FULLY RESOLVED SIMULATION OF PARTICULATE FLOW USING A SHARP INTERFACE DIRECT FORCING IMMERSED BOUNDARY METHOD

Jianming Yang∗
IIHR – Hydroscience and Engineering
University of Iowa
Iowa City, IA 52242
Email: jianming-yang@uiowa.edu

Frederick Stern
IIHR – Hydroscience and Engineering
University of Iowa
Iowa City, IA 52242
Email: frederick-stern@uiowa.edu

ABSTRACT
In this paper, the sharp interface, direct forcing immersed boundary method developed by Yang and Stern (A simple and efficient direct forcing immersed boundary framework for fluid-structure interactions, J. Comput. Phys. 231 (2012) 5029-5061) is applied to the fully resolved simulation of particulate flow. This method uses a discrete forcing approach and maintains a sharp profile of the fluid/solid interface. Also, it employs a strong coupling scheme for fluid-structure interaction through a predictor-corrector algorithm. The fluid flow solver is not included in the predictor-corrector iterative loop thanks to the direct forcing idea, which makes the overall algorithm highly efficient and very attractive for the fully resolved simulation of particulate flow with numerous solid particles. Several cases including sedimenting and buoyant particles and the interaction of two sedimenting particles showing kissing, drafting, and tumbling phenomenon are examined and compared with the reference results to demonstrate the simplicity and applicability of our method in particulate flow simulations.

INTRODUCTION
In Peskin’s immersed boundary methods [1], the effect of an immersed structure on a fluid flow is represented by a forcing term formulated as a Dirac delta function added to the momentum equation. In finite difference simulations using regular grids, this forcing term has to be discretized on grid points, but in general the immersed boundary and correspondingly the Dirac delta function won’t align with the grid points. Usually the Dirac delta function was smoothed out and given as discrete delta functions of various forms. Therefore, discrete delta functions play the crucial role of connecting the fluid flow and the immersed boundaries, i.e., to interpolate the fluid velocity from the (Eulerian) grid points to the (Lagrangian) structural elements and to spread out the boundary forces from the latter to the former in most implementations and variations of Peskin’s methods. On the other hand, Mohd–Yusof [2] derived a direct forcing formulation by imposing the velocity boundary condition at the immersed boundary “exactly” in the discrete-time equations, instead of starting from the Dirac delta function in the governing equations. This approach can be practically considered as a local solution reconstruction procedure for imposing the desired boundary conditions at the immersed boundary. Therefore, an explicit forcing term was not required to appear in the momentum equation, as detailed in [3]. Of course, it can also be formulated as adding a forcing term to the momentum equation, but this forcing term is no longer defined in the continuous space and associated with a boundary force density function. Interestingly, Uhlmann [4] combined this direct forcing idea with Peskin’s discrete delta function formulation, and developed a new immersed boundary method widely used in the numerical simulations of particulate flows [5,6]. A main motivation of Uhlmann’s development was to address the spurious force oscillation problem from the original finite difference formulation in [3] when applied to moving boundary problems. With a regularized Dirac delta function, the effect of immersed boundary represented by the momentum forcing function can be dis-
tributed smoothly among surrounding Eulerian grid points and this procedure is not very sensitive to the movement of the immersed boundary. Uhlmann’s original method was limited to particulate flows with large density ratios \( \rho_s/\rho_f > 1 \) with \( \rho_s \) and \( \rho_f \) the solid and fluid densities, respectively. This limitation was remedied along with other improvements in [5] and the improved method can handle cases with \( \rho_s/\rho_f > 0.3 \).

It should be noted that most studies using Uhlmann’s approach have been focused on simple two-dimensional (2D) geometries (such as circular cylinders and elliptical foils) and three-dimensional (3D) spherical solid particles. One important requirement for using this approach is the discretization of the immersed boundaries/surfaces, or generation of the Lagrangian meshes. It is a simple task for 2D geometries and 3D spherical surfaces on uniform grids. However, for 3D complex geometries, the surface mesh generation can be complicated: its resolution has to be consistent with that of the local Eulerian grid for accuracy consideration. In addition, for the improvement for low density ratio problems proposed in [5], the volume fraction function by the solid particles on the Eulerian grid has to be defined, which could be an expensive and complicated procedure too. Depending on the width of the regularized delta function, each Lagrangian point on the immersed surface has a group of Eulerian grid points as its support points. In general, the sharp interface between solid and fluid is blurred due to the smoothing procedure. This blurring interface affects the accuracy. In [6], an ad hoc retraction of the Lagrangian mesh toward the particle center was suggested to improve the overall accuracy, but the applicability of this treatment for different discrete delta functions or particles of different geometries is unknown. It is generally believed that approaches utilizing discrete delta functions are limited to low Reynolds number flows, since a blurring fluid/solid interface could be detrimental for resolving the boundary layers in higher Reynolds number flows.

