A coupled polygonal DEM-LBM technique based on an immersed boundary method and energy-conserving contact algorithm

Min Wang[a], Y.T. Feng[b,*], T.M. Qu[b], T.T. Zhao[c]

[a] *Fluid Dynamics and Solid Mechanics Group, Theoretical Division, Los Alamos National Laboratory, Los Alamos, NM 87545, USA*

[b] *Zienkiewicz Centre for Computational Engineering, College of Engineering, Swansea University, Swansea, Wales SA1 8EP, UK*

[c] *Institute of applied mechanics and biomedical engineering, Taiyuan University of Technology, Taiyuan 030024, Shanxi, China*

Abstract

This work presents a framework of coupling polygonal discrete elements and the lattice Boltzmann method using a direct forcing immersed boundary scheme. In this technique, an energy-conserving contact algorithm is utilized to handle the interactions between convex and concave polygonal particles. The surface of a polygon is represented by discrete boundary points which includes vertices of polygonal particles and/or points interpolated from vertices. The fluid-particle coupling is obtained through the interactions of the boundary points and the imaginary fluid particles using a direct-forcing immersed boundary method. Validations of the proposed technique are made by single particle and multiple arbitrarily-shaped particle sedimentation tests, and the effect of particle shape is illustrated using a drafting-kissing-tumbling benchmark.

Keywords: concave polygonal particle, discrete element method, lattice Boltzmann method, immersed boundary method, multiphase flow

1. Introduction

The coupled discrete element method and lattice Boltzmann method (DEMLBM) has been attracting more and more attention in various disciplines because of its simpler implementation and higher computational efficiency (Cook et al., 2004; Feng et al., 2007) than the coupled discrete element method and computational fluid dynamics (DEMCFD). Applications

*Corresponding author: y.feng@swansea.ac.uk

of DEMLBM can be found in geotechnical engineering, chemical engineering, multiphase fluid flows and many other fields. The combination of DEM and LBM can be achieved by various coupling schemes, such as the modified bounce-back (MBB) method (Ladd, 1994), momentum-interpolation (MI) method (Mei et al., 1999), immersed moving boundary (IMB) scheme (Noble and Torczynski, 1998) and immersed boundary method (IBM) (Feng and Michaelides, 2004). Among them, the IMB and IBM are two most popular approaches.

The IMB method fully takes advantage of the particle nature of LBM by introducing an additional collision operator to the lattice Boltzmann equation, which accounts for the contribution of solid particles to the lattice nodes partially covered by the particles. Based on the momentum conservation, the hydrodynamic forces applied to a solid particle can be directly determined from this additional collision term. Meanwhile, to smoothly represent the boundary profile of a solid particle a weighting function associated with a nodal solid area was introduced. Two key steps of the IMB scheme are to fast identify fluid and solid boundary nodes and calculate the corresponding weighting functions associated with each nodal solid area. These steps mainly govern the computational efficiency and accuracy of IMB. Different approximate methods of calculating the nodal solid area, such as cell decomposition and polygon approximation, were discussed by Owen et al. (2011). Recently, searching algorithms for efficiently identifying boundary nodes were reported by Wang et al. (2017, 2018b), and a fast computation method of a nodal solid area was proposed by Jones and Williams (2017). To eliminate the relaxation time dependency of hydrodynamic forces, the two-relaxation-time model was implemented into DEM-LBM and a modified weighting function for IMB was proposed by Wang et al. (2018a). A more efficient searching algorithm and a Gaussian integration for calculating a nodal solid area were developed to improve the computing efficiency in Wang et al. (2019) and used by Zhao et al. (2019). Instability of IMB and its treatments for particulate simulations involving multi-covered nodes were reported in a recent work(Wang et al., 2020b).

The IBM was first proposed for CFD by Peskin (1977). It was introduced to LBM by Feng and Michaelides (2004). The basic idea of the initial IBM is to treat the particle boundary

as a series of Lagrangian points with high stiffness. The desired position of a boundary point can be determined from translation and rotation of the solid particle. The actual position of a boundary node is updated by the fluid velocity, interpolated from surrounding fluid nodes, at this boundary point. The hydrodynamic force can be obtained from the derivation between the desired position of the boundary point and the current position using a spring model with high stiffness. This force is dependent on the value of the specified spring stiffness. Then, a direct forcing IBM proposed by Uhlmann (2005) based on momentum exchange was introduced into LBM by Niu et al. (2006). Different from the calculation of the hydrodynamic force in the initial IBM, the derivation of the desired velocity of the boundary point, which can be determined from particle movement, and a virtual velocity of the boundary point interpolated from the surrounding fluid nodes was used to determine the fluid-particle interaction from momentum conservation. Meanwhile, the velocity difference is distributed to the surrounding fluid nodes.

It is reported, however, that the non-slip boundary condition is not fully enforced in the direct forcing IBM due to the one-time momentum exchange (Wu and Shu, 2009). Then, an IB-LBM using an implicit force density formulation was developed in Wu and Shu (2009) where an unknown velocity correction is prescribed. This implicit scheme can enforce the non-slip boundary condition at the fluid-solid interface but requires complex matrix inversions and a higher computational memory cost. To improve the accuracy of the direct forcing IBM, a multi-direct forcing IBM proposed by Wang et al. (2008) was adopted in LBM by Hayashi et al. (2012). A fixed momentum exchange iteration was used to approximately satisfy the non-slip condition. Later, Dash et al. (2014) proposed an implicit flexible forcing IBM based on the multi-direct forcing IBM. Instead of using a fixed iteration number in the multi-direct forcing IBM, a flexible sub-iteration for the velocity correction is terminated when the convergence criterion is satisfied. It was found that the implementation of IBM is simpler than IMB, but some acceptable oscillation of hydrodynamic forces can be observed in IBM compared to IMB (Wang et al., 2020a).

To the best of our knowledge, most simulations of particle-fluid systems by DEMLBM

adopt disks and spheres for two and three dimensional problems respectively. However, the angularity of irregular particles plays an important role in particle-fluid systems, and disks or spheres are unable to capture some physical behaviour such as the sufficient resistance to rolling motion. Only a few references accounted for irregular particle shapes in the DEMLBM simulations coupled by IMB (Han et al., 2007; Galindo-Torres, 2013; Li et al., 2019). To handle the contact interaction between irregular particles, a simplified spheropolyhedra method (Galindo-Torres, 2013) and a convex decomposition method (Li et al., 2019) were respectively utilized. As mentioned before, the fluid boundary nodes of each moving particle should be identified every time step in IMB. Although an efficient boundary point searching algorithm was proposed in our previous work (Wang et al., 2019), and can be extended to polygons, this search has to be applied at each time step, and therefore can be very costly. In contrast, using IBM the boundary points of a particle is exclusively determined before the simulation and they can be explicitly updated from particle translation and rotation at each time step. Thus no search is involved any more.

