurements," *Particuology* **7**, 297–306 (2009).
- ⁷⁴W. D. Fullmer, G. Liu, X. Yin, and C. M. Hrenya, "Clustering instabilities in sedimenting fluid–solid systems: Critical assessment of kinetic-theory-based predictions using direct numerical simulation data," *J. Fluid Mech.* **823**, 433–469 (2017).
- ⁷⁵M. Ray, F. Chowdhury, A. Sowinski, P. Mehrani, and A. Passalacqua, "Eulerian modeling of charge transport in bi-disperse particulate flows due to tribo-electrification," *Phys. Fluids* **32**, 023302 (2020).
- ⁷⁶S. K. Panda, K. Singh, K. Shenoy, and V. V. Buwa, "Numerical simulations of liquid-liquid flow in a continuous gravity settler using OpenFOAM and experimental verification," *Chem. Eng. J.* **310**, 120–133 (2017).
- ⁷⁷S. Ergun and A. Oring, "Fluid flow through randomly packed columns and fluidized beds," *Ind. Eng. Chem.* **41**, 1179–1184 (1949).
- ⁷⁸H. J. Subramani, M. Mothivel Balaiyya, and L. R. Miranda, "Minimum fluidization velocity at elevated temperatures for Geldart's group-b powders," *Exp. Therm. Fluid Sci.* **32**, 166–173 (2007).
- ⁷⁹M. Lappa, "On the transport, segregation, and dispersion of heavy and light particles interacting with rising thermal plumes," *Phys. Fluids* **30**, 033302 (2018).
- ⁸⁰R. Chassagne, R. Maurin, J. Chauchat, J. M. N. T. Gray, and P. Frey, "Discrete and continuum modelling of grain size segregation during bedload transport," *J. Fluid Mech.* **895**, A30 (2020).
- ⁸¹C.-H. Lee, Z. Huang, and M.-L. Yu, "Collapse of submerged granular columns in loose packing: Experiment and two-phase flow simulation," *Phys. Fluids* **30**, 123307 (2018).
- ⁸²C.-H. Lee and Z. Huang, "A two-phase flow model for submarine granular flows: With an application to collapse of deeply-submerged granular columns," *Adv. Water Resour.* **115**, 286–300 (2018).
- ⁸³R. Zhang, D. Su, G. Lei, and X. Chen, "Three-dimensional granular column collapse: Impact of column thickness," *Powder Technol.* **389**, 328–338 (2021).