Actually, in [3] validation was focused on stationary immersed boundaries, and the implications of boundary movement on a fixed grid in a time-splitting scheme, such as the fractional-step method, were not addressed. It turned out that non-physical historical information may enter the flow field in a time step when some grid points with reconstructed solutions at the previous time step become normal fluid points, as shown by Yang and Balaras [7]. This is one major source of the spurious force oscillation when the original method in [3] is applied to moving boundary problems. Yang and Balaras [7] proposed a field extension strategy to recover the correct historical information at those problematic points. Basically, the flow field is extended into the grid points with non-physical values near the immersed boundary through extrapolations at the end of each time step. The accuracy and effectiveness of this simple approach was demonstrated through a variety of moving boundary problems ranging from laminar to turbulent flows. Then this method was extended in [8] to fluid-structure interaction (FSI) problems with multiple rigid bodies using a strong coupling predictor-corrector scheme. Recently, Yang and Stern [9] greatly simplified the sharp interface immersed boundary method in [7] and expedited the fluid-structure coupling scheme in [8]. The new method involves (almost) minimized geometric operations for immersed boundary setup and moves the fluid solver out of the predictor-corrector iterative loop. In this method, simple geometries with analytical representations like sphere do not require a surface discretization. For complex geometries, arbitrary triangulations can be provided as long as the immersed boundaries are represented with enough resolution for the sole purpose of solid boundary approximation. Essentially, the immersed boundary discretization is not needed since there is no transfer of force/velocity information between from the immersed boundary (e.g., a triangulation element for complex geometries) and Eulerian grid points at all. Therefore, this approach can handle arbitrary geometries in real world applications. In terms of computational expense, the reformulated fluid-structure coupling scheme involves an iterative loop, in which the immersed boundary positions are adjusted and the momentum forcing term is re-evaluated until convergence. In principle, this is very similar to the iterative procedure for the Euler-Lagrange coupling in [5] and also the multi-direct forcing scheme used in [6] for better imposition of boundary conditions at the immersed boundary, except the particle position was pre-determined based on the fluid force on the particle from the previous time step. Nonetheless, in case of collision, the particle position and velocity were also iteratively adjusted in [6]. In general, our strongly coupled sharp interface method is very competitive in terms of computational efficiency. Furthermore, one additional apparent advantage of our sharp interface approach is that applications with much higher Reynolds numbers can be handled with ease through non-uniform grids and even wall modeling techniques. In [10] large-eddy simulation of a ship flow was performed on a Cartesian grid using immersed boundary method with the turbulent boundary layer solution from a RANS solver, which gave a conceptual verification of using wall modeling within the framework of sharp interface immersed boundary method. Currently there are some research toward this direction in the literature, but further discussion is beyond the scope of this paper.

The objective of this work is to demonstrate the capabilities of our method in fully resolved simulation of particulate flow. In [9], general fluid-structure interaction problems have been addressed and many different cases ranging from vortex-induced vibration and galloping to tumbling and fluttering have been shown. Here, incompressible viscous flows laden with spherical particles will be focused on. The rest of this paper is organized as follows: In the next section the mathematical model for particulate flow problems is introduced. Then in the numerical method section, following a summary of our approach given in [9], some simplifications specific to spherical particles will be discussed. In the results section, the accuracy of our method
will be first evaluated using a sedimenting particle problem. Low density ratio cases will be demonstrated. Then the interaction of two spherical particles during sedimenting will be studies, the drafting, kissing, and tumbling phenomenon will be discussed.

**MATHEMATICAL MODEL**

For general FSI problems, usually two types of coordinate systems are involved: an inertial reference frame fixed to or moving at a constant velocity with respect to the earth, and numerous non-inertial reference frames with each attached to a moving body under consideration. All fluid variables are conveniently defined in the inertial reference frame, whereas for a moving body, the motion can be described by the linear and angular displacement with respect to the inertial reference frame, and the linear and angular velocity with respect to the its own body-fixed non-inertial reference frame. For spherical particles, the mathematical formulation is much simplified thanks to the fact that the angular displacement of a particle is not necessary to be calculated.