Given the straightforward implementation of IBM and a lesser computational cost than IMB in the treatment of polygons or polyhedra, this work develops a coupled polygonal DEM-LBM-IBM framework, in which the vertices or boundary points interpolated from vertices will be directly used for momentum exchange and subsequent calculation of hydrodynamic forces. Meanwhile, an advanced energy-conserving contact algorithm proposed in the references (Feng and Owen, 2004; Feng et al., 2012; Feng, 2021a,b) is implemented to handle the contact interaction between polygonal particles including concave shapes.

2. Methodologies

In this section, we will focus on the contact interaction between polygonal particles, the direct forcing IBM and their coupling. Details of the fundamentals of DEM and LBM can be found in the literature.

2.1. Energy-conserving contact algorithm for polygons

Most existing contact algorithms for polygons are using a heuristic or simplified way to deal with the contact between two polygons. Basic issues of polygon-polygon contact are mainly how to determine contact point, unit normal and tangential vectors, and accurately calculate the contact force and torque. A robust energy-conserving contact model for arbitrarily shaped particles was proposed in (Feng and Owen, 2004; Feng et al., 2012), from which the detailed derivation process of this energy-conserving contact model can be found. The rest of this part will highlight the important issues mentioned before.

Based on the energy conservation principle, for two polygons in contact, the normal contact force $\boldsymbol{F}_n$ is defined as

$$\boldsymbol{F}_n = -\frac{\partial \phi(\boldsymbol{x}, \theta)}{\partial \boldsymbol{x}} \tag{1}$$

where $\phi(\boldsymbol{x}, \theta)$ is a contact energy/potential function; and $\boldsymbol{x}$ and θ are the centroid coordinates and rotation angle of the first polygons in contact. $\boldsymbol{F}_n$ is the force exerted from the second polygon to the first one.

The corresponding contact moment $\boldsymbol{M}$ is defined as

$$\boldsymbol{M} = -\frac{\partial \phi(\boldsymbol{x}, \theta)}{\partial \theta} \tag{2}$$

Take two polygonal particles in contact (the first case shown in Figure 1) for instance. Let $p(x_p, y_p)$ and $q(x_q, y_q)$ be the penetrating vertices of particle 1 and particle 2 respectively, and $g(x_g, y_g)$ and $h(x_h, y_h)$ be the two intersections. The contact energy function ϕ can be chosen as a monotonically increasing function of the overlap area (A) of the two polygons. Then, the normal contact force can be expressed as

$$\boldsymbol{F}_n = -\frac{\partial \phi(A)}{\partial \boldsymbol{x}_p} = -\frac{\partial \phi(A)}{\partial A} \cdot \frac{\partial A}{\partial \boldsymbol{x}_p} = F_n \boldsymbol{n} \tag{3}$$

where F_n and $\boldsymbol{n}$ are the magnitude of normal contact force and normal direction to be defined

below. The term $\frac{\partial A}{\partial \boldsymbol{x}_p}$ is given by

$$\frac{\partial A}{\partial \boldsymbol{x}_p} = \{(y_h - y_g), -(x_h - x_g)\} \tag{4}$$

Thus, $d = \|\frac{\partial A}{\partial \boldsymbol{x}_p}\|$ is the contact width or distance between the two intersections g and h.

By choosing a special form of the function $\phi(A)$, the magnitude of the normal contact force F_n in the form of a Hertz type can be defined as

$$F_n = k_n A^{\frac{1}{2}} d \tag{5}$$

where k_n is the normal stiffness.

The unit normal vector $\boldsymbol{n}$ and the tangential vector $\boldsymbol{t}$ are given by

$$\boldsymbol{n} = -\frac{\frac{\partial A}{\partial \boldsymbol{x}_p}}{\|\frac{\partial A}{\partial \boldsymbol{x}_p}\|} = \frac{\{-(y_h - y_g), (x_h - x_g)\}}{d} \tag{6}$$

$$\boldsymbol{t} = \frac{\{(x_h - x_g), (y_h - y_g)\}}{d} \tag{7}$$

In this energy-conserving contact algorithm, the middle point m of the contact line gh is proved to be the preferred choice as the contact point. The general normal contact model also works for node-edge and edge-edge contact situations as special cases shown in Figure 1. Treatments of other complex contact cases can be found in the reference (Feng et al., 2012; Feng, 2021b).

Besides, Based on the classic Coulomb friction model, the friction force is updated by adding the incremental tangential contact force $\Delta \boldsymbol{F}_t$

$$\boldsymbol{F}_t = \boldsymbol{F}_{t-\Delta t} + \boldsymbol{\Delta} F_t; \qquad \Delta \boldsymbol{F}_t = -k_t \Delta \boldsymbol{S}, \qquad |\boldsymbol{F}_t| \leq |\boldsymbol{F}_{t,max}| = \mu |\boldsymbol{F}_n| \tag{8}$$

where k_t is the tangential stiffness, $\Delta \boldsymbol{S}$ is the incremental shearing displacement at the current

time step, and μ is the coefficient of friction.

2.2. LBM-IBM

The lattice Boltzmann method is a mesoscopic particle-based CFD method where imaginary particles are introduced and treated as clusters of fluid molecules. The flow behavior of fluid is reproduced by collision and streaming processes of these imaginary particles. Collision and streaming processes can be described by the lattice Boltzmann equation below derived from statistical mechanics

$$\boldsymbol{f}_i(\boldsymbol{x} + \boldsymbol{e}_i\Delta t, t + \Delta t) - \boldsymbol{f}_i(\boldsymbol{x}, t) = \boldsymbol{\Omega}_i + \boldsymbol{F}_i\Delta t \tag{9}$$

where $\boldsymbol{f}_i$ is the fluid density distribution function in the i-direction; $\boldsymbol{x}$ and $\boldsymbol{e}$ are the coordinates of the lattice node under consideration and the velocity vectors of imaginary fluid particles; t, $\boldsymbol{\Omega}_i$ and $\boldsymbol{F}_i$ are, respectively, the current time, collision operator and body force term.

