Research on Particle Collision Forward Search Algorithm and the CFD-DEM Variable Time Step Coupling Calculation Method

Xuefei Wang
Northeast Petroleum University
Suling Wang (wl20220312@163.com)
Northeast Petroleum University
Ming Wang
Northeast Petroleum University
Xuemei Li
Qiqihar University
Lin Chi
Qiqihar University
Dandan Sun
Eighth Oil Production Plant, Daqing Oil Field Co

Research Article

Keywords: CFD-DEM, particle collision algorithms, solid–liquid two-phase flow

Posted Date: November 30th, 2021

DOI: https://doi.org/10.21203/rs.3.rs-1046946/v1

Research on particle collision forward search algorithm and the CFD-DEM variable time step coupling calculation method

Xuefei Wang [1, 2, 5], Suling Wang[1, ✉], Ming Wang [1], Xuemei Li[2, 5], Lin Chi[3], Dandan Sun[4]

[1] School of Mechanical Science and Engineering, Northeast Petroleum University 163318 Daqing, Chaina. [2] School of Mechanical and electrical engineering, Qiqihar University 161006 Qiqihar, Chaina. [3] School of Chemistry and Chemical Engineering, Qiqihar University 161006 Qiqihar, Chaina. [4] Eighth Oil Production Plant, Daqing Oil Field Co,163318 Daqing, Chaina. [5] Heilongjiang Province Intelligent Manufacturing Equipment Industrialization Collaborative Innovation Center, 161006, Qiqihar, Chaina.

✉email: wl20220312@163.com

Abstract: In CFD-DEM coupling calculations, an excessively large selection for particle calculation time step affects the calculation accuracy, and an extremely small selection affects the calculation efficiency. A search ball is constructed by taking each target particle as the center particle with the fastest displacement in the calculation domain. Subsequently, the particles that may collide are screened to establish a search list, and a forward search method is used to determine particle collisions. Finally, a particle calculation time step is proposed. The improved DEM method, which automatically adjusts the collision time, resolves the contradiction between particle calculation time step selection, accuracy, and efficiency. The relative error between the numerical simulation results of particle collision and the theoretical solution was less than 3%. The three calculation time steps selected in this study can guarantee excellent calculation accuracy and efficiency. For multi-particle and fluid coupling simulations, the traditional CFD-DEM method selects 10^{-7}s or less in the calculation time step to obtain an accurate solution. The method proposed in this paper selects 10^{-5}s to obtain an accurate solution, which increased the calculation efficiency by 19.8%.

Keywords：CFD-DEM; particle collision algorithms; solid–liquid two-phase flow

1. Introduction

Solid–liquid two-phase flow exists in many chemical, petroleum, agricultural, biochemical, and food production processes [1]. Numerical methods are widely used to improve process equipment design and avoid verbose and time-consuming experiments [2]. Currently, there are two commonly used numerical models to describe the solid–liquid two-phase flow: the two-fluid model (TFM) and the computational fluid dynamics-discrete particle model (CFD-DEM) [3]. In a two-fluid model (TFM), fluids and solids are regarded as quasi-fluids or quasi-continuums that penetrate each other [4]. In the discrete particle model (CFD-DEM), the particle is considered a discrete medium, and the discrete element method is used to solve it. Newton's second law of motion was used to calculate the trajectory of the particle [5]. The fluid adopts the Navier-Stokes (N-S) and continuity equations, with the energy equation sometimes applied. The CFD and DEM coupling interface realize the interaction between particles and fluid, wherein CFD simulates fluid flow, and DEM simulates particle motion and collision between particles [6]. Unlike the two-fluid model (TFM), the discrete particle model (CFD-DEM) can track each moving particle [7]. Hence, the discrete particle model (CFD-DEM) is widely used when considering the impact of particle collision.

In traditional CFD-DEM coupling calculations, the time step of fluid calculation is set as an integral multiple of the time step of particle calculation ($\Delta t_f = j \cdot \Delta t_p$) to reconcile the disparity between the fluid calculation efficiency and the particle calculation accuracy [8]. The fluid was used as a continuous medium. A relatively large calculation time step helps to optimize the fluid calculation efficiency. In contrast, a smaller time step can help avoid missed particle collisions and ensure particle calculation accuracy. Usually, a fixed time step Δt_p is set, and after solving each calculation time step, the collision between any two particles is checked. Assuming that the collision occurs, the soft sphere or the hard sphere model is used to analyze it. The problem with this method is that a pair of particles may collide, overlap, and then separate again in a time step, leaving no evidence of collision at the end. Because every particle pair has a different running time before the collision, it is impossible to guarantee that the calculation time step ends at the same time as the collision. Reducing the particle calculation time step cannot completely solve the problem described above. Instead, it increases the number of particle calculation iterations, affecting particle

calculation efficiency. The particle collision algorithm is the key to improving particle calculation efficiency and collision calculation accuracy.

Many researchers have studied particle collision algorithms, such as the grid element method proposed by Hockney and Eastwood [9]. This algorithm divides the analysis space into regular grids and assigns a grid for each particle. The particle or boundary in its grid and adjacent grid are the neighboring units for each particle. When detecting particle collisions, it is only necessary to detect the unit and adjacent unit of the particle. Allen and Tildesley [10] proposed a neighbor-chain list method. A ball is drawn with a particle as the center and a certain length r (usually $r > 1.5R_{\max}$, where $R_{\max}$ is the radius of the largest particle in this system) as the radius. Then, other particles in the ball are used as the neighboring units of the particle, and collision detection is required for them. Cohen [11], Baraff [12], and Lin [13] proposed the bounding box method. The bounding boxes parallel to the coordinate plane and connected with the particles are projected vertically to the coordinate axes x and y. If two particles collide, the projection of the boundary box on both coordinate axes must overlap.

Based on grid elements, neighbor lists, and boundary box algorithms, researchers have studied the accuracy [14], speed [15][16], and process [17] adaptability with grid optimization [18], size, and shape. For example, Nezami [19] et al. proposed a new fast common plane (FCP) collision search algorithm for solving the common plane between polygon particles. The no-binary search (NBS) contact collision detection algorithm, in which the total detection time is proportional to the total number of particles, was proposed by Munjiza [20] et al. Chenyinmi [21] proposed a direct simulation optimization algorithm for particle collisions in a three-dimensional space. Pischke [22] proposed a hybrid collision algorithm that combines deterministic collision thought with random collision thought. Wang et al. [23] proposed a two-step accurate collision detection fast algorithm For 'bounding sphere-maximum detection area' preprocessing. The determination of particle collision in the above algorithm adopts a fixed time step to perform propulsion calculations.

This study proposes a particle collision forward search algorithm to resolve the discrepancy between particle time step selection, calculation accuracy, and efficiency in the CFD-DEM coupling calculation. In the proposed method, the particle's initial calculation time step has no influence on the particle calculation accuracy. Furthermore, the particle collision time step automatically adjusts the particle calculation time step.

Thus, under the premise of accurate description of particle collision, a large time-step propulsion calculation is realized, and the and the number of particle collisions determines the number of fluid-particle coupling.

2. Particle-fluid coupling calculation method

2.1. Fluid control equation

In the coupling calculation of particles and fluid, the position of particles is projected onto the CFD calculation unit grid, and the fluid volume fraction occupied by the residual fluid in the computational element mesh α_f is calculated. Next, the fluid force and torque are calculated using the fluid velocity $\boldsymbol{u}_f$, particle velocity $\boldsymbol{u}_p$, fluid density, ρ_f and fluid dynamic viscosity μ. The volume-average reaction phase of these fluid forces and torque is $\boldsymbol{S}_f$. The iterative process of the particle-fluid coupling calculation is shown in Fig. 1.

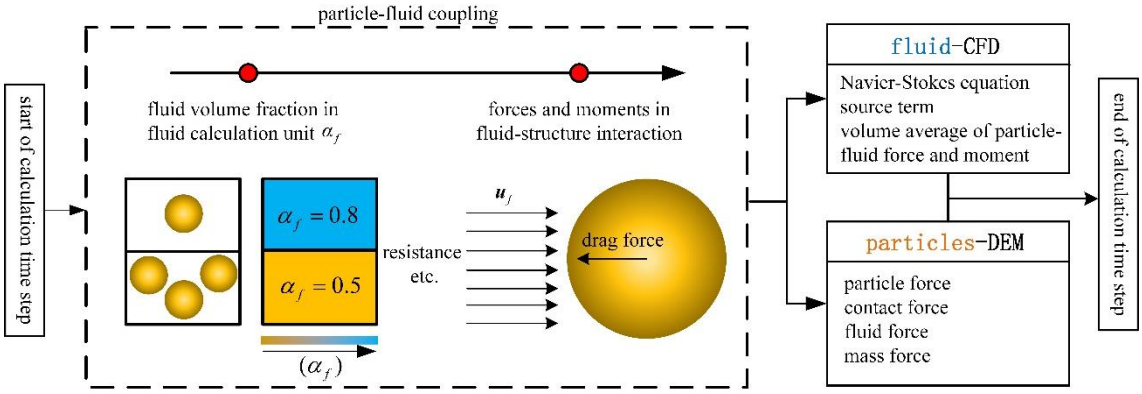

Fig. 1. Schematic diagram of particle and fluid coupling

The fluid volume fraction in the unit[24] α_f is calculated as follows:

$$\alpha_f = 1\text{-}\alpha_p = 1 - \frac{1}{V_{\mathrm{E}}} \sum_{i=1}^{n_p} V_p^{(i)} \tag{1}$$

where α_p is the volume fraction of particles in a unit, V_E is the unit volume, and $V_p^{(i)}$ is the volume of a single particle in a unit. The source of interaction between the fluid and the particle in the fluid calculation unit $\boldsymbol{S}_f$ [25] is calculated as:

$$\boldsymbol{S}_f = \frac{\sum_{j=1}^{n_p} \boldsymbol{F}_d}{V_E} \tag{2}$$

where V_E represents the fluid unit volume, α_p represents the volume fraction of particles in the fluid unit, $V_p^{(i)}$ is the i th particle volume in the fluid unit, $\boldsymbol{F}_d$ is the i th particle in a fluid unit subjected to drag force, and n_p represents the number of particles in the fluid unit.