**Navier–Stokes Equations**

In the inertial reference frame defined above, the Navier–Stokes equations governing the incompressible viscous flows can be written as:

\[
\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot (\mathbf{u} \mathbf{u}) = \frac{1}{\text{Re}} \nabla^2 \mathbf{u} + \mathbf{f},
\]

\[
\nabla \cdot \mathbf{u} = 0,
\]

where \( t \) is the time, \( \mathbf{f} \) represents the momentum forcing term due to the immersed boundary. The Reynolds number in Eq. (1) is defined as \( \text{Re} = \rho \mathbf{u} D / \mu \) with \( \rho \) the fluid density and \( \mu \) the dynamic viscosity, and the reference length scale \( D \) can be the particle diameter and the reference velocity scale \( \mathbf{u} \) can be the terminal settling velocity of the particle, for example.

**Equations for Particle Dynamics**

The dynamics of a spherical particle is described by Newton’s second law and Euler’s equation:

\[
m_s \frac{d \mathbf{v}}{d t} = \mathbf{F} + (m_s - m_f) \mathbf{g} + \mathbf{F}_c,
\]

\[
I_s \frac{d \Omega}{d t} = \mathbf{T},
\]

where \( m_s \) is the mass of the solid body, \( I_s \) is the moment of inertia of the body, \( \mathbf{g} \) the gravitational acceleration, \( m_f = \rho_f V_s \) the mass matrix of the fluid displaced by the solid body of volume \( V_s \). \( \mathbf{F} \) and \( \mathbf{T} \) are the fluid force and moment, respectively. \( \mathbf{F}_c \) is the collision force between particles. The external moment due to collision is not considered in this study.

**NUMERICAL METHOD**

The details of many aspects of the computational method in this work have been presented in several previous studies [7, 8, 10, 9]. Here only a summary is given with some simplifications specific to spherical particles.

**Basic Navier–Stokes Solver**

A fractional-step method is employed for velocity-pressure coupling, in which a pressure Poisson equation is solved to enforce the continuity equation. An second-order Adams–Bashforth scheme is used for time advancement. Semi-implicit schemes are available in the solver. The algorithm can be written as follows:

\[
\frac{\mathbf{u}^* - \mathbf{u}^{n-1}}{\Delta t} = \frac{3}{2} (D - C)^{n-1} - \frac{1}{2} (D - C)^{n-2} + \mathbf{f}^n;
\]

\[
\nabla^2 p^n = \frac{1}{\text{Re}} \nabla \cdot \mathbf{u}^*;
\]

\[
\mathbf{u}^n = \mathbf{u}^* - \Delta t \nabla p^n,
\]

where superscript \( n \) denotes time step, \( \mathbf{u}^* \) is the intermediate velocity vector, \( C \) and \( D \) are spatial operators containing the convective and viscous terms, \( \nabla \cdot \mathbf{u}^* \) and \( \frac{1}{\text{Re}} \nabla^2 \mathbf{u} \), respectively. The evaluation of the momentum forcing term \( \mathbf{f} \) will be discussed next.

The spatial derivatives are discretized using a second-order central difference scheme on a staggered Cartesian grid. The Poisson equation is solved using a semi-coarsening multigrid solver in the Hypre library [11].

**Momentum Forcing and Field Extension**

The definition of a moving spherical particle on the Cartesian grid is relatively easy. As shown in Fig. 1, a grid point inside the particle is identified as a solid point by checking if the distance between this point and the particle center \( O \) is smaller than the particle radius \( R \); if a grid point is in the fluid phase and one or more grid line segments connecting its immediate neighboring grid points are intersected by the fluid-particle interface, then this grid point is defined as an interface point; all the rest grid points are defined as fluid points.