There are mainly two collision operators and they are known as the Bhatnagar-Gross-Krook (BGK) (Qian et al., 1992) and multiple relaxation time (MRT) (d'Humieres, 2002) collision operators. In this work, the BGK model, D2Q9, is used and the corresponding collision operator, $\boldsymbol{\Omega}_i$, is given by

$$\boldsymbol{\Omega}_i = -\frac{\Delta t}{\tau}(\boldsymbol{f}_i(\boldsymbol{x}, t) - \boldsymbol{f}_i^{eq}(\boldsymbol{x}, t)) \tag{10}$$

where $\tau > 0.5$ is a relaxation parameter; $\boldsymbol{f}_i^{eq}$ is the equilibrium distribution function in the i-direction and defined as

$$f_i^{eq}(\boldsymbol{x}, t) = \omega_i\rho(1 + \frac{3}{C^2}\boldsymbol{e}_i \cdot \boldsymbol{u} + \frac{9}{2C^4}(\boldsymbol{e}_i \cdot \boldsymbol{u})^2 - \frac{3}{2C^2}\boldsymbol{u} \cdot \boldsymbol{u}) \tag{11}$$

in which ω_i is the weighting factor of the fluid particle in the i-direction; ρ and $\boldsymbol{u}$ are the macroscopic density and velocity of the lattice node; C is the lattice speed defined by

$$C = \frac{h}{\Delta t} \tag{12}$$

with h as the lattice size.

The body foce term $\boldsymbol{F}_i$ in Equation (9) is defined by

$$\boldsymbol{F}_i = \omega_i(1 - \frac{1}{2\tau})(\frac{\boldsymbol{e}_i - \boldsymbol{u}}{C_s^2} + \frac{\boldsymbol{e}_i \cdot \boldsymbol{u}}{C_s^4}) \cdot \boldsymbol{f} \tag{13}$$

where $C_s = C/\sqrt{3}$, $\boldsymbol{f}$ is the body force (vector) density and needs to be determined from IBM.

In the direct forcing IBM, the solid boundary is discretized into several Lagrangian boundary points (see Figure 2)). At each time step, the (desired) velocity at a boundary points of a particle can be determined from the rigid body motion of the particle. Meanwhile, the fluid velocity at the boundary point, with position $\boldsymbol{X}_b = (X_b, Y_b)$, can be interpolated from the four surrounding lattice nodes, with position $\boldsymbol{x}_{ij} = (x_{ij}, y_{ij})$, using the Kernel distribution function, $D(\boldsymbol{x}_{ij} - \boldsymbol{X}_b)$, definded below.

$$D(\boldsymbol{x}_{ij} - \boldsymbol{X}_b) = \frac{1}{h^2}\delta(\frac{x_{ij} - X_b}{h})\delta(\frac{y_{ij} - Y_b}{h}) \tag{14}$$

where $\delta()$ is the discrete delta function(Dash et al., 2014).

Due to the non-slip boundary condition at the fluid-solid interface, the interpolated fluid velocity needs to be equal to the desired boundary velocity. To achieve this non-slip condition, the velocity difference, $\Delta\boldsymbol{U}_b$, between the interpolated and desired velocities will be distributed to its surrounding fluid nodes using the following equation

$$\boldsymbol{u}(\boldsymbol{x}_{ij}) = \boldsymbol{u}(\boldsymbol{x}_{ij}) + \Delta\boldsymbol{u}(\boldsymbol{x}_{ij}) = \boldsymbol{u}(\boldsymbol{x}_{ij}) + \sum_k \Delta\boldsymbol{U}_b D(\boldsymbol{x}_{ij}) s_k \tag{15}$$

where $\boldsymbol{x}_{ij}$ and $\boldsymbol{u}(\boldsymbol{x}_{ij})$ are the coordinates and velocity of the lattice node around a boundary point; k is the serial number of boundary points associated with the fluid lattice nodes and s_k is the arc length between two consecutive boundary points.

The force density of the fluid lattice node adjacent to a particle is updated by

$$\boldsymbol{f} = \boldsymbol{f} + \frac{2\rho\Delta\boldsymbol{u}(\boldsymbol{x}_{ij})}{\Delta t} \tag{16}$$

The hydrodynamic force, $\boldsymbol{F}_b$, applied to the boundary point can be computed as

$$\boldsymbol{F}_b = \frac{2\rho\Delta\boldsymbol{U}_b}{\Delta t} \tag{17}$$

Then, the resultant hydrodynamic force and torque applied to the solid particle can be obtained from its boundary points.

2.3. Coupling procedures

In the simulation of fluid-particle systems, the timestep of LBM is normally greater than that of DEM. A DEM subcycling algorithm was proposed within an LBM timestep in (Feng et al., 2007). The specific DEMLBM coupling procedure is given in Figure 3. This framework works for the polygonal DEM-LBM technique coupled by any IBM scheme which is employed in Step S9 in Figure 3. For illustration, we only adopt the direct forcing IBM in this work. It should be highlighted that the boundary points of polygonal particles are predefined once before the time integration of LBMDEM.

3. Numerical experiments and validations

3.1. Single particle sedimentation

In this section, we will first validate the accuracy of our in-house code BPLBM. The single particle sedimentation is one of the benchmark tests for the validation of disk/sphere-fluid coupling, but there is no such benchmark for the coupling between polygon/polyhedron and fluid. In the two-dimensional case, an indirect way is to approximate the disk using a multi-vertex polygon.

As shown in Figure 4, a solid polygonal particle with 16 vertices, which is used to mimic a disk with diameter 0.25 cm, is placed at the position (1 cm, 4 cm) in a fluid-filled tube with 2 cm width and 6 cm height. The density of the solid particle is 1.25 g/cm^3. The density and kinematic viscosity of the fluid are 1.0 g/cm^3 and 10^{-5} m^2/s respectively. In this model, the relaxation parameter (τ) is 0.65. To ensure an accurate simulation, the lattice spacing ($h = 0.01$) is selected so that the ratio of the particle diameter to the lattice spacing is greater

than 20 which is sufficient for an accurate DEMLBM simulation (Feng and Michaelides, 2004; Niu et al., 2006; Wang et al., 2020b). The contact stiffness between the solid particle and the walls is 10^7 N/m. Due to the applied gravity force, the particle will fall through the fluid and finally collide with the bottom wall and settle.

Figure 4 shows the snapshots of this particle sedimentation process at different time instants. The velocity shown in the legend is in the dimensionless value of the lattice system. To validate the accuracy of the polygonal DEMLBM technique coupled by the direct forcing IBM, another numerical model with the same parameters except for using a circular particle is carried out. Both simulations are compared with the result obtained by the direct numerical simulation (DNS) of a disk sedimentation using CFD.

Figure 5 compares the vertical movement of the solid particle. It can be seen that the movement of the 16 sided polygon matches the simulation of the disk using DEMLBM coupled by IBM, and both of them are consistent with the DNS by CFD. To investigate the effect of particle shape, Figure 6 compares the variation of particle rotation during the sedimentation process. In the single-particle case, the rotation of the disk and polygon is opposite, but the rotation magnitude of both is very limited, which is the primary reason that there is nearly no difference in the movement of the disk and polygon. To further validate the DEMLBM-IBM technique, velocities of particles and drag forces are compared in Figures 7 and 8, respectively. An acceptable difference between DEMLBM and DNS simulations can be observed. Again, the vertical velocity and drag force of the disk and polygon are in very good agreement.