The conservation equation of fluid quality[26] can be calculated by：

$$\frac{\partial(\alpha_f \rho_f)}{\partial t} + \nabla \cdot \rho_f \alpha_f \boldsymbol{u}_f = 0 \tag{3}$$

The conservation equation of fluid momentum is ：

$$\frac{\partial}{\partial t}\left(\alpha_f \rho_f \boldsymbol{u}_f\right) + \nabla \cdot \left(\alpha_f \rho_f \boldsymbol{u}_f \boldsymbol{u}_f\right) = -\alpha_f \nabla p_f + \alpha_f \nabla \cdot \overline{\overline{\tau}} + \boldsymbol{S}_f + \alpha_f \rho_f \boldsymbol{g} \tag{4}$$

where ρ_f fluid density, $\boldsymbol{u}_f$ fluid velocity vector, and $\overline{\overline{\tau}}$ fluid viscous stress tensor,

On the left side of the momentum equation are the time-varying items caused by the instability of the flow field and the migration items caused by the non-uniformity of the flow field, and on the right are the pressure gradient item, the viscous force item, the momentum exchange source item, and the gravity item.

The expression $\overline{\overline{\tau}}$ can be defined as [27]：

$$\overline{\overline{\tau}} = \mu_f (\nabla \boldsymbol{u}_f - \nabla u_f^T) - (\lambda_f - \frac{2}{3}\mu_f)(\nabla \cdot \boldsymbol{u}_f)\overline{\overline{I}} \tag{5}$$

where μ_f is the fluid viscosity coefficient, λ_f is the fluid volume viscosity, and $\overline{\overline{I}}$ is the turbulent kinetic energy intensity of the fluid item.

2.2. Particle dynamics equation and force analysis

(1) Particle dynamics equation:

The translation and rotation of particles in a particle-fluid flow system obey Newton's second law. The dynamics equation is [28]:

$$m_i \frac{d\boldsymbol{u}_i}{dt} = (\boldsymbol{F}_{cn,ij} + \boldsymbol{F}_{ct,ij}) + \boldsymbol{F}_f + \boldsymbol{F}_g \tag{6}$$

$$I_i \frac{d\boldsymbol{\omega}_i}{dt} = \boldsymbol{M}_c \tag{7}$$

where m_i represents the quality of the particle i, $\boldsymbol{u}_i$ represents the translational velocity of the particle, $\boldsymbol{F}_{cn,ij}$ and $\boldsymbol{F}_{ct,ij}$ represent the contact collision force between particle i and particle j and particle i and the wall, respectively, $\boldsymbol{F}_f$ represents the resultant fluid force on particles, $\boldsymbol{F}_g$ represents the quality force of particles, I_i represents the inertia moment of particle i, $\boldsymbol{\omega}_i$ represents the angular velocity of particles, and $\boldsymbol{M}_c$ represents the torque caused by particle collision.

(2) Particles subjected to fluid force.

The particles are wrapped inside the fluid and follow fluid motion. The unstable flow inside the fluid generates various forces on the particles, and the resultant fluid force on the particles is $\boldsymbol{F}_f$, calculated by：

$$\boldsymbol{F}_f = \boldsymbol{F}_d + \boldsymbol{F}_p + \boldsymbol{F}_{vm} + \boldsymbol{F}_{saff} + \boldsymbol{F}_{ml} + \boldsymbol{F}_b \tag{8}$$

where $\boldsymbol{F}_d$, $\boldsymbol{F}_p$, $\boldsymbol{F}_{vm}$, $\boldsymbol{F}_{saff}$, $\boldsymbol{F}_{ml}$, and $\boldsymbol{F}_b$ represent the resistance of particles (drag), pressure gradient force, virtual mass force, Suffman shear lift, Magnus rotation lift, and Bethel force, respectively, as shown in Table 1. Reference [29] [30] shows that resistance is the main force.

Table 1. Force acting on particles in fluid

force or moment	equation
pressure gradient force	$\boldsymbol{F}_p = -\pi D_p^3 (dp_p / dx) / 6$
virtual mass force [31]	$\boldsymbol{F}_a = 2/3\pi D_p^3 \rho_f d(\boldsymbol{u}_f - \boldsymbol{u}_p) / dt$
Suffman shear lift [32]	$\boldsymbol{F}_{saff} = 1.62 D_P^2 \left(\mu\rho_f\right)^{1/2} k^{1/2} \left(\boldsymbol{u_p} - \boldsymbol{u_f}\right)$
Magnus rotation lift [33]	$\boldsymbol{F}_{ml} = \pi D_p^3 \rho_f \boldsymbol{\omega_p} (\boldsymbol{u_p} - \boldsymbol{u_f})[1 + o(\mathrm{Re})]$
Bethel force [34]	$\boldsymbol{F}_b = 6 D_p^2 \sqrt{\pi\rho_f \mu} \int_{t_0}^{t} \sqrt{(t-\tau)}^{-1} \frac{d(u_f - u_p)}{d\tau} d\tau$

where ρ_f represents the density of the fluid, $\boldsymbol{u}_p$ represents the particle motion velocity, dp_p / dx represents the pressure gradient on the microelement surface, $\boldsymbol{\omega}_p$ represents the particle's rotation angular

velocity, $o(\mathrm{Re})$ represents the lowest order of Re in the series remainder that is not explicitly written as 1, and D_p represents the particle diameter.

The drag force computation formula proposed by Di Felice is[35]:

$$\boldsymbol{F}_d = \frac{1}{8} C_d \pi \rho_f D_p^2 \left| \boldsymbol{u}_f - \boldsymbol{u}_p \right| (\boldsymbol{u}_f - \boldsymbol{u}_p) \alpha_f^{-(\chi-1)} \tag{9}$$

$$\chi = 3.7 - 0.65 \exp[-(1.5 - \lg \mathrm{Re}_p)^2 / 2)] \tag{10}$$

$$C_d = (0.63 + \frac{4.8}{\mathrm{Re}_p})^2 \tag{11}$$

$$\mathrm{Re}_p = \frac{\alpha_f \rho_f D_p \left| \boldsymbol{u_f} - \boldsymbol{u_p} \right|}{\mu_f} \tag{12}$$

where C_d is the resistance coefficient, ρ_p is the particle density, and Re_p is the Reynolds number of the particle.

2.3. Contact force of particles

To establish a calculation method for particle collisions, it is necessary to assume that: (i) The particles are rigid spherical bodies. (ii) There are only two body collisions between particles. (iii) The particle densities are the same, the diameters can be different, and the initial state of the particle group is uniformly distributed. If any two particles i and j are taken, the diameter of the particle i is set as D_p^i, the mass is set to m_i, the diameter of the particle j is set as D_p^j, and the mass is set as m_j. The mass ratio of the two particles is given by: $q = \frac{m_i}{m_j}$

According to the momentum theorem, if two particles collide, the following equation must be satisfied:

$$-m_i \left(\boldsymbol{u}_i' - \boldsymbol{u}_i \right) = m_j \left(\boldsymbol{u}_j' - \boldsymbol{u}_j \right) = \boldsymbol{J} \tag{13}$$

$$I_i \left(\boldsymbol{\omega}_i' - \boldsymbol{\omega}_i \right) = \frac{D_p^i}{2} \left(\boldsymbol{k} \times \boldsymbol{J} \right), -I_j \left(\boldsymbol{\omega}_j' - \boldsymbol{\omega}_j \right) = \frac{D_p^j}{2} \left(\boldsymbol{k} \times \boldsymbol{J} \right) \tag{14}$$

where $\boldsymbol{u}_i$ and $\boldsymbol{u}_i'$ represent the leveling velocity before and after particle i collisions, respectively, $\boldsymbol{\omega}_i$ and $\boldsymbol{\omega}_i'$ represent the rotational speed before and after particle i collisions, respectively, $\boldsymbol{k}$

represents the normal unit vector of particle relative velocity, $\boldsymbol{J}$ represents the impulse in particle collision, and I_i and I_j represent the rotational inertia of particle and i particle j, respectively.

According to the elastic mechanics collision theory the collision time t_c of the two particles is:

$$t_c = 1.47\left(\frac{5M}{4w}\right)\left(\frac{1}{\left|\boldsymbol{u}_{ij}^*\right|}\right)\left(1+\frac{1}{e}\right) \tag{15}$$

where $\boldsymbol{u}_{ij}^*$ represents the relative velocity of particles after correction, $\boldsymbol{u}_{ij}^* = (1-\chi)\boldsymbol{u}_{ij} + \chi\boldsymbol{u}_{ij}'$, $\boldsymbol{u}_{ij}'$ represents the relative velocity after collision, χ is the weight coefficient, M represents the equivalent mass of particle i and particle j, $M = m_i m_j / m_i + m_j$, $w = \frac{4}{3\pi(\alpha_1+\alpha_2)}\sqrt{\frac{D_p^i D_p^j}{D_p^i + D_p^j}}$ represents the correlation coefficient of the collision force, $\alpha_1 = \left(1-\mu_i^2\right)/\pi E_i$, $\alpha_2 = \left(1-\mu_j^2\right)/\pi E_j$, μ_i and μ_j represent the Poisson ratio of particles i and j, respectively, and E_i and E_j represent the elastic modulus of the particles i and j, respectively.

According to the impulse theorem, the formula for calculating the collision force of the two particles is:

$$\boldsymbol{F}_{c,ij} = \frac{\boldsymbol{J}}{t_c} = \frac{e\left(\boldsymbol{k}\cdot\boldsymbol{u}_{ij}^*\right)\boldsymbol{k}E_iE_j}{E_i\left(1-\mu_i^2\right)+E_j\left(1-\mu_j^2\right)}\sqrt{\frac{D_p^i D_p^j}{D_p^i + D_p^j}} \tag{16}$$

First, based on the relative velocity $\boldsymbol{u}_{ij}$ of collision particles i and j, the particle slip is determined, and the velocity after collision is calculated. Then, using the interpolation algorithm, the relative velocities before and after particle collision $\boldsymbol{u}_{ij}$ and $\boldsymbol{u}_{ij}'$ are weighted and interpolated to obtain the $\boldsymbol{u}_{ij}^*$ particle collision time t_c and collision force $\boldsymbol{F}_{c,ij}$ by substituting them into equations (14) and (15). Next, they are substituted into equations (9) and (10) to obtain the velocity after particle collision. Finally, the set value of the velocity after the collision is compared with the obtained value; if the difference between the two satisfies the speed convergence tolerance ε_u^{tol}, the collision calculation is completed; if it is not satisfied, these steps are repeated until the speed convergence tolerance ε_u^{tol} is satisfied [36].

