Mirror fluid method for numerical simulation of sedimentation of a solid particle in a Newtonian fluid

Chao Yang* and Zai-Sha Mao†
Institute of Process Engineering, Chinese Academy of Sciences, Beijing 100080, China
(Received 27 November 2003; revised manuscript received 30 November 2004; published 16 March 2005)

The mirror fluid method is proposed for simulating solid-fluid two-phase flow. The whole computational domain is modeled as an Eulerian one for the fluid with a Lagrangian subdomain embedded in it. The boundary condition is enforced implicitly on solid-fluid surface segments by mirror relations. Thus, the total flow is solved in the one domain, in which the solid particle region is replaced with the virtual flow as the mirror image of outside flow. The mirror fluid method is implemented to compute the motion of a rigid spherical or elliptic particle in a Newtonian fluid for the purpose of method validation. The control volume formulation with the SIMPLE algorithm incorporated is used to solve the governing equations on a staggered grid in a two-dimensional coordinate system. A number of numerical experiments on falling particles are performed and the computational results are in good agreement with the reported experimental data.

DOI: 10.1103/PhysRevE.71.036704 PACS number(s): 83.85.Pt, 02.70.Bf

I. INTRODUCTION

Solid-fluid systems are encountered frequently in various important industrial processes such as fluidization, slurry transport, and permeation through packed beds. The motion of particulates has been explored theoretically or experimentally [1] and nowadays extensively via various computational fluid dynamics (CFD) approaches.

In order to obtain accurate solutions including the microstructure of clustered solid particles in a Newtonian or non-Newtonian fluid, Joseph and co-workers [2–6] have used the direct simulation approach to solve the governing equations for two phases by the body-fitted finite-element method. In their simulations, the sedimentation or fluidization of single, dozens of, and even hundreds of solid particles have been simulated successfully. A distributed Lagrange multiplier and fictitious domain method was also developed for simulating the motion of rigid particles suspended in a fluid [4,5]. The structured and regular meshes over an extended time-independent domain including the regions occupied by the particles were used to avoid repeated remeshing. The governing equations for the fluid were solved everywhere in the domain and the motion of the rigid particles was enforced using the distributed Lagrange multiplier method, whereas the finite-element method is generally more intricate than the finite-difference method and the continuity equation is not easy to be satisfied for incompressible fluids.

Ryskin and Leal [7] provided a method for the generation of an orthogonal boundary-fitted coordinate system, which is found to be very efficient in the solution of some free boundary problems with a deformable gas-liquid or liquid-liquid interface [8–10] and of viscous flow past solid spheres [11,12], with good accuracy in enforcing the boundary condition. However, it is very difficult to construct the orthogonal curvilinear coordinates for complicated and seriously deformed interfaces and extend the mapping procedure to three-dimensional flow domains. Unverdi and Tryggvason [13] presented a front-tracking method on a Eulerian grid by finite differences and the interface was explicitly tracked using a separate, unstructured grid that moves through a stationary grid. Good results of the direct simulation of bubbles motion were reported, but the algorithm seems difficult to be implemented for the need of repeatedly restructuring the grid.

The immersed boundary method is also a useful algorithm for simulating particulate flows by representing the role of the solid surface (usually through the boundary conditions) with a body force in the momentum balance equation *explicitly*. A direct momentum-forcing algorithm was proposed by Fadlun *et al.* [14] to overcome the disadvantage of the previous feedback forcing with two flow-dependent constants used normally to represent a solid body. The interpolation procedure for computing the momentum forcing in [14] was then improved by Kim *et al.* [15] using a linear or bilinear interpolation scheme and Gilmanov *et al.* [16] with the reconstruction