3.2. Drafting-kissing-tumbling (DKT) phenomenon

To further validate the polygonal DEMLBM technique coupled by IBM, the DKT test of two polygonal particles with 8 sharp vertices will be used for illustration and the investigation of the effect of particle shape on particle-fluid coupling.

The numerical model with a rectangular domain of 2 cm $\times$ 8 cm is filled with water. Two particles with a density of 1.01 g/cm^3 are placed, respectively, at positions (1.0 cm, 7.2 cm) and (1.0 cm, 6.8 cm). The fluid density and kinematic viscosity are 1.0 g/cm^3 and 10^{-6} m^2/s respectively. The relaxation time and lattice spacing are selected as 0.65 and 0.01 cm. Two

simulations are performed. The one simulates the DKT phenomenon of two disks of 0.2-cm diameter with 16 boundary points. For comparison, the other simulates the sedimentation process of two 8-vertex polygons. The stiffness of the solid particles is 10^7 N/m. To ensure an accurate simulate, 8 more boundary points interpolated from the vertices are used in the second simulation.

Figure 9 compares snapshots of DKT phenomenon in the two simulations at different time instants. The variations of particle positions in both horizontal and vertical directions are compared in Figure 10, where $P1$ and $P2$ represent particle 1 on the top of particle 2. At the beginning of the sedimentation process, $P1$ and $P2$ start falling with a close distance. $P2$ is in the front of movement and subject to a larger drag force than $P1$. Then the trailing particle (P1) moves gradually faster. This is the $drafting$ period. When $P1$ and $P2$ get closer, the wake of $P2$ could cause a weak lateral disturbance due to the slight particle rotation. $P1$ then slightly deviates its original lateral position. Subsequently, they will contact with each other. This period is the so-called $Kissing$ phase. Gradually, $P1$ will overtake $P2$ because of a larger instant vertical velocity. This is the $tumbling$ period. It is found that the DKT phenomenon can be observed from the two simulations, and it appears much earlier in the simulation with two polygons. This test case clearly demonstrates the effect of particle shape on the flow behavior of particles in the fluid-particle system.

Figure 11 compares the velocity variation of the two particles in the two simulations. The variation magnitude of velocity is of the same order. In contrast, the variations of particle rotation and angular velocity show apparent differences (see Figure 12). It indicates that when the two particles are (nearly) in contact with each other, the angular velocity of the polygons increases abruptly while the variation of the translational velocity is relatively small. When the interaction between the boundary points and the fluid happens, the angular velocity will cause a non-negligible velocity compared to the translational velocity. Therefore, there are obvious differences of hydrodynamic forces applied to the particles in the two simulations, which is confirmed by Figure 13 where both lift and drag forces are given.

3.3. Multiple-particle sedimentation

A multiphase flow case involving arbitrarily-shaped polygons with different vertices is also numerically tested to validate the robustness of the developed DEMLBM technique coupled by the direct forcing IBM. In this 2 $cm \times 8$ cm box filled with water, 18 convex and concave polygonal particles with vertices ranging from 4 to 10 are considered. The density of polygonal particles is 2.75 g/cm^3. Other parameters of LBM and DEM are the same as those in the above DKT model.

Snapshots of the velocity contour of the fluid and particle sedimentation process at different instants are given in Figure 14. The particles are all settled at the end. The whole sedimentation process of all the particles is successfully simulated, which demonstrates the robustness of the polygonal DEMLBM technique.

4. Conclusions

In this work, a coupled polygonal discrete element and lattice Boltzmann method is developed. The contact interaction between polygons is handled using an energy-conserving contact model. Compared to simplified treatments of polygons or polyhedra, this contact model has a rigorous theoretical foundation and a clear geometric perspective, which guarantees a robust contact treatment between any shaped particles. The coupling of fluid and particles is achieved by the immersed boundary method, where the predefined boundary points are used to represent the polygon profile. By applying the non-slip boundary conditions to the boundary points and the corresponding imaginary fluid points at the same positions, hydrodynamic forces can be directly determined from momentum conservation. Meanwhile, the fluid properties of lattice nodes surrounding a boundary point is updated by the Delta distribution function. From our preliminary investigation, there is a marginal difference between the direct forcing and multi-direct forcing immersed boundary methods. Hence, only the direct forcing IBM is considered in this paper.

The accuracy and robustness of the proposed framework is validated by both single-particle and multiple-particle sedimentation tests. Comparisons with the DNS by CFD show good

agreement. In addition, the drafting-kissing-tumbling phenomenon is investigated by two polygons and two disks, respectively. It is found the particle shape has a significant influence on the dispersed multiphase flow behaviour.

The proposed methodologies can be extended to arbitrarily shaped particles in 3D.

References

Cook, B.K., Noble, D.R., Williams, J.R., 2004. A direct simulation method for particle-fluid systems. Engineering Computations: Int J for Computer-Aided Engineering 21, 151–168.

Dash, S., Lee, T., Lim, T., Huang, H., 2014. A flexible forcing three dimension ib–lbm scheme for flow past stationary and moving spheres. Computers & Fluids 95, 159–170.

d'Humieres, D., 2002. Multiple–relaxation–time lattice boltzmann models in three dimensions. Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences 360, 437–451.

Feng, Y., 2021a. Energy-conserving contact interaction models for arbitrarily shaped discrete elements: Contact volume based model and computational issues. Computer Methods in Applied Mechanics and Engineering 373, 113493.

Feng, Y., 2021b. An energy-conserving contact theory for discrete element modelling of arbitrarily shaped particles: Basic framework and general contact model. Computer Methods in Applied Mechanics and Engineering 373, 113454.

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

Feng, Y., Han, K., Owen, D., 2012. Energy-conserving contact interaction models for arbitrarily shaped discrete elements. Computer methods in applied mechanics and engineering 205, 169–177.

Feng, Y., Owen, D., 2004. A 2d polygon/polygon contact model: algorithmic aspects. Engineering Computations .

Feng, Z.G., Michaelides, E.E., 2004. The immersed boundary-lattice boltzmann method for solving fluid–particles interaction problems. Journal of computational physics 195, 602–628.

Galindo-Torres, S., 2013. A coupled discrete element lattice boltzmann method for the simulation of fluid–solid interaction with particles of general shapes. Computer Methods in Applied Mechanics and Engineering 265, 107–119.

Han, K., Feng, Y., Owen, D., 2007. Numerical simulations of irregular particle transport in turbulent flows using coupled lbm-dem. Computer Modeling in Engineering and Sciences 18, 87.

Hayashi, K., Rojas, R., Seta, T., Tomiyama, A., 2012. Immersed boundary-lattice boltzmann method using two relaxation times. The Journal of Computational Multiphase Flows 4, 193–209.