2.4. Fluid-particle coupling calculation method

In the calculation of fluid-particle coupling, the influence of fluid on particles is determined by substituting the fluid force acting on particles into the particle motion equation to solve the particle motion, and the influence of particles on the fluid is determined by replacing the fluid volume fraction α_f and momentum exchange force source term $\boldsymbol{S}_f$ into the fluid control equation. The movement of particles directly changes the fluid volume fraction α_f and momentum exchange force source term $\boldsymbol{S}_f$. Therefore, the convergence criterion of fluid-particle coupling is established using the variation of the fluid volume fraction α_f and momentum exchange force source term $\boldsymbol{S}_f$.

$$\left\|\alpha_f^N - \alpha_f^0\right\| \le \varepsilon_{f,\alpha}^{tol} \tag{17}$$

$$\left\|\boldsymbol{S}_f^N - \boldsymbol{S}_f^0\right\| \le \varepsilon_{f,\alpha}^{tol} \tag{18}$$

where $\varepsilon_{f,\alpha}^{tol}$ represents the convergence tolerance of the fluid volume fraction, $\varepsilon_{f,S}^{tol}$ represents the convergence tolerance of the momentum exchange force source term, $\boldsymbol{S}_f^0$ and $\boldsymbol{S}_f^N$ represent the starting and end momentum exchange force source terms, respectively, α_f^0 and α_f^N represent the starting and ending fluid volume fractions, respectively.

As shown in Fig. 2, the SIMPLE algorithm is used to solve the fluid control equation. First, the fluid force of particles is calculated based on the relative velocity of particles and fluid. Then, it is transmitted to the particles, and the DEM solver is started. The particle motion equation is used to solve the particle velocity at the current time and judge the particle collision in the initial calculation time step. The collision time step is taken as the particle calculation time step to advance. The volume fraction α_f and momentum exchange force source term of the fluid $\boldsymbol{S}_f$ are calculated based on the change in the particle position and force in the computational domain. If the convergence criterion is not satisfied, the DEM solver is stopped, and the CFD solution is started.

The CFD solution is shown in Fig. 2 (a), The fluid domain was initialized, and the fluid completed a time step calculation. The initial calculation time step was used to calculate the fluid force $\boldsymbol{F}_f$ on the particles. Once the CFD solver was over, the DEM solution was started.

The DEM solution is shown in Fig. 2 (b). The particles set the initial calculation time step $\Delta t_p' = \Delta t_f$; however, the particles do not perform the initial calculation time step calculation. The particle collision within the initial calculation time step was determined based on the position and velocity at the beginning of the particle calculation. If the particle does not collide, the particle forward time step is considered equal to the initial calculation time step of the particle $\Delta t_p = \Delta t_p'$. After the calculation, convergence was determined again. If the calculation results converge, the DEM solution is completed. If it does not converge, the time step of particle calculation is shortened to $\Delta t_p^* = \Delta t_p \cdot k^n$, and DEM is solved again until convergence is completed. If the particle collides, the calculation of the particle collision time step Δt_c^i needs to judge the particle collision time step and the residual time of the initial calculation time step (i represents the number of particle collisions, $\sum_{i=1}^{m} \Delta t_c^{i-1} = 0$ when $i = 1$). If it is not tenable, the residual time in the initial time step of the particle is taken as the calculation time step $\Delta t_p = \Delta t_p' - \sum_{i=1}^{m} \Delta t_c^{i-1}$ to solve. After the solution was completed, the calculation results met the convergence criteria, and the DEM solution ended. If it did not meet the convergence criteria, the particle calculation time $\Delta t_p^* = \Delta t_p \cdot k^n$ was shortened, and the DEM solution was carried out again until convergence was completed. If $\Delta t_c^i < \Delta t_p' - \sum_{i=1}^{m} \Delta t_c^{i-1}$ was established, the particle moved forward $\Delta t_p = \Delta t_c^i$ with the collision time step as the particle calculation time step. After the particle collision calculation is completed, convergence judgment is performed. If it is convergent, the particle collision search is continued until it is not convergent or the initial calculation time of the particle is completed, the formula $\Delta t_c^i < \Delta t_p' - \sum_{i=1}^{m} \Delta t_c^{i-1}$ is not tenable, and the DEM solution is completed. If it does not converge, the particle collision time step $\Delta t_c^* = \Delta t_c^i \cdot k^n$ is shortened, and the DEM is solved again until convergence is completed.

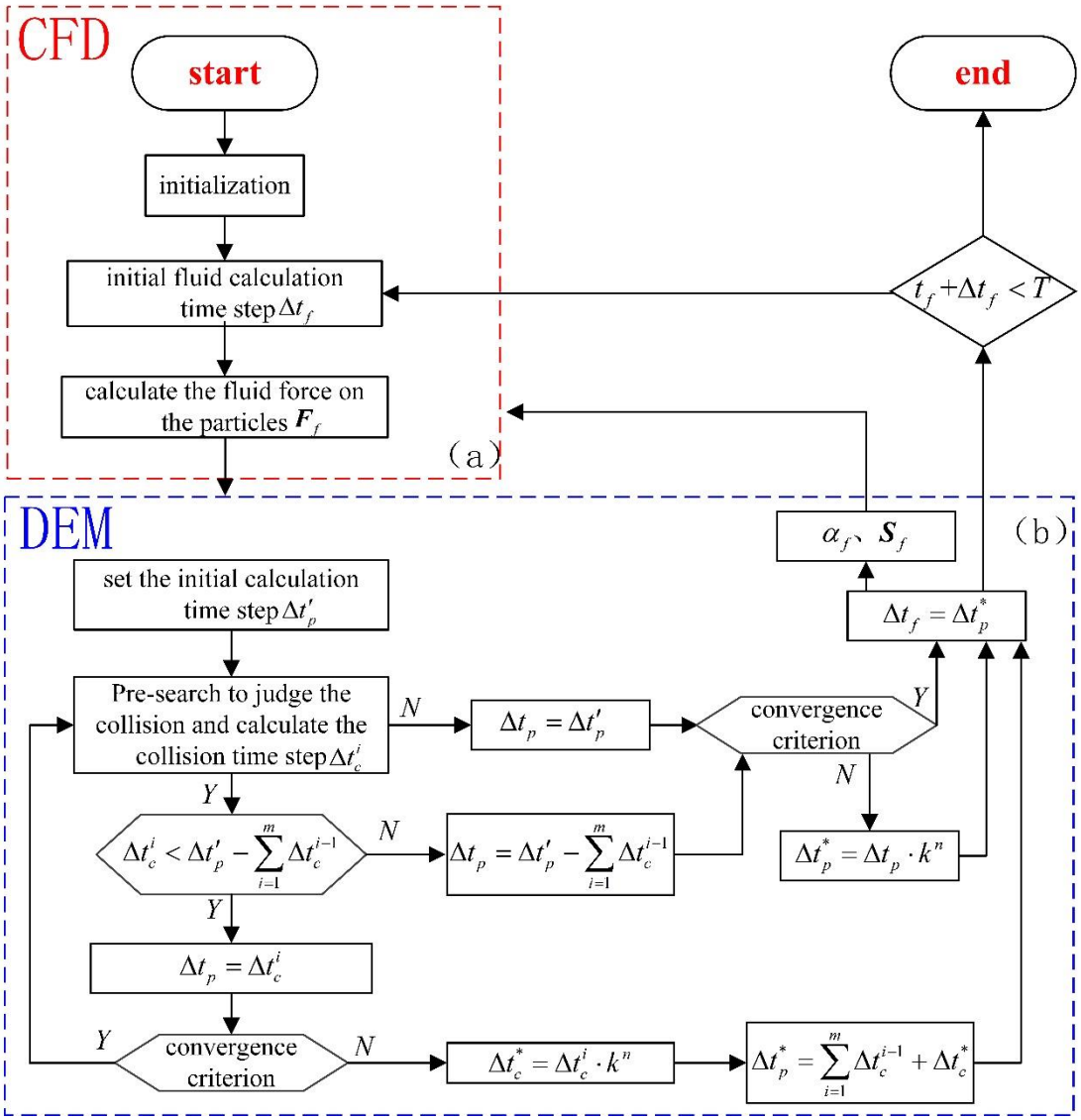

Fig. 2. Flow chart of fluid and particle coupling calculation

After the DEM solution was completed, the fluid volume fraction α_f and momentum exchange force source item S_f were calculated. The fluid calculation time step $\Delta t_f = \Delta t_p^*$ was adaptively adjusted according to the calculation time step Δt_p^* completed by the particles. The fluid domain was resolved to determine whether $t_f + \Delta t_f < T$ is tenable. If it is tenable, α_f and S_f are transferred to CFD, and the CFD solution is restarted. Otherwise, the coupling calculation is terminated.

Particle collision search is a forward search. The particle collision before $t + \Delta t_p'$ is determined according to its position and velocity at t. If this occurs, the collision time step Δt_c^i can be calculated. If there is no collision, $\Delta t_p = \Delta t_p'$ can be considered for the calculation. When there is a collision between

particles, the particle collision calculation is carried out with the collision time step as the calculation time step to avoid collisions between the missing particles. The particles were calculated over a large time step with improved calculation efficiency when there was no collision between the particles. A larger calculation time step increases the particle displacement, and makes the fluid volume fraction α_f and momentum exchange force source term $\boldsymbol{S}_f$ change too fast. The convergence criteria in equations (17) and (18) can used to limit this change, which ensures that the calculation error of the fluid-particle coupling does not change with the time step.