Both solid points and interface points are defined as forcing points, since a forcing term is imposed on these points to represent the effect of an immersed rigid body on the fluid flow. In our direct forcing immersed boundary method, the solution near the immersed boundary is directly reconstructed such that the boundary conditions on the immersed boundary are satisfied.
With the current explicit time advancement scheme, the pressure gradient term is added to the intermediate velocity field \( \mathbf{u}^* \) for solution reconstruction with improved accuracy:

\[
\frac{\mathbf{\tilde{u}} - \mathbf{u}^*}{\Delta t} = -\nabla \mathbf{p}^{n-1},
\]

where \( \mathbf{p} \) is the pressure. The momentum forcing term is constructed on top of the auxiliary velocity field \( \mathbf{\tilde{u}} \). Essentially, to satisfy the velocity distribution inside and near the immersed body, a correction has to be applied to \( \mathbf{\tilde{u}} \),

\[
\mathbf{u}^n_{nf} = \mathbf{\tilde{u}} + \Delta t \mathbf{\tilde{f}}^n,
\]

such that the effect of the immersed body is included in \( \mathbf{u}^n_{nf} \). Therefore, substituting \( \mathbf{\tilde{u}} \) in Eq. (9) with the definition from Eq. (8), we can obtain the momentum forcing term based on \( \mathbf{\tilde{u}} \):

\[
\mathbf{\tilde{f}}^n = \frac{\mathbf{u}^n_{nf} - \mathbf{\tilde{u}}}{\Delta t} = \frac{\mathbf{u}^n_{nf} - \mathbf{u}^*}{\Delta t} + \nabla \mathbf{p}^{n-1}.
\]

For a particle undergoing a rigid body motion, the velocity at an arbitrary point on the particle surface or inside the solid is given by

\[
\mathbf{u}^n_{p} = \mathbf{v}^n + \Omega^n \times \mathbf{r},
\]

where \( \mathbf{r} \) is the position vector from the particle center to the point where the velocity is to be obtained. For a solid point, it is evident that \( \mathbf{u}^n_{p} = \mathbf{u}^n_{nf} \). For an interface point, in general it will not locate exactly on the body surface and a local reconstruction is necessary to obtain \( \mathbf{u}^n_{p} \). We use a linear interpolation stencil, which include one point on the immersed boundary \( (\mathbf{u}^n_{nf}) \) and three (two for 2D cases) fluid points near the interface point, as shown in Fig. 1, to reconstruct the expected \( \mathbf{u}^n_{p} \) using the auxiliary velocity field \( \mathbf{\tilde{u}} \). For details, the reader is referred to [7, 8, 10, 9]. Note that the body position \( \mathbf{d} \), and velocity \( \mathbf{u}^n_{nf} \), are unknown in a FSI problem and have to be obtained by solving the dynamic equations for rigid body motion.

In [9], a field extension operation was proposed to impose on the pressure gradient instead of the pressure field itself. The operation is similar to the local reconstruction of the velocity at an interface point, but on the pressure gradient component collocated at the same location. Also, for solid points, the pressure gradient is set to zero. The field extension strategy can greatly ameliorate the spurious force oscillation problems in sharp interface immersed boundary method. The reader is referred to [9] for a detailed discussion.

### Force and Moment Evaluation

For rigid body dynamics considered here, essentially, the integration of the forcing term \( f \) in the momentum equation for an immersed body, \( \int \rho f \mathbf{d}V \), represents the total effect of this body on the fluid flow, including the force, i.e., \( m_f \mathbf{v} \), required to impose a rigid body motion of the portion of fluid enclosed by the immersed boundary. Due to the field extension operation, the actual forcing term is

\[
f^n = \mathbf{u}^n_{nf} - \mathbf{\tilde{u}} - \nabla \mathbf{p}^{n-1}.
\]

The fluid force and moment on the particle can be written as

\[
\mathbf{F} = -\left( \rho_f \int f \mathbf{d}V - m_f \mathbf{v} \right),
\]

\[
\mathbf{T} = -\left( \rho_f \int \mathbf{r} \times f \mathbf{d}V - I_f \frac{d\Omega}{dt} \right),
\]

respectively. \( I_f \) is the moment of inertia of the portion of fluid enclosed in the volume \( V_f \) defined by the immersed boundary.
Predictor-Corrector Scheme

The equations for particle dynamics are rewritten as follows

\[ \dot{v}^n = m_s^{-1} F^n + m_s^{-1} F_c^n + \left(1 - \frac{m_f}{m_s}\right) g, \quad (15) \]

\[ \dot{\Omega}^n = I_s^{-1} T^n. \quad (16) \]

Hamming’s 4th-order predictor-corrector scheme [12] is used to solve the system of first-order ordinary differential equations (ODE).