Jones, B.D., Williams, J.R., 2017. Fast computation of accurate sphere-cube intersection volume. Engineering Computations 34, 1204–1216.

Ladd, A.J., 1994. Numerical simulations of particulate suspensions via a discretized boltzmann equation. part 1. theoretical foundation. Journal of fluid mechanics 271, 285–309.

Li, X., Wang, F., Zhang, D., Gu, S., Gao, X., 2019. Fluid-solid interaction simulation for particles and walls of arbitrary polygonal shapes with a coupled lbm-imb-dem method. Powder Technology 356, 177–192.

Mei, R., Luo, L.S., Shyy, W., 1999. An accurate curved boundary treatment in the lattice boltzmann method. Journal of computational physics 155, 307–330.

Niu, X., Shu, C., Chew, Y., Peng, Y., 2006. A momentum exchange-based immersed boundary-lattice boltzmann method for simulating incompressible viscous flows. Physics Letters A 354, 173–182.

Noble, D., Torczynski, J., 1998. A lattice-boltzmann method for partially saturated computational cells. International Journal of Modern Physics C 9, 1189–1201.

Owen, D., Leonardi, C., Feng, Y., 2011. An efficient framework for fluid–structure interaction using the lattice boltzmann method and immersed moving boundaries. International Journal for Numerical Methods in Engineering 87, 66–95.

Peskin, C.S., 1977. Numerical analysis of blood flow in the heart. Journal of computational physics 25, 220–252.

Qian, Y.H., d'Humières, D., Lallemand, P., 1992. Lattice bgk models for navier-stokes equation. EPL (Europhysics Letters) 17, 479.

Uhlmann, M., 2005. An immersed boundary method with direct forcing for the simulation of particulate flows. Journal of Computational Physics 209, 448–476.

Wang, D., Leonardi, C.R., Aminossadati, S.M., 2018a. Improved coupling of time integration and hydrodynamic interaction in particle suspensions using the lattice boltzmann and discrete element methods. Computers & Mathematics with Applications 75, 2593–2606.

Wang, M., Feng, Y., Owen, D., Qu, T., 2019. A novel algorithm of immersed moving boundary scheme for fluid–particle interactions in dem-lbm. Computer Methods in Applied Mechanics and Engineering 346, 109–125.

Wang, M., Feng, Y., Pande, G., Zhao, T., 2018b. A coupled 3-dimensional bonded discrete element and lattice boltzmann method for fluid-solid coupling in cohesive geomaterials. International Journal for Numerical and Analytical Methods in Geomechanics 42, 1405–1424.

Wang, M., Feng, Y., Qu, T., 2020a. On the implicit immersed boundary method in coupled discrete element and lattice boltzmann method. International Journal for Numerical and Analytical Methods in Geomechanics 44, 516–532.

Wang, M., Feng, Y., Qu, T., Tao, S., Zhao, T., 2020b. Instability and treatments of the coupled dem-lbm by the immersed moving boundary scheme. International Journal for Numerical Methods in Engineering In press.

Wang, M., Feng, Y., Wang, Y., Zhao, T., 2017. Periodic boundary conditions of discrete element method-lattice boltzmann method for fluid-particle coupling. Granular Matter 19, 43.

Wang, Z., Fan, J., Luo, K., 2008. Combined multi-direct forcing and immersed boundary method for simulating flows with moving particles. International Journal of Multiphase Flow 34, 283–302.

Wu, J., Shu, C., 2009. Implicit velocity correction-based immersed boundary-lattice boltzmann method and its applications. Journal of Computational Physics 228, 1963–1979.

Zhao, T., Feng, Y., Tan, Y., 2019. Characterising 3d spherical packings by principal component analysis. Engineering Computations 37, 1023–1041.

5. Figures

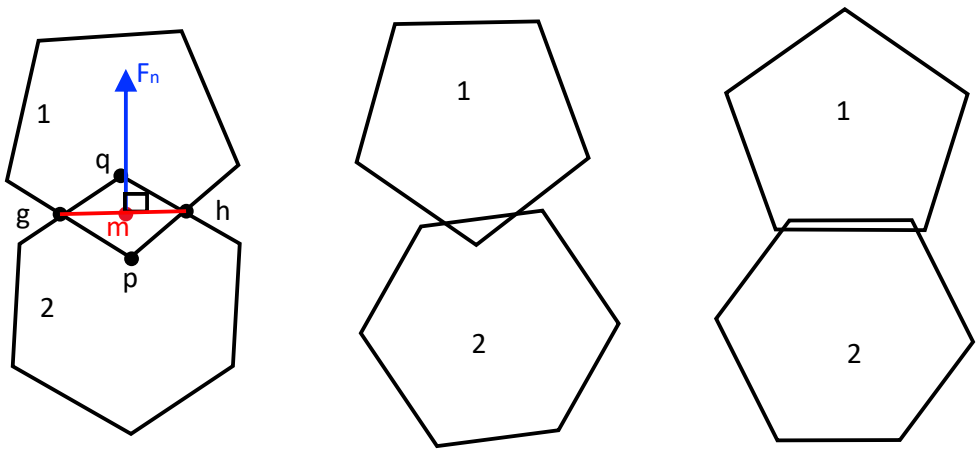


Figure 1: Two polygons in contact

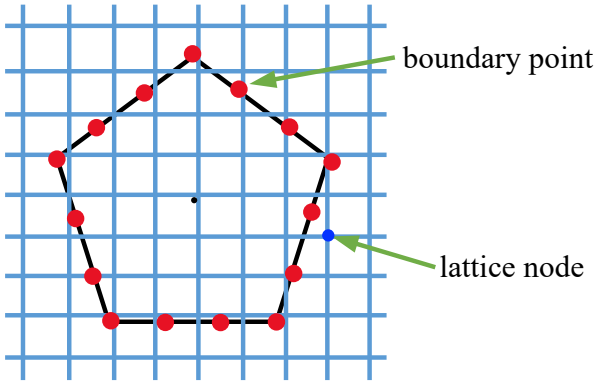


Figure 2: illustration of IBM scheme

S1: Read boundary points, including vertices, of polygons;
S2: Calculate centroid, area, mass, moment of inertia for particles;
Loop over LBM iterations:

Loop over DEM sub-cycling:

DEM

S3: *Update bounding box of particles;*
S4: *Perform global contact detection to build up potential contact list;*
Loop over contact pairs:
S5: *Check local contact and determine intersections;*
S6: *Calculate contact force and corresponding torque;*
S7: *Update particle velocity and angular velocity;*
S8: *Update centroid and rotation of the polygon, coordinates of boundary points;*