3. Particle collision search algorithm

3.1. Structure of particle collision search region

In particle collision search, the center particles can only collide with the surrounding particles, and a specific range around the center particles is divided into collision search areas. The method employed is as follows: Assuming that the center particle i collides with the fastest moving particle j in the computational domain at the end of the computational time step, the center particle collision search space surrounding the sphere is established as the distance R between the center particle i and the particle j as the radius. It is assumed that all particles in the sphere may collide with the center particle i, as shown in Fig. 3.

$$R = L_i + D_p^i / 2 + L_{\max} + D_p^j / 2 \tag{19}$$

where L_i represents the motion displacement of the central particle, $L_i = \boldsymbol{u}_i \cdot \Delta t_p$, $L_{\max} = \boldsymbol{u}_j \cdot \Delta t_p$ represents the motion displacement of the fastest moving particle, D_p^i represents the diameter of the central particle, D_p^j represents the diameter of the fastest moving particle, $\boldsymbol{u}_i$ represents the velocity of the center particle i, and $\boldsymbol{u}_j$ represents the motion velocity of the particle j.

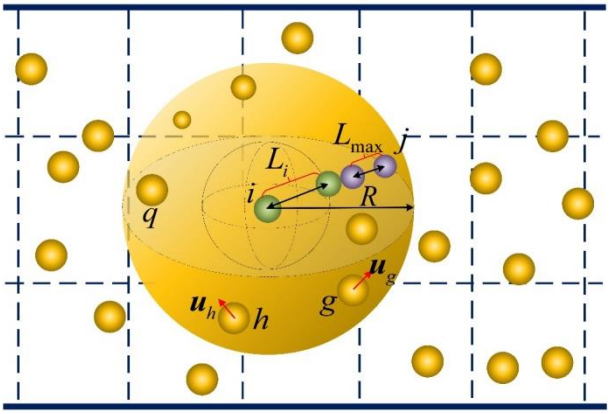

Fig. 3. Diagram of sphere surrounded by space

3.2. The establishment of particle collision list

The particles inside the sphere surrounded by the center particle collision search space were further screened to retain the possible collision particles and improve the collision search efficiency. The time center particle coordinates were set at (x_i, y_i, z_i) at time t. The coordinates of particle g were set as (x_g, y_g, z_g), making the distance between both particles $L_{ig} = \sqrt{(x_i - x_g)^2 + (y_i - y_g)^2 + (z_i - z_g)^2}$. In the calculation time step, if the sum of the movement displacement of the central particle i and the particle g was greater than the distance L_{ig} between the two particles at time t, the particle g may collide with the central particle i.

$$L_{ig} \le L_i + D_p^i / 2 + L_g + D_p^g / 2 \tag{20}$$

where $L_g = \boldsymbol{u}_g \cdot \Delta t_p$ represents the movement displacement of particle g.

Thus, a central particle collision search list was established by the remaining particles after screening.

3.3. Forward search algorithm for collision particles

The forward-searching particle collision was adopted to improve the calculation accuracy of particle collisions. The collision time of the particles was prejudged based on the velocity, direction, and other information of particles at the current time, and the calculation time step was adjusted. If the particles did not collide, a larger calculation time step Δt_p was used for the propulsion calculation.

As shown in Fig. 4, at the initial time t, the initial position of the particle i was (x_i, y_i, z_i) and particle j was (x_j, y_j, z_j), making the particle relative distance $L_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2}$. The fluid force of particles i and j were $F_{i(t)}$ and $F_{j(t)}$, the particle accelerations were $\boldsymbol{a}_{i(t)} = \dfrac{\boldsymbol{F}_{i(t)}}{m_i}$ and $\boldsymbol{a}_{j(t)} = \dfrac{\boldsymbol{F}_{j(t)}}{m_j}$, the velocities were $\boldsymbol{u}_{i(t)}$ and $\boldsymbol{u}_{j(t)}$, and the spatial positions were $\boldsymbol{r}_i$ and $\boldsymbol{r}_j$, respectively. $\boldsymbol{k}$ and represents the unit vector. The direction considered was from the center of particle i to the center of particle j.

When the distance between the particles at time t was $L_{ij} > \dfrac{1}{2}(D_p^i + D_p^j)$, the particles do not collide. In the process of particles moving forward, the velocity and acceleration generated components in the unit vector $\boldsymbol{k}$, and the two particles gradually moved closer. At time $t + \Delta t_c$, when the distance between the two particles was $L_{ij} \le \dfrac{1}{2}(D_p^i + D_p^j)$, the particles collided. The equation for solving the particle collision time is given by:

$$L_{ij} - [(\boldsymbol{u}_{i(t)} \cdot \boldsymbol{k})\Delta t_c + \frac{1}{2}\boldsymbol{a}_{i(t)} \cdot \Delta t_c^2] - [(\boldsymbol{u}_{j(t)} \cdot \boldsymbol{k})\Delta t_c + \frac{1}{2}\boldsymbol{a}_{j(t)} \cdot \Delta t_c^2] = \frac{1}{2}(D_p^i + D_p^j) \tag{21}$$

The equation can be simplified as a linear duality equation

$$\Delta t_c^2 + A\Delta t_c + B = 0 \tag{22}$$

$$\begin{cases} A = -\dfrac{2(\boldsymbol{u}_{i(t)} - \boldsymbol{u}_{j(t)}) \cdot \boldsymbol{k}}{(\boldsymbol{a}_{i(t)} - \boldsymbol{a}_{j(t)}) \cdot \boldsymbol{k}} \\ B = -\dfrac{2L - (D_p^i + D_p^j)}{(\boldsymbol{a}_{i(t)} - \boldsymbol{a}_{j(t)}) \cdot \boldsymbol{k}} \end{cases} \tag{23}$$

Two solutions $\Delta t_{c,1}, \Delta t_{c,2}$ were obtained by solving the equation, where the collision time ranged from 0 to t_p $(0 < \Delta t_c < \Delta t_p)$. If both solutions were within this range, the smaller solution was considered as the particle calculation time step. If neither solution was in this range, the particles do not collide in the calculation time step.

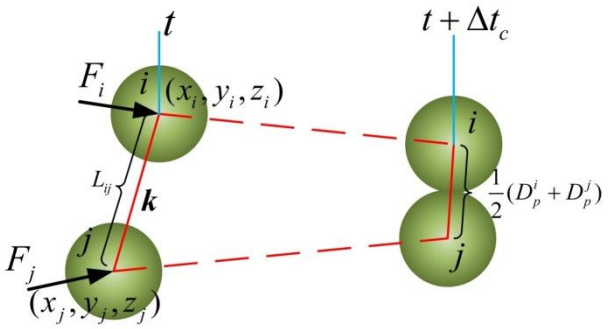

Fig. 4. Schematic diagram of the particle collision time solution

3.4. Particle collision solving process

The particle collision solving process is shown in Fig. 5, where it is first necessary to load the characteristic parameters of the model, including the number of input particles, grid parameters, and boundary conditions. The particle velocity was calculated based on the particle position and accept force, and the fastest particle velocity in the current calculation time step was determined. The radius of the space enclosing sphere was established using the sum of the center particle displacement and the fastest moving particle displacement. With n particles, $n-1$ collision lists were established and updated before calculating the time step. The particles moving out of the space surrounding the sphere were deleted from the original list, and he re-entered particles were added. Assuming that the collision search list was established after the center particle i searched and judged particle j, which was established as the center particle to establish lists; hence, it was unnecessary to re-judge particle i, which can be directly mapped from the total collision list.

The earliest collision time for each list was added to the summary table as the calculation time step. As the calculation time in the part calculation domain moved forward and performed collisions, the list was updated again; In case no collisions occurred, only the completed calculation time was subtracted from the original collision time to update the total list.

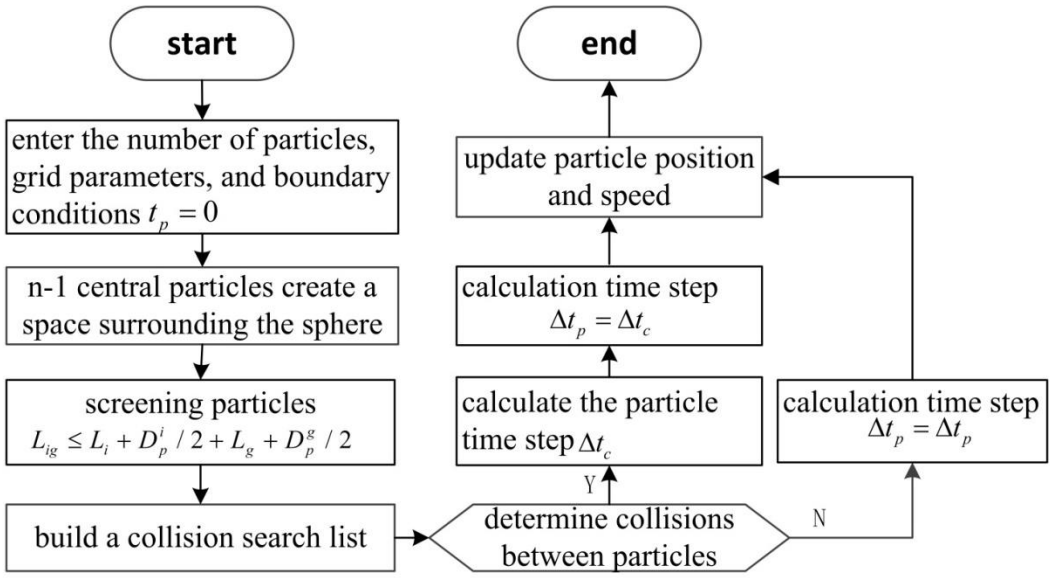

Fig. 5. Flow diagram of particle collision solution

4. CFD-DEM coupling analysis and particle collision search algorithm verification

Based on the DEM platform, the Delphi language was used to write the proposed particle collision forward search algorithm program. In addition, the proposed algorithm was compared with the existing DEM software to verify its feasibility. The simulation used an Intel Core i7-9700 @ 3.00 GHz eight-core processor, main display card AMD Radeon RX 550 Series, and 16 GB of computer memory.

4.1. Two particle collision examples

Based on the particle collision forward search algorithm in this study, the improved DEM-M method and the existing DEM method were used to numerically simulate the collision process of two particles. This method is expected to be both efficient and accurate.