The fluid-structure coupling procedure in one time step can be given as follows:

1. Perform the field extension operations for pressure gradient components at forcing points.
2. Perform the velocity predictor step without the forcing term.
3. Solve the particle dynamics equations and check convergence.
4. With the particle displacement and velocity from the above step, redefine the particle position and the fluid/solid/interface status of all grid points, and set up the local reconstruction and field extension stencils.
5. Define and integrate the momentum forcing term to obtain the fluid force and moment. Invert the governing equations for rigid body motion to acquire the acceleration of the structures.
6. If the predictor-corrector solver in step (3) reached convergence, go to the next step; otherwise go back to step (3).
7. Add the forcing term to the predicted velocity field.
8. Solve the Poisson equation to obtain the pressure field for the current time step.
9. Perform the velocity corrector to obtain the velocity field for the current time step.

In the above procedure, the predictor-corrector loop only includes the solution of the ODE system, the setup of immersed boundaries, and the definition of the momentum forcing term. Compared to the approaches in [5, 6], only the immersed boundary setup may produce extra computational cost. However, this step is very simple for a spherical solid object and doesn’t affect the overall efficiency at all.

RESULTS

In this part, a sedimenting particle in quiescent fluid will be calculated and compared with experimental results. Low density ratio buoyant particle cases will be demonstrated. Then the interaction of two spherical particles during sedimenting will be studied, the drafting, kissing, and tumbling phenomenon will be compared with reference simulations.

Settling and Rising Spherical Particles in a Quiescent Flow

In [13], the settling velocities of spherical particles of different materials and diameters were measured in a water tank. Case no. 2 was used in both [4] and [5] for validation. Here we use the same case and compare with there references. The Reynolds number is \( Re = \frac{v_c D}{\nu} = 367.4 \) and Froude number is \( Fr = \frac{v_c}{\sqrt{g D}} = 1.797 \) with the terminal speed of the particle \( v_c \). The computational domain is \([-16D, 16D] \times [-16D, 16D] \times [0, 0.614D]\) (in the two horizontal directions and the gravitational direction, respectively) and the grid is \( 96 \times 96 \times 1024 \). Similar to [5], uniform grid spacing of 0.06D and periodic flow condition is applied to the gravitational direction. However, to reduce the blockage effect from side boundaries, the grid is stretched in the two horizontal directions whereas the grid keeps a uniform spacing 0.06D around the sphere. The density ratio of the solid particle and the fluid is \( \rho_s/\rho_f = 2.56 \). As in [4] and [5], we use reference velocity scale \( v_{ref} = \sqrt{|g|D} \) and reference time scale \( t_{ref} = \sqrt{D/g} \) to normalize the results. The normalized time step is \( \Delta t = 0.012t_{ref} \).

The particle reaches its terminal settling velocity after about 25\( t_{ref} \). In Fig. 2, The instantaneous contours of the \( u_x \) and \( u_y \) velocity components and pressure are shown together with the \( \omega_y \) vorticity around the particle. At the present Reynolds number, the axisymmetric flow structure cannot sustain in reality. But in our simulation, the grid is very coarse and the simulation time is not long enough to trigger the mechanics which may break the axisymmetrical status of the sphere wake shown in the figure. One limitation of the current setup using periodic boundary condition in the gravitational direction is that the flow around the sphere will not be affected by the wake. In [9], a different from [13] was simulated. Therein, to avoid a too big computational domain, a non-inertial reference attached to the sphere was used with a slight modification of the convection terms. Of course, it can only be applied to cases with a single solid in an unbounded domain. The average terminal speed from the experiment gives \( v_c/v_{ref} = 1.797 \), i.e., the Froude number. The present simulation gives an average terminal speed of \( v_c = 1.817v_{ref} \), which is 1% higher than the experimental result. The time history of the settling velocity is shown in Fig. 3 and compared with the reference data with satisfactory agreement.