LBM

S9: *Approach fluid-particle interactions using IBM: update velocity, force density of lattice nodes associated to the solid particle, and calculate the hydrodynamic force using Eqs. 14-16;*
S10: *Fluid particles collide at each lattice node (so-called collision process);*
S11: *Apply bounce-back rule to lattice nodes occupied by stationary walls and particles;*
S12: *Propagate fluid particles to adjacent nodes (so-called streaming process);*
S13: *Apply LBM boundary conditions.*

Figure 3: Coupling procedure of DEMLBM-IBM

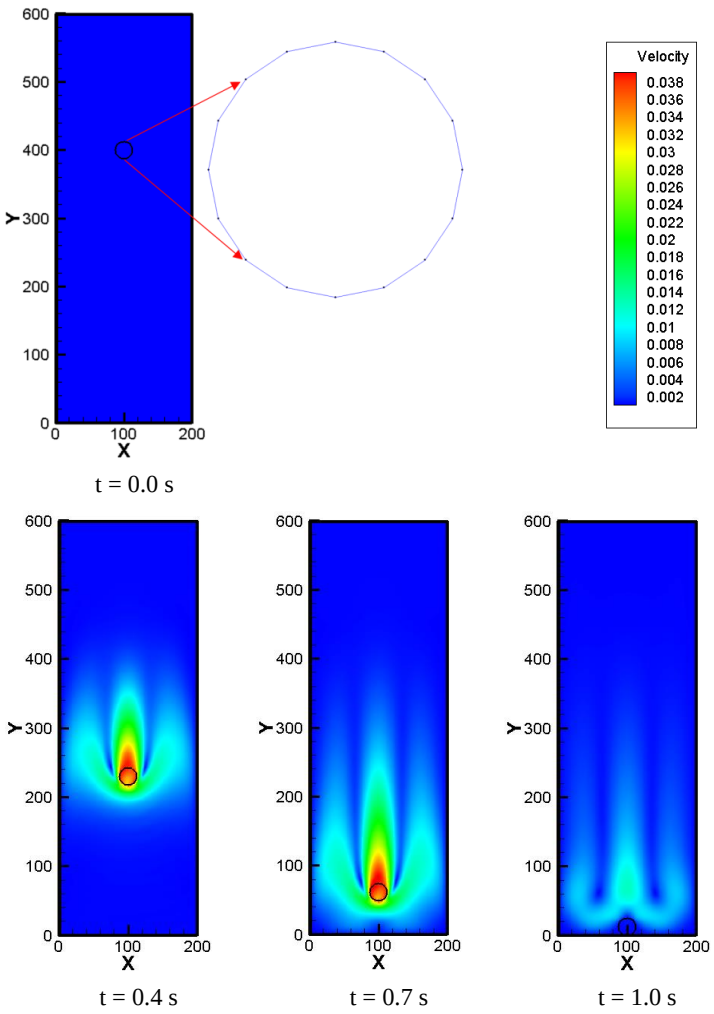