As shown in Fig. 6, the distance between two centers of the circle was taken as 50 mm, the radius of two particles A and B was 5 mm, the density was 2500 $\mathrm{kg/m^3}$, Poisson ' s ratio was 0.25, and the elastic modulus was 210 GPa. Working condition 1 was defined as: particles A and B move uniformly. The initial velocity of particle A was $u_A = 3\,\mathrm{m/s}$ in the positive x axial direction, and the initial velocity of particle B was $u_B = 3\,\mathrm{m/s}$ also in the positive x direction. On the other hand, working condition 2 was defined as: particles A and B exhibit variable motion. The initial velocity of the particle A along the positive x axial direction was $u_A = 3\,\mathrm{m/s}$, and the acceleration along the same direction was $10\,\mathrm{m/s^2}$. The

initial velocity of particle B along the positive x axial direction was $u_B = 1\,\mathrm{m/s}$, and the acceleration along the same direction was $10\,\mathrm{m/s^2}$. The velocities of the A and B particles after collision were considered as u'_A and u'_B, respectively, and the collision force of particles A and B was F_{cn}.

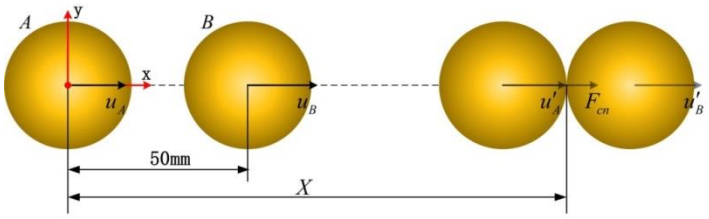

Fig. 6. Schematic diagram of particle position

DEM and DEM-M methods were adopted to simulate the chase and collision problems of the two particles, respectively. The calculation time steps were taken as 10^{-4} s, 10^{-5} s and 10^{-6} s, and the calculation time was often set to 0.1 s, Results of the collision time, collision force, collision velocity, collision position were obtained, which are listed in Table 2 together with the analytical solution as the reference value.

As shown in Fig. 7, under the condition of uniform velocity, when the DEM method calculates the time step of 10^{-4} s, and the calculated results are compared with the analytical solution, the errors in the collision velocity and the collision time were both greater than 30 %, and the error in collision force reached 138.605 %. When the calculation time step was 10^{-5} s, the maximum error in the velocity ranged from 1.47 % to 2.96 %, the maximum error in the collision position was 2.187 %, the error in collision force was 2.243 %, and the error in collision time was 2 %. The errors in the simulation results for each parameter were less than 3 %. When the calculation time step was 10^{-6} s, the maximum error in velocity ranged from 0.4 % to 0.16 %, the maximum error in collision position was 0.05 %, the error in collision force was 1.069 %, and the error in collision time was 0.5 %. The errors in the simulation results for each parameter were less than 2 %. The A2 and A3 calculation times of the working condition were 27.73 s and 44.07 s, respectively. The above analysis shows that the calculation accuracy of the existing DEM method is greatly affected by the calculation time step, and the calculation efficiency decreases significantly with a decrease in the time step. Adopting the DEM-M method, the errors in different calculation time steps between the calculation results and the analytical solution obtained were less than 2 %, in which the maximum error in velocity was 1.867 %,

the maximum error in collision position was 0.211 %, the maximum error in collision force was 1.365 %, the error in collision time was 1.5 %, and B1, B2, and B3 calculation times of the working condition were 23.01 s, 25.14 s, 38.06 s, respectively. The above analysis shows that the calculation accuracy of the improved DEM-M method of the particle collision forward search algorithm proposed in this paper is little affected by the calculation time step. Moreover, the calculation efficiency decreases with a decrease of the time step. For example, although the B2 and the A2 calculation time steps were equal, the calculation time is shortened by 2.59 s (9.34 %) when using the proposed algorithm. Furthermore, when the calculation time step B1 was one order of magnitude larger than A2, the calculation time was shortened by 4.72 s (17.02 %).

As shown in Fig. 8, under variable speed conditions, for the DEM method, when the calculation time step C1 was 10^{-4} s, the error between the calculated results and the theoretical solution was too large, and the velocity, collision force, and collision time errors were all greater than 30 %. When the calculation time steps C2 and C3 were 10^{-5} s and 10^{-6}s, respectively, the calculation results of the physical parameters were less than 3 %, and the calculation times were 32.53 s and 46.21 s. It can be observed that the calculation accuracy increases and the calculation efficiency decreases with a decrease in the time step. Using the improved DEM-M method in this study, the error between the calculation results and the theoretical results obtained was less than 2 % in different calculation time steps, in which the maximum error in velocity was -1.370 %, the maximum error in collision position was 0.181 %, the error in collision force was 1.213 %, the error of collision time was 1.5 %, and the D1, D2, and D3 calculation times were 26.5 s, 29.20 s, and 39.24 s, respectively. For D2, which was similar to the calculation time step C2, the calculation time was shortened by 3.33 s (10.23 %). For D1, which had an order of magnitude larger than the C2 calculation time step, the calculation time was shortened by 6.03 s (18.54 %). The above analysis shows that the traditional DEM method is consistent with the conclusions and laws of the forward search algorithm that improved the DEM in simulating variable and uniform motion.

In summary, the DEM-M method proposed in this study was used to numerically simulate the uniform and variable motion of particles. The calculation results were barely affected by the size of the selected time step. When the calculation time step was one order of magnitude larger than that of DEM, the calculation efficiency of uniform motion increased by 17.02 % and that of variable motion increased by 18.54 %, which

effectively solved the contradiction between the selection of the time step and the calculation accuracy and efficiency.

Table 2. Theoretical and numerical simulation results of movement and collision of two particles

Calculation method			Speed/m/s		Collision location /mm	Collision force	Moment of collision	Computing time/ s
			u'_A	u'_B	x	F_{cn} / N	t_c / s	
Uniform speed								
Theoretical solution			1.500	2.500	60.000	37.500	0.0200	-------
DEM	A1	1×10^{-4}	0.875	3.497	54.350	89.477	0.0138	20.800
		error %	-41.667	39.88	-9.417	138.605	-31.000	-------
	A2	1×10^{-5}	1.522	2.426	61.312	38.341	0.0204	27.730
		error %	1.470	-2.960	2.187	2.243	2.000	-------
	A3	1×10^{-6}	1.506	2.496	60.030	37.901	0.0201	44.070
		error %	0.400	-0.160	0.050	1.069	0.500	-------
DEM-M	B1	1×10^{-4}	1.528	2.459	60.127	38.012	0.0203	23.010
		error %	1.867	-1.640	0.211	1.365	1.500	-------
	B2	1×10^{-5}	1.517	2.478	60.107	37.954	0.0202	25.140
		error %	1.133	-0.880	0.178	1.210	1.000	-------
	B3	1×10^{-6}	1.509	2.489	60.047	38.011	0.0201	38.060
		error %	0.600	-0.440	0.078	1.363	0.500	-------
Variable speed								
Theoretical solution			1.700	2.700	62.000	37.500	0.0200	-------
DEM	C1	1×10^{-4}	0.704	4.389	72.157	110.012	0.0297	22.040
		error %	-58.588	62.556	16.382	193.365	48.500	-------
	C2	1×10^{-5}	1.683	2.778	62.087	37.450	0.0203	32.530
		error %	-1.000	1.888	0.140	-0.133	1.500	--------
	C3	1×10^{-6}	1.703	2.698	62.020	37.012	0.0201	46.210
		error %	0.176	-0.407	0.032	-1.301	0.500	-------
DEM-M	D1	1×10^{-4}	1.719	2.663	62.112	37.955	0.0203	26.500
		error %	1.118	-1.370	0.181	1.213	1.500	-------
	D2	1×10^{-5}	1.689	2.725	62.070	37.824	0.0202	29.200
		error %	-0.647	0.926	0.113	0.864	1.000	-------
	D3	1×10^{-6}	1.706	2.687	62.004	37.579	0.0201	39.240
		error %	0.353	-0.481	0.006	0.210	0.500	-------

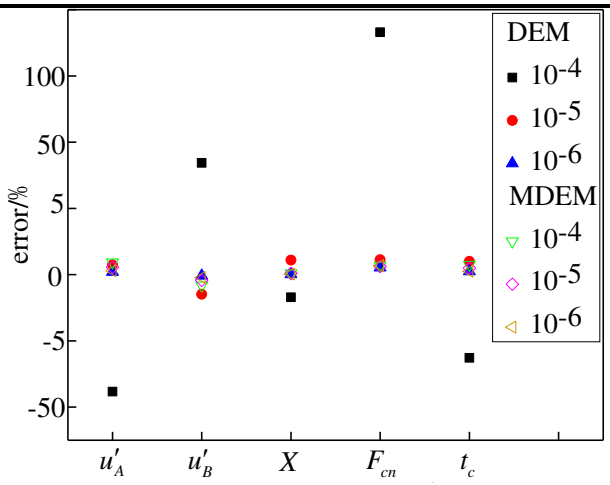

Fig. 7. Error analysis diagram of two particles in uniform motion

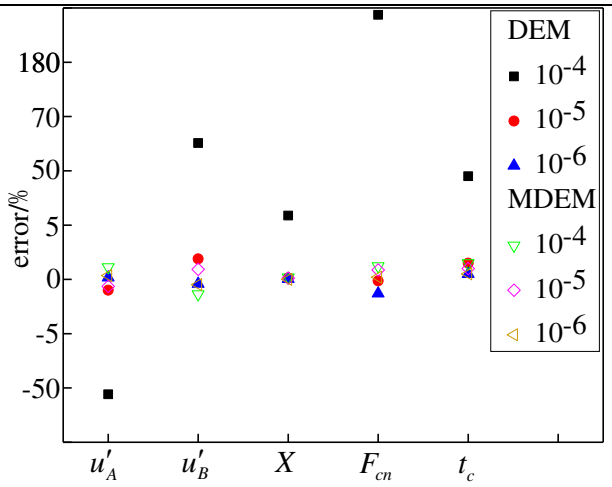

Fig. 8. Error analysis diagram of the variable speed movement of two particles

4.2. Multiple particle collision examples

The initial positions of the particles are shown in Fig. 9, where 12 particle spacings of 20 mm were arranged in a square, numbered as a, b, c, e, f, g, h, i, j, k and, l respectively. The particle material properties were the same as those in the calculation example in section 4.1. The horizontal and vertical velocities of particle a and b were 5 m/s., The vertical velocity of particle c was 2 m/s. The horizontal and vertical velocities of particle d were -5 m/s and 5 m/s, respectively. The vertical velocities of particles e and f were -5 m/s. The horizontal and vertical velocities of particle g were -5 m/s. The vertical velocities of particle h and i were -5 m/s and -2 m/s, respectively. The horizontal and vertical velocities of particle j were -5 m/s., The horizontal velocities of particles k and l were 5 m/s.