One of the major improvements over in Uhlmann’s method reported in [5] is an integration step for the flow field inside the particle to extend the applicable density ratio range down to 0.3. This was verified with the above simulation setup except different density ratios in [5]. Here, we perform similar simulations to show that our sharp interface direct forcing immersed boundary method with strong fluid-structure coupling can achieve the same low density ratios. Fig. 4 shows several different density ratios \( \rho_s/\rho_f = 0.30, 0.40, 0.50, 0.80, 0.90, 1.00, 1.10, 1.20 \) without any numerical stability issues. Here we also report the case
\( \rho_s/\rho_f = 0.29 \), which might be not accessible with the weakly fluid-structure coupling scheme in [5]. Interestingly, an attempt with an even lower density ratio \( \rho_s/\rho_f = 0.28 \) doesn’t success with our current method too, although in our method the structural part is strongly coupled with the fluid flow solver. One possible major reason is that the momentum forcing term is evaluated based on the predicted intermediate velocity field, and there is an \( O(h^2) \) difference between it and the final velocity field. The difference may act as a decoupling factor in cases with lower density ratios.

Two Drafting-Kissing-Tumbling Spherical Particles

In particulate flows, the interaction, i.e., collision, between particles have to be considered. In this part, we simulate two spherical particles of the same density and diameter sedimenting in a closed square container filled with incompressible vis-
cous fluid. The same conditions considered in [14] were used except some slight adjustments. The density ratio between the solid particles and the fluid is $\rho_s/\rho_f = 1.14$. The gravitational acceleration is $g = 981$ and the fluid viscosity is $\nu_f = 0.01$. The computational domain is $[0,6D] \times [0,6D] \times [-6D, 24D]$ (in the two horizontal directions and the gravitational direction, respectively) with $D$ the particle diameter. In [14] $[0,24D]$ was used in the vertical direction. We choose a “deep” container as a smaller collision force is used in our study. For the $i$th particle, the collision force [14] from $j$th particle is given by

$$F_{c,i} = \frac{m_j g}{\epsilon} \left[ \max \left( 0, -\frac{|d_{i,j}| - D - c_{\text{range}}}{c_{\text{range}}} \right) \right] \frac{d_{i,j}}{|d_{i,j}|}, \quad (17)$$

where $d_{i,j} = (x_i - x_j)$ is the distance vector between the centroids of particles $i$ and $j$, $c_{\text{range}}$ is the range of the repulsion force between particles related to the grid spacing in a numerical simulation, and $\epsilon$ is the model constant. In [14], two uniform grids of $\Delta h = 0.1D$ and $\Delta h = 0.075D$ were used, the corresponding grid dimensions are $60 \times 60$ and $80 \times 80$, respectively. They had $c_{\text{range}} = 0.1D$, i.e., the grid spacing of the coarse grid, and $\epsilon = 10^{-4}$. Here we use the same $c_{\text{range}}$, but with $\epsilon = 10^{-2}$ for milder collisions. Four uniform grids are used in our simulations with grid spacings $\Delta h = 0.1D, 0.075D, 0.05D$, and $0.025D$. The grid dimensions are $60 \times 60 \times 300$, $80 \times 80 \times 400$, $120 \times 120 \times 600$, and $240 \times 240 \times 1200$, respectively. The time steps on these four grids are $\Delta t = 1.0 \times 10^{-3}, 7.5 \times 10^{-4}, 4.0 \times 10^{-4}$, and $2.0 \times 10^{-4}$, respectively. In [14], the centroids of the two spherical particles are located on the vertical axis of the domain at $(3D, 3D, 21D)$ and $(3D, 3D, 18.96D)$. In our study, it was found that at this low Reynolds number the symmetry was difficult to break after the collision. Therefore, similar to [6], an offset is used in placing the two particles with the centroid of the upper one at $(3.03D, 3.03D, 21D)$ and that of the lower one at $(2.97D, 2.97D, 18.96D)$. All domain boundaries are solid walls and corresponding boundary conditions are applied.

The centroid trajectories of these two particles are shown in Fig. 5(a). The particles at positions corresponding to what given in other sub-figures (except (f)) are also illustrated. At $t = 0$, the particles are released with zero initial linear and angular velocities in the still fluid. These two particles fall down in the fluid due to gravity, the upper particle moves in the wake of the lower particle, which contains a downward motion. Therefore, the upper particle drops faster than the lower one and is being “drafted” toward the lower one (Fig. 5(b), (c), (d), and (e)). At a certain instant, the distance between these two particles is smaller enough and a collision happens between them (Fig. 5(f)). Of course, with the current collision mode, these two particles do not actually contact each other. For an obvious period after collision, these two particles keep a very close distance, which is also called the “kissing” stage (Fig. 5(g) and (h)). During the