Figure 4: Snapshots of plogonal particle sedimentation

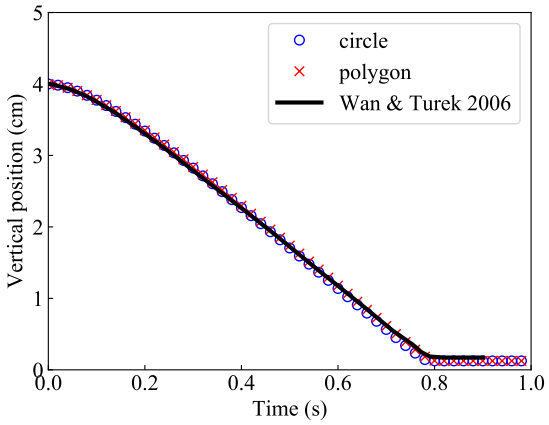


Figure 5: Comparison of vertical position with DNS

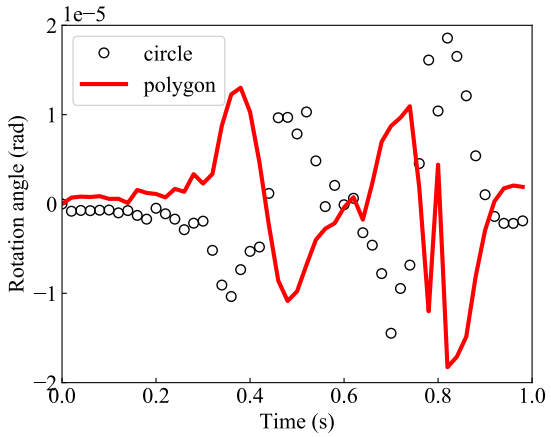


Figure 6: Comparison of particle rotation

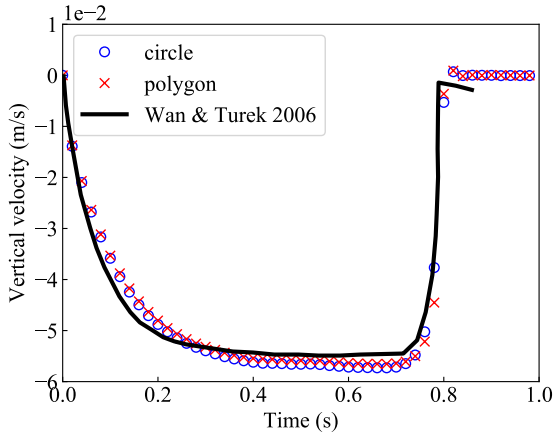


Figure 7: Comparison of vertical velocity with DNS

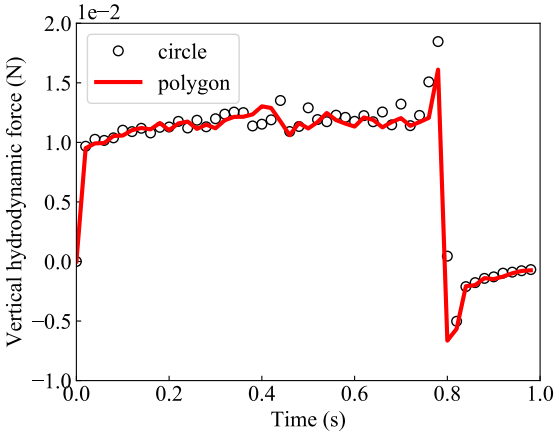


Figure 8: Comparison of drag force

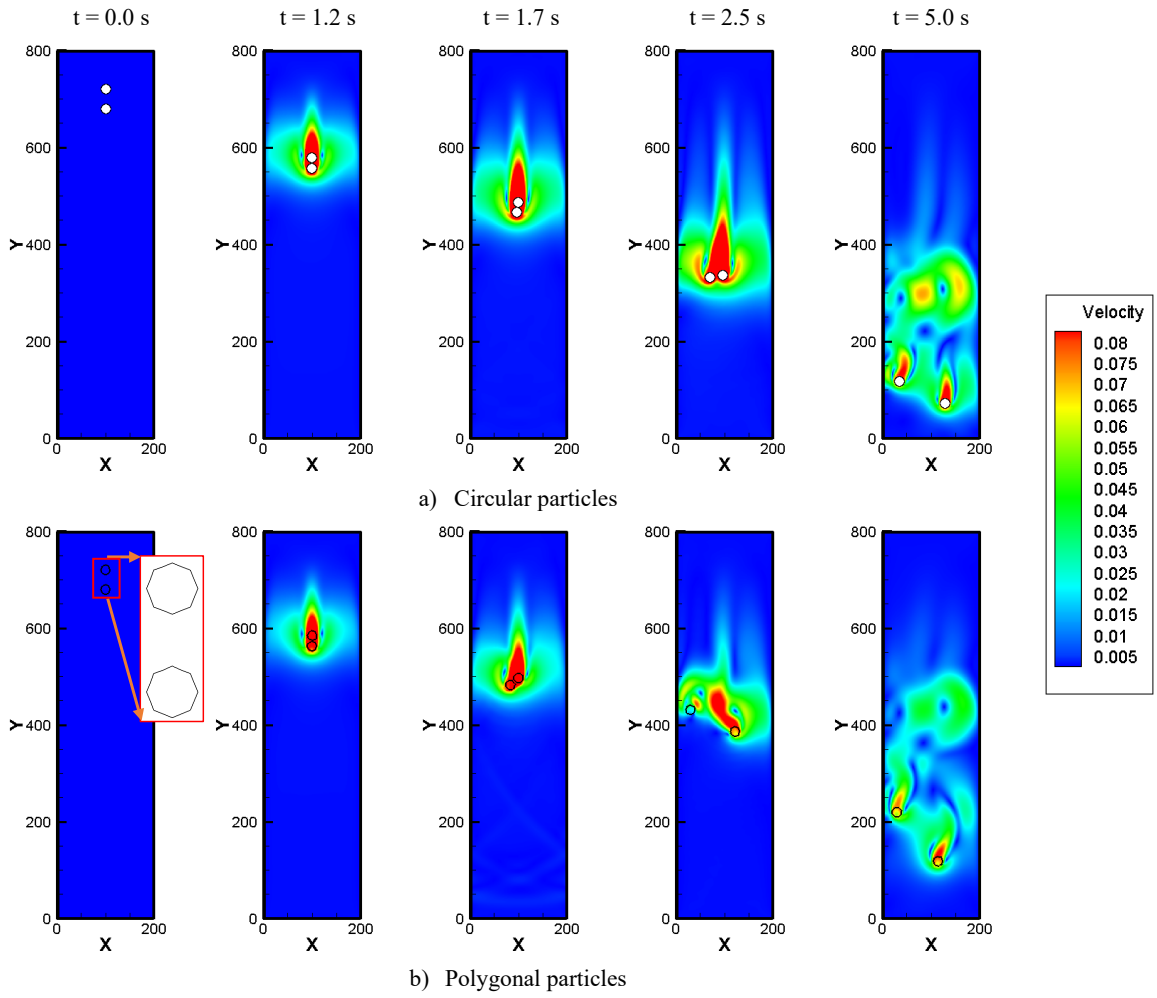
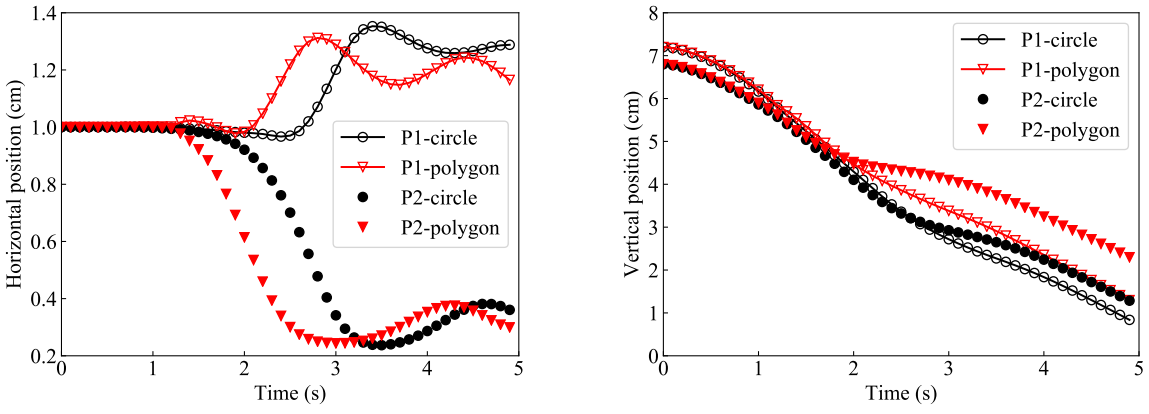


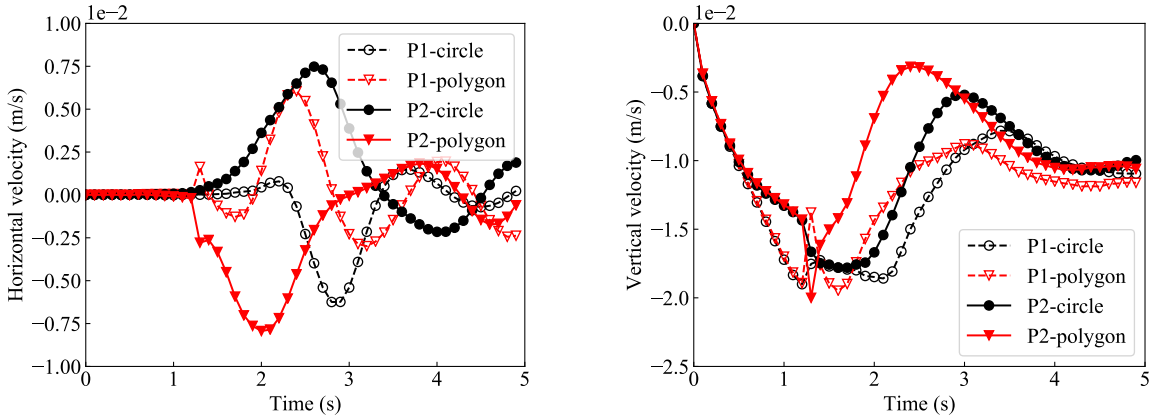
Figure 9: Comparison of DKT phenomena with different particle shapes