Theoretical analysis shows that at 0.0020 s, six pairs of particles, a-b, a-l, d-e, g-f, g-h, and j-k will collide for the first time at positions (-15, -20) (-20, -15), (20,-15), (20,15), (15,20), and (-20,15), respectively. Continuing particle collisions are too complex for theoretical solutions to be obtained.

The DEM simulation selected the calculation time steps of 10^{-6} s, 10^{-7} s, and 10^{-8} s, and the calculation time of 0.1 s. The numerical simulation results show that 37 particle collisions occurred when the calculation time step was 10^{-6} s, and 41 particle collisions occurred when the calculation time steps were 10^{-7} s and 10^{-8} s, as listed in Table 3. The analysis shows that when the calculation time step was 10^{-6}s , the DEM missed four collisions, and the results were not reliable, as shown in Fig. 10. In contrast, when the calculation time steps were 10^{-7} s and 10^{-8} s, the first collision time of particles occurred, and the error between the collision position and the theoretical solution was less than 1 %. The calculation time was 239.18 s and 1532.56 s, respectively.

Therefore, with a decrease in the calculation time step, the calculation efficiency of the numerical simulation of multi-particle collisions was significantly reduced.

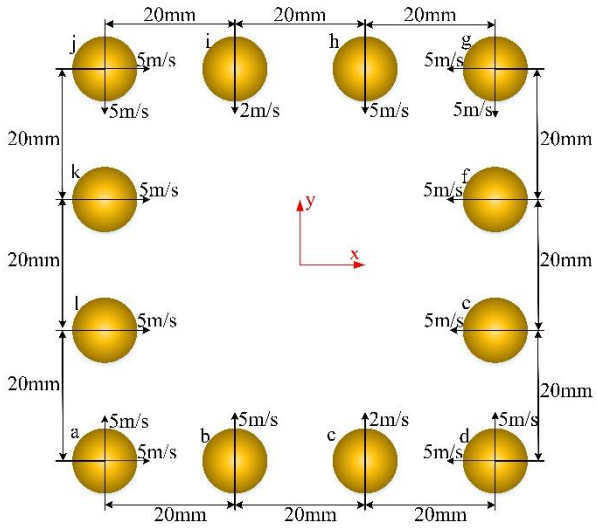

Fig. 9**. Schematic diagram of initial position of multi-particle collision**

The improved DEM-M numerical simulation results show that 41 collisions can be obtained when the calculation time steps were 10^{-5} s, 10^{-6} s, and 10^{-7} s. The results are shown in Table 3. As shown in Fig. 11, for the first collision time from different calculation time steps, the error between the collision position and the theoretical solution was less than 1 %. The calculation times were 193.75 s, 198.05 s and 247.23 s, respectively. When the calculation time steps of DEM and DEM-M were 10^{-7}s, the calculation time of DEM-M was 8.05 s longer than that of DEM owing to the collision time of forward searching particles in each calculation time step. When the calculation time step of DEM-M was extended by one order of magnitude (10^{-6} s), the calculation time was shortened by 41.13 s, that is, the calculation time reduced by 17.19 %. When the calculation time step of DEM-M was extended by two orders of magnitude (10^{-5} s), the calculation time was shortened by 45.43 s, that is, the calculation time reduced by 18.99 %. The particles with higher number of collisions were k, e, g, i, a, and c. Particles a, k, e, g had 4, 12, 12, and 6 collisions, respectively. Particles i and c had two collision trajectories, as shown in Fig. 12 (the dotted line represents DEM-M and the solid line represents DEM). It can be seen that in the simulation of multi-particle collisions, the DEM-M method proposed in this study can adopt large time steps and not only improve the calculation efficiency but also ensure calculation accuracy.

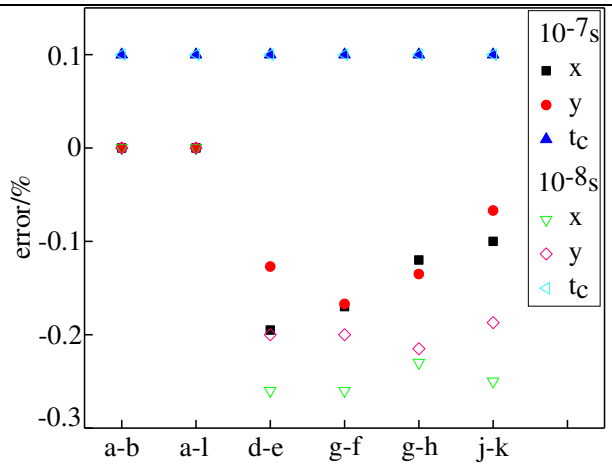

Fig. 10. DEM particle collision time and position error analysis diagram

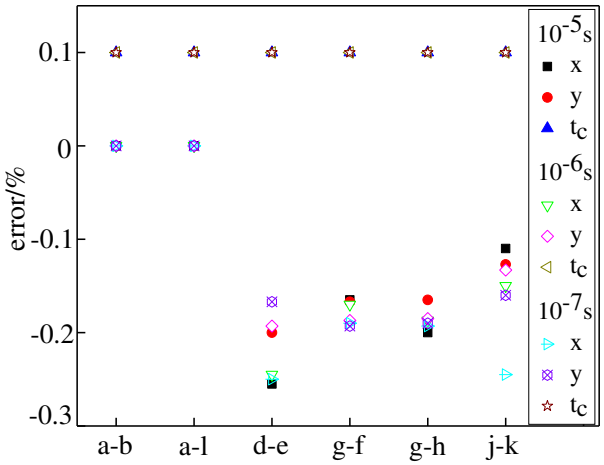

Fig. 11. DEM-M particle collision time and position error analysis diagram

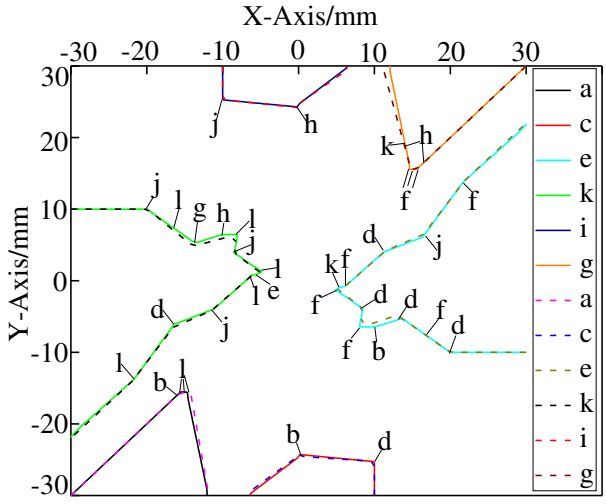

Fig. 12. Movement trajectories and collisions of more than one particle

In summary, 10^{-5} s, 10^{-6} s, and ,10^{-7} s time steps were selected to conduct numerical simulations of multiple particle collisions in this study, and accurate calculation results were obtained. The selection of the

calculation time step was one order of magnitude greater than that of the DEM method, reducing the calculation time by 18.99 %.

Table 3. Numerical simulation results of multi - particle motion and collision

Collision particle pair	DEM					
	Time step 10^{-7}s			Time step 10^{-8}s		
	Moment of collision t_c / s	Collision location /mm		Moment of collision t_c / s	Collision location /mm	
		x	y		x	y
a-b	0.0022	-15.000	-20.000	0.0022	-15.000	-20.000
a-l	0.0022	-20.000	-15.000	0.0022	-20.000	-15.000
d-e	0.0022	19.961	-14.981	0.0022	19.948	-14.970
g-f	0.0022	19.966	14.975	0.0022	19.948	14.970
g-h	0.0022	14.982	19.973	0.0022	14.965	19.957
j-k	0.0022	-19.980	14.990	0.0022	-19.950	14.972
d-c	0.0025	14.094	-22.404	0.0026	14.076	-22.390
j-i	0.0025	-14.094	22.404	0.0026	-14.076	22.390
e-f	0.0035	-14.411	0.326	0.0035	-14.387	0.327
k-l	0.0035	14.411	-0.326	0.0035	14.377	-0.327
g-k	0.0041	-12.647	10.782	0.0041	-12.607	10.776
d-e	0.0041	12.647	-10.782	0.0041	12.652	-10.782
h-k	0.0044	-3.721	8.986	0.0045	-3.719	8.987
b-e	0.0044	3.721	-8.986	0.0045	3.719	-8.987
a-l	0.0045	-13.761	-10.866	0.0046	-13.771	-10.869
f-g	0.0045	13.761	10.866	0.0046	13.771	10.869
l-k	0.0050	-9.367	-1.087	0.0051	-9.367	-1.086
l-b	0.0050	-6.204	-8.925	0.0051	-6.205	-8.924
f-e	0.0050	9.379	1.069	0.0051	9.376	1.071
f-h	0.0050	6.204	8.925	0.0051	6.204	8.925
a-l	0.0053	-12.659	-10.975	0.0054	-12.660	-10.982
f-g	0.0053	12.659	10.975	0.0054	12.660	10.981
b-c	0.0058	-0.540	-19.332	0.006	-0.550	-19.341
h-i	0.0058	0.540	19.332	0.006	0.550	19.341
j-k	0.0063	-9.255	8.092	0.0062	-9.220	8.077
d-e	0.0063	9.255	-8.092	0.0062	9.224	-8.078
k-l	0.0070	-9.019	-1.838	0.0071	-8.956	-1.913
e-f	0.0070	9.018	1.838	0.0071	8.951	1.912
k-e	0.0087	0.000	0.000	0.0088	0.000	0.000
g-h	0.0090	-2.636	13.463	0.0091	-2.602	13.454
b-d	0.0090	2.636	-13.463	0.0091	2.602	-13.454
k-l	0.0099	-8.868	-2.996	0.0095	-8.701	-2.872
e-f	0.0099	8.875	2.999	0.0095	8.701	2.872
j-k	0.0106	-6.147	5.820	0.0107	-6.108	5.786
d-e	0.0106	6.147	-5.818	0.0107	6.108	-5.786
b-l	0.0208	-14.438	-16.293	0.0223	-15.738	-17.135
f-h	0.0208	14.438	16.293	0.0223	15.738	17.135
j-e	0.0262	7.788	6.840	0.0249	6.987	6.686
k-d	0.0262	-7.789	-6.840	0.0249	-6.987	-6.686
k-l	0.0314	-19.598	-12.404	0.0419	-26.980	-17.559
e-f	0.0314	19.598	12.404	0.0419	26.980	17.559