**FIGURE 5. INSTANTANEOUS VORTICITY CONTOURS AT SEVERAL INSTANTS FOR TWO PARTICLES SEDIMENTING IN A FLUID: (A) TRAJECTORY; (B) $t = 0.08$; (C) $t = 0.16$; (D) $t = 0.24$; (E) $t = 0.32$; (F) $t = 0.328$, COLLISION; (G) $t = 0.40$; (H) $t = 0.48$; (I) $t = 0.56$; (J) $t = 0.64$; (K) $t = 0.72$; AND (L) $t = 0.80$**
“kissing” stage, the upper particle moves to a similar height of the lower one. Both particles start to developed their own wakes and the distance between them increases in this “tumbling” process. And finally they totally separate from each other, but their vortical wakes still interact (Fig. 5 (i), (j), (k), and (l)). These results were obtained from the finest grid with $\Delta h = 0.025D$.

Figures 6 and 7 show the time histories of the horizontal coordinates of the particle centroids and the horizontal velocity components of the particles on the four different grids. Due to the symmetrical arrangement of the $x$– and $y$–coordinates in the initial positions of the particles and the low Reynolds number range of the flow (on the finest grid, the maximum Reynolds number is 121.67), both horizontal velocity components (and horizontal coordinates of the centroid, correspondingly) for each particle exhibit very similar behaviors. On the coarsest grid with $\Delta h = 0.1D$, some oscillations in horizontal velocity components appear after the collision. But the oscillations disappear after the grid is slightly refined to $\Delta h = 0.075D$.

Figure 8 shows the time histories of the vertical coordinate of the particle centroids and the vertical velocity component of the particles on the four different grids. The differences in the results before collision from different grids clearly show a grid convergence trend as the grid is refined. Interestingly, the collision model depends heavily on the spatial and temporal resolution and as seen in the present case. Here the range of repulsion force remains a constant $c_{range} = 0.1D$, i.e., the grid spacing on the coarse grid, for all four grids. On the coarsest grid, due to the lack of spatial resolution, the distance between two particles tends to oscillate and frequently triggers the collision model to be involved, which is evident in the time evolution of the vertical velocity component following the first collision in Fig. 8(b). The oscillations disappear when the distance between these two particles is far enough and the particles enter the “tumbling” stage. On the two medium grids, the periods of repeated collisions are much shorter. However, on the finest grid, there are several clear rebounds after the first collision. This is because realistic collision has much smaller time and spatial scales than what can be resolved with our numerical discretization with reasonable cost. And the collision model used here is merely a mechanism to keep the particles apart. More advance collision models have to be involved if consistent results are to be sought regardless of spatial and temporal resolution.

Figure 9 shows the time history of the distance between two particles, which is a clear illustration of the “drafting”, “kissing”, and “tumbling” process. The upper particle approaches the lower one from above during the “drafting” stage. Then the collision happens and is followed by several rebounds between two particles. In the “kissing” stage the distance between two particles almost remains as a constant. Finally they slowly separate from each other in the “tumbling” stage.

These results are very similar to what presented in [14] and [6], although a direct comparison is not performed due to some small differences in the parameters.

**CONCLUSIONS**

In this work, we have applied our sharp interface direct forcing immersed boundary method developed in [9] to the fully re-
solved simulation of particulate flow. The advantages of the current sharp interface approach with strong fluid-structure coupling scheme, compared with the discrete delta function approach with a weak coupling scheme have been discussed. The numerical method given in [9] has been summarized and the simplifications for flows laden with spherical particles have been addressed. Our method uses a discrete forcing approach and the sharp profile of the fluid/solid interface is retained. High Reynolds number flows are accessible with the current method. The strong coupling scheme for fluid-structure interaction through a predictor-corrector algorithm doesn’t include the fluid flow solver in the predictor-corrector iterative loop. The overall algorithm is highly
efficent and can be applied for the fully resolved simulation of particulate flow with numerous solid particles. We have validated our method using the settling spherical particle case and verified that our method can handle low density ratio cases. The accuracy and applicability of our method has been further demonstrated through the case of two sedimenting particles undoing drafting-kissing-tumbling motion.

ACKNOWLEDGMENT

This work was sponsored by the Office of Naval Research (ONR) under grant N000141-01-00-1-7, with Dr. Patrick Purcell as the program manager. The simulations presented in this paper were performed at the Department of Defense (DoD) Supercomputing Resource Centers (Navy DSRC) through the High Performance Computing Modernization Program (HPCMP).

REFERENCES