(a) X coordinates

(b) Y coordinates

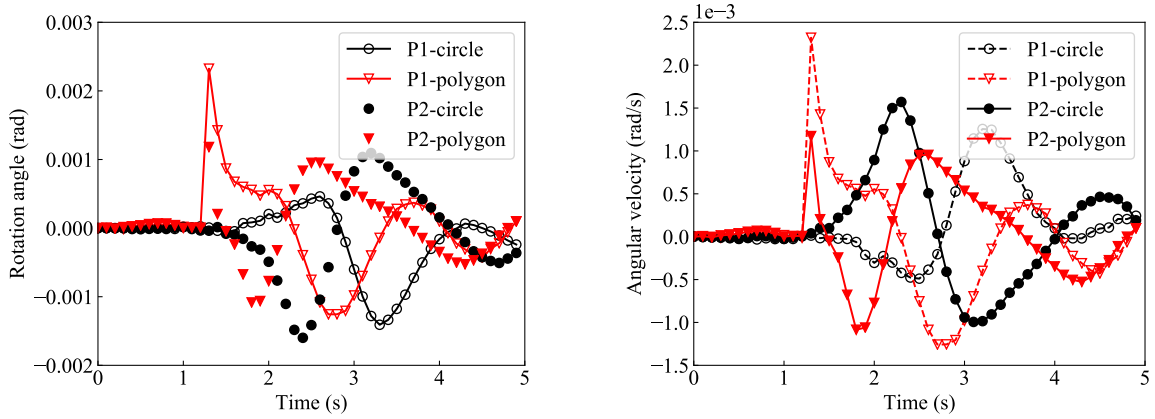
Figure 10: Comparisons of position of particles with different shapes



(a) horizontal velocity

(b) vertical velocity

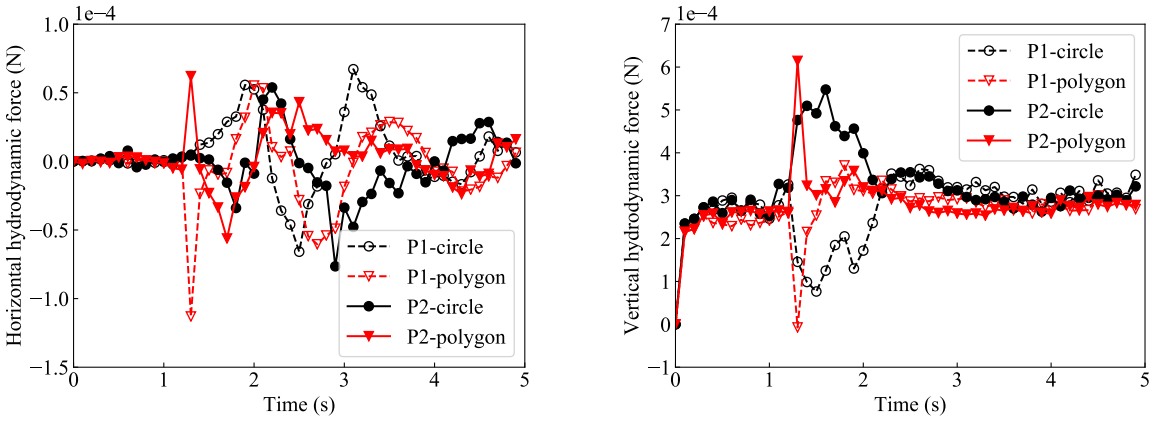
Figure 11: Comparisons of translational velocities of particles with different shapes



(a) rotation angle

(b) angular velocity

Figure 12: Comparisons of rotation of particles with different shapes



(a) lift force

(b) drag force

Figure 13: Comparisons of hydrodynamic forces

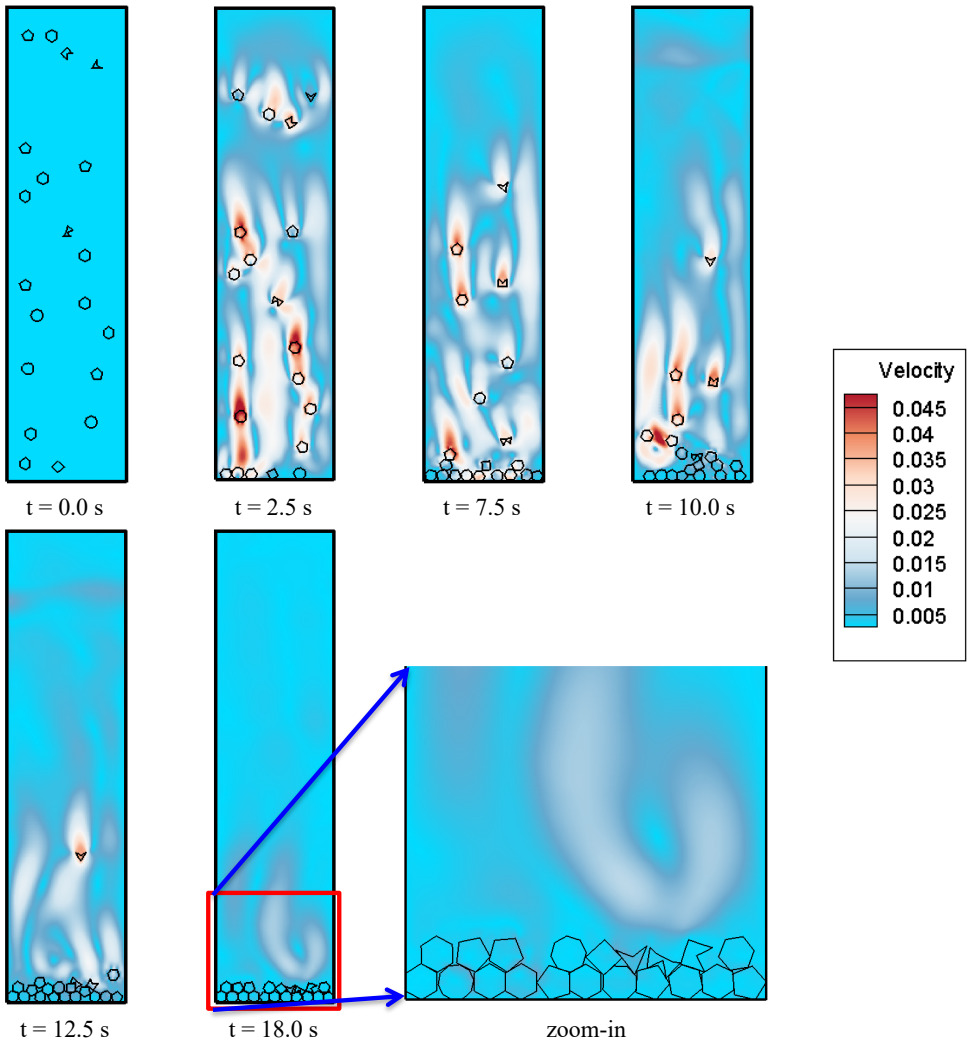


Figure 14: Snapshots of multiple-particle sedimentation process