Collision particle pair	DEM-M								
	Time step 10^{-5}s			Time step 10^{-6}s			Time step 10^{-7}s		
	Moment of collision	Collision location /mm		Moment of collision	Collision location /mm		Moment of collision	Collision location /mm	
	t_c / s	x	y	t_c / s	x	y	t_c / s	x	y
a-b	0.0022	-15.000	-20.000	0.0022	-15.000	-20.000	0.0022	-15.000	-20.000
a-l	0.0022	-20.000	-15.000	0.0022	-20.000	-15.000	0.0022	-20.000	-15.000
d-e	0.0022	19.955	-14.970	0.0022	19.951	-14.971	0.0022	19.950	-14.975
g-f	0.0022	19.967	14.975	0.0022	19.966	14.972	0.0022	19.962	14.971
g-h	0.0022	14.970	19.967	0.0022	14.972	19.963	0.0022	14.971	19.962
j-k	0.0022	-19.978	14.981	0.0022	-19.970	14.980	0.0022	-19.951	14.976
d-c	0.0025	14.081	-22.390	0.0025	14.084	-22.394	0.0026	14.077	-22.393
j-i	0.0025	-14.074	22.391	0.0025	-14.084	22.394	0.0026	-14.076	22.393
e-f	0.0035	-14.400	0.326	0.0035	-14.401	0.326	0.0035	-14.388	0.327
k-l	0.0035	14.401	-0.327	0.0035	14.401	-0.326	0.0035	14.377	-0.328
g-k	0.0041	-12.617	10.775	0.0041	-12.627	10.772	0.0041	-12.607	10.776
d-e	0.0041	12.657	-10.782	0.0041	12.647	-10.782	0.0041	12.651	-10.782
h-k	0.0044	-3.715	8.987	0.0044	-3.711	8.986	0.0045	-3.719	8.988
b-e	0.0044	3.720	-8.988	0.0044	3.720	-8.987	0.0045	3.719	-8.988
a-l	0.0045	-13.772	-10.869	0.0045	-13.762	-10.866	0.0046	-13.770	-10.869
f-g	0.0045	13.771	10.869	0.0045	13.771	10.866	0.0046	13.770	10.869
l-k	0.0050	-9.367	-1.089	0.0050	-9.367	-1.087	0.0051	-9.367	-1.086
l-b	0.0050	-6.204	-8.926	0.0050	-6.204	-8.926	0.0051	-6.206	-8.926
f-e	0.0050	9.379	1.069	0.0050	9.379	1.069	0.0051	9.377	1.072
f-h	0.0050	6.205	8.925	0.0050	6.205	8.925	0.0051	6.206	8.926
a-l	0.0053	-12.660	-10.986	0.0053	-12.669	-10.976	0.0054	-12.663	-10.983
f-g	0.0053	12.669	10.986	0.0053	12.659	10.986	0.0054	12.662	10.983
b-c	0.0058	-0.550	-19.333	0.0058	-0.550	-19.333	0.006	-0.551	-19.340
h-i	0.0058	0.551	19.343	0.0058	0.550	19.343	0.006	0.551	19.340
j-k	0.0063	-9.225	8.085	0.0063	-9.225	8.095	0.0064	-9.219	8.075
d-e	0.0063	9.315	-8.085	0.0063	9.355	-8.085	0.0064	9.227	-8.079
k-l	0.0070	-9.009	-1.837	0.0070	-9.009	-1.837	0.0073	-8.959	-1.911
e-f	0.0070	8.908	1.938	0.0070	9.008	1.838	0.0073	8.955	1.915
k-e	0.0087	0.000	0.000	0.0087	0.000	0.000	0.0088	0.000	0.000
g-h	0.0090	-2.617	13.463	0.0090	-2.627	13.463	0.0091	-2.605	13.457
b-d	0.0090	2.607	-13.453	0.0090	2.617	-13.453	0.0091	2.605	-13.457
k-l	0.0099	-8.758	-2.896	0.0099	-8.768	-2.896	0.0095	-8.707	-2.874
e-f	0.0099	8.755	2.889	0.0099	8.775	2.899	0.0095	8.702	2.871
j-k	0.0106	-6.117	5.800	0.0106	-6.127	5.800	0.0107	-6.116	5.799
d-e	0.0106	6.117	-5.788	0.0106	6.117	-5.718	0.0107	6.116	-5.799
b-l	0.0208	-15.638	-17.193	0.0208	-15.438	-16.293	0.0223	-15.747	-17.144
f-h	0.0208	15.738	16.993	0.0208	15.438	16.893	0.0223	15.747	17.144
j-e	0.0262	6.988	6.640	0.0262	6.988	6.840	0.0249	6.995	6.691
k-d	0.0262	-6.989	-6.790	0.0262	-6.989	-6.890	0.0249	-6.995	-6.691
k-l	0.0314	-26.898	-17.404	0.0314	-26.598	-17.404	0.0419	-26.983	-17.567
e-f	0.0314	26.898	17.404	0.0314	26.598	17.404	0.0419	26.983	17.567

4.3. CFD-DEM coupling analysis example

The CFD-DEM and CFD-DEM-M methods were adopted to numerically simulate the coupling of multiple particles and fluid as described in section 4.2. The calculation fluid domain was a 400 $\text{mm} \times 400$ $\text{mm} \times 400$ mm cube. The fluid density $\rho=1\text{g/cm}^3$, viscosity $\mu_f = 0.1\text{g/(cm}\times\text{s)}$, and initial state of the fluid was static. The computational domain was discretized using regular rectangular grids. The upper boundary was set as an open boundary, whereas the other walls adopted non-slip boundary conditions. The

geometric parameters, material, initial position, and calculation parameters of the particles were exactly the same as those in section 4.2. The CFD-DEM fluid calculation time step was set to 10 times the particle calculation time step. The initial time step of the CFD-DEM-M particle calculation was equal to the initial time step of the fluid calculation. The real calculation time step of the particles and fluid in the coupling calculation process was adjusted and corrected in real time based on the coupling convergence condition. The time step was set as shown in Table 4.

Table 4. Time step setting of particle - fluid coupling analysis

Calculation time step /s	CFD-DEM			Calculate the initial time step /s	CFD-DEM-M		
	A1	A2	A3		B1	B2	B3
fluid	10^{-5}	10^{-6}	10^{-7}	fluid	10^{-5}	10^{-6}	10^{-7}
particle	10^{-6}	10^{-7}	10^{-8}	particle	10^{-5}	10^{-6}	10^{-7}

The results of the CFD-DEM calculation show that the number of particle collisions obtained in the calculation time steps A1, A2, and A3 were 31, 36, and 36, respectively. The analysis showed that the A1 working condition was unreliable because of the deceleration of particles caused by fluid resistance, which resulted in five missed particle collisions and a reduction in the number of collisions by five times compared with the example in section 4.2. The calculation times for each working condition were 27, 53, and 109 min, respectively. Only when the particle calculation time step was less than or equal to 10^{-7} s, an accurate value could be obtained using the CFD-DEM coupling calculation.

The CFD-DEM-M calculation results show that, the number of particle collisions obtained by the calculation time steps B1, B2, and B3 were all 36, and the calculation times were 42.5, 47 min and 67.2 min, respectively. The change in the initial time step of particle calculation did not affect the accuracy of the CFD-DEM-M coupling calculation.

The time step of the fluid calculation A2 was one order of magnitude larger than the initial time step of the fluid calculation B3. The fluid solver started 10^5 times, the particle solver started 106 times, and the fluid and particle solvers both started 1001783 times. The particle solver A2 started 1783 times more than the particle solver B3 because the DEM-M algorithm corrected the initial time step according to the coupling convergence condition, and the real calculation time step was smaller than the initial time step with B3 taking 14.2 min more than the A2 calculation. The initial time step of the fluid calculation B2 was equal to that of the fluid calculation time step A2, whereas the initial time step of the particle calculation

B2 was one order of magnitude larger than that of the particle calculation A2. The average number of B2 startups of fluid and particle solvers were 105329. The number of solver startups was 5329 more than that of B2 and A2 fluid solvers. Moreover, the number of B3 multi-starts of the solver increased from 1783 to 5329 compared with A2 owing to the expansion of the initial time step of particle calculation and the non-convergence of the particle solution. B2 took 6 min less than the calculation time of A2 and the calculation time was reduced by 11.3 %. The initial time step of the fluid calculation B1 was one order of magnitude larger than that of the fluid calculation A2. In addition, the initial time step of the particle calculation was two orders of magnitude larger than that of the particle calculation A2. Both B1 fluid and particle solvers started 13254 times, which was less than the A2 calculation time of 10.5 minutes, reducing the calculation time by 19.8 %. The A2 and B1 motion trajectories of particles, a, c, e, g, i, and k, were basically consistent, as shown in Fig. 13.

In summary, the CFD-DEM-M method was used to numerically simulate the coupling of multiple particles and fluid. The initial time steps of fluid and particle calculation were 10^{-5} s, 10^{-6} s, and 10^{-7} s, which could obtain accurate results, and the calculation time was 19.8 % lower than that of the CFD-DEM method.

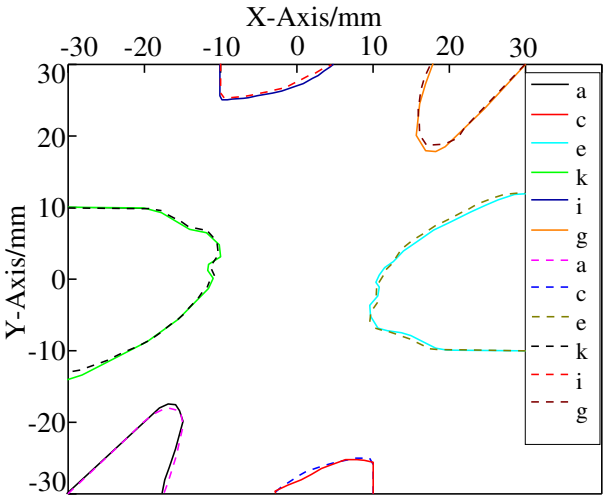

Fig. 13. Multiple particle trajectories in fluid

5 Conclusions

In the CFD-DEM coupling calculation used in the study, when the selection of the particle calculation time step was too small, the calculation efficiency was low. On the other hand, when the time step selection was too large, collision particles were missed. Through algorithm research and example verification, the following conclusions were drawn:

An improved particle collision search algorithm was proposed in the study, where the target particle was taken as the center particle, and the displacement between the center particle and the fastest moving particle in the calculation domain was considered the radius to construct the search ball. The particles that might collide were screened to establish the search list, which reduced the particle collision search time. The forward search method was used to judge the particle collision, avoiding the error caused by the large time step selection in the traditional DEM and ensuring an accurate description of the particle collision. The motion and collision results of two-particle and multi-particle simulations show that the particle calculation time step selection of this method has little effect on the particle collision calculation accuracy. The calculation time of the numerical simulation of two particles under uniform and variable speed conditions was reduced by 17.02 % and 18.54 %, respectively, and the calculation time of the multi-particle numerical simulation was reduced by 18.99 % compared with the traditional DEM method.

A coupling algorithm for automatically adjusting the time step of the particle and fluid calculation was established. The selection of the initial time step for particle and fluid calculations can be identical. The fluid calculation time can be adjusted in real-time according to the particle propulsion time. The automatic adjustment of the fluid calculation time step with particle calculation time was realized, and the calculation efficiency of particle and fluid coupling was improved. The numerical simulation of the multi-particle and fluid coupling example shows that the calculation accuracy of the algorithm is less affected by the initial calculation time step of particles and fluid, and the calculation efficiency can be improved by setting a large particle calculation time step. The calculation time of the algorithm was reduced by 19.8 % compared with the traditional CFD-DEM method.

CRediT authorship contribution statement

Xuefei Wang: Methodology, Software, Visualization, Writing of the original draft. Suling Wang: Methodology, Funding acquisition, Supervision. Ming Wang: Data curation, formal analysis. Xuemei Li: Data curation, language check. Lin Chi, Dandan Sun: Software, results validation.

Declaration of competing interests

The authors declare that they have no known competing financial interests or personal relationships that could have influenced the work reported in this paper.

Acknowledgments

We gratefully acknowledge the financial support from the National Natural Science Foundation of China (11972115); Supported by the China Postdoctoral Science Foundation(2019M661249); Funded by Heilongjiang Postdoctoral Fund Project(LBH-Z19123);

References

[1] L. Qiu, C. Yu, A hybrid DEM/CFD approach for solid-liquid flows, J. Hydrodyn. 26 (2014) 19-25.
[2] Z. Wang, Y. Teng, M. Liu. A semi-resolved CFD-DEM approach for particulate flows with kernel based approximation and Hilbert curve based searching strategy, J. Comput. Phys. 384 (2019) 151-169.
[3] C. Kloss, C. Goniva, A. Hager, Models, algorithms and validation for opensource DEM and CFD-DEM, Prog. Comput Fluid Dy. 12 (2012) 140-152.
[4] M. Lungu, J. Siam, L Mukosha, Comparison of CFD-DEM and TFM approaches for the simulation of the small scale challenge problem 1, Powder Technol. 378 (2020) 85-103.
[5] Alobaid, Falah. A particle–grid method for Euler–Lagrange approach. Powder Technology 286 (2015) 342-360.
[6] P. H. Zhu, Y. Z. Zhou, Y. R. Yang, et al. Discrete particle simulation of particulate systems: Theoretical developments[J]. Chem. Eng. Sci. 63 (2007) 5728-5770.
[7] F. Durst, D. Miloievic, B. Schnung, Eulerian and Lagrangian predictions of particulate two-phase flows: a numerical study. Appl. Math. Model. 8 (1984) 101-115.
[8] L. M. Yao, Z, M, Xiao, J, B, Liu,. An optimized CFD-DEM method for fluid-particle coupling dynamics analysis[J]. Int. J. Mech. Sci. 174 (2020) 105503.
[9] R. W. Hockney, J. W. Eastwood, Computer Simulation Using Particles, MrGrawHill, New York: 1981.

[10] M. P. Allen, D, J. Tildesley, Computer Simulation of Liquids, Clarendon Press, Oxford, 1987.
[11] J. D. Cohen, M. C. Lin, D. Manocha. I-collide: An interactive and exact collision detection system for large-scale environments, symposium on interactive 3d graphics. ACM, 1995 189-196.
[12] D.Baraff, Interactive simulation of solid rigid bodies, IEEE comput. graphics appl. 15 (1995) 63-75.
[13] M. C. Lin, Efficient Collision detection for animation and robotics, Ph. D Dissertation, University of California at Berkeley, 1993.
[14] B. C. Vemuri, L. Chen, L. V. Quoc, X. Zhang. Efficient and accurate collision detection for granular flow simulation, Graph. Mod. Image Proc. 60 (1998) 403-422.
[15] A. Schinner, Fast algorithms for the simulation of polygonal particles, Granul. Matter 2 (1999) 35-43.
[16] G. Erfan, Nezami,M. A. Youssef, Hashash, A fast contact detection algorithm for 3-D discrete element method - ScienceDirect. Comput. Geotech. 31 (2004) 575-587.
[17] X. L. Liu, J. V. Lemos, Procedure for contact detection in discrete element analysis, Adv. Eng. Softw. 32 (2001) 409-415.
[18] H. Mio, A. Shimosaka, Y. Shirakawa, J. Hidaka, Cell optimization for fast contact detection in the discrete element method algorithm, Adv. Powder Technol. 18 (2007) 441-453.
[19] E. G. Nezami, M. A. Youssef, D. Zhao J. Ghaboussi, A fast contact detection algorithm for 3-D discrete element method. Comput. Geotech. 31 (2004) 575-587.
[20] A. Munjiza, K. R. F. Andrews, NBS contact detection algorithm for bodies of similar size, Int. J. Numer. Meth. Eng. 43 (1998) 131-149.
[21] M. Chen, H. Liu, G. Zheng, Direct simulation algorithm for particle collision, Comp. Phys. 21 (2004) 421-426.
[22] P. Pischke, R. Kneer, D. P. Schmidt. A comparative validation of concepts for collision algorithms for stochastic particle tracking, Comput. Fluids 113 (2015) 77-86.
[23] F. Wang, X. Huang, F. Wang, Accurate collision detection fast algorithm for particle agglomeration process, Journal of Zhejiang University 53 (2019) 1148-1156.
[24] K. Sharma; S. S. Mallick, A. Mittal, A study of energy loss due to particle to particle and wall collisions during fluidized dense-phase pneumatic transport, Powder Technol. 362 (2019) 707-716.
[25] S. Yang, K. Luo, M. Fang, K. Zhang, J Fan, Parallel CFD–DEM modeling of the hydrodynamics in a lab-scale double slot-rectangular spouted bed with a partition plate, Chem. Eng. J. 236 (2014) 158-170.
[26] M. Banaei, J. Jegers, S. Van, J. Kuipers, NG. Deen, Tracking of particles using TFM in gas-solid fluidized beds, Adv. Powder Technol. 10 (2018) 88–99.
[27] J. J. Nieuwland, M. Annaland, JAM Kuipers, S. WPM, Hydrodynamic modeling of gas/particle flows in riser reactors, AICHE J. 42 (1996) 1569-1582.
[28] Z. Y. Zhou, S. B. Kuang, K. W. Chu, Discrete particle simulation of particle–fluid flow: model formulations and their applicability, J. Fluid Mech. 661 (2010) 482-510.
[29] M. A. Habib, H. M. Badr, R. Ben-Mansour , M. Kabir, Erosion rate correlations of a pipe protruded in an abrupt pipe contraction, Int. J. Impact Eng. 34 (2007) 1350-1369.
[30] Y. Zhang, Application and improvement of computational fluid dynamics (CFD) in solid particle erosion modeling, The University of Tulsa (2006).
[31] M. Loewenberg, The unsteady Stokes resistance of arbitrarily oriented, finite-length cylinders, Phys. Fluids 5 (1993) 3004-3006.
[32] D. Eskin, J. Ratulowski, K. Akbarzadeh, Modeling of particle deposition in a vertical turbulent pipe flow at a reduced probability of particle sticking to the wall, Chem. Eng. Sci. 66 (2011) 4561-4572.
[33] R. J. Liu, Xiao R, M. Ye, Z. Liu, Analysis of particle rotation in fluidized bed by use of discrete particle model, Adv. Powder Technol. (2018) S0921883118301304.
[34] P. J. Thomas, On the influence of the Basset history force on the motion of a particle through a fluid, Phys. Fluids 4 (1992) 2090-2093.
[35] R. D. Felice, The voidage function for fluid-particle interaction systems, Int. J. Multiphas. Flow 20 (1994) 153-159.
[36] J. B. Liu, M. Wang, X. F. Wang, L. M. Yao, M. Yang, Q. B. Yue, Research on propulsion algorithms of particle swarm collision search and cfd-dem coupling domain solving, Chin. J. Theor. Appl. Mech. 15 (2021) 1569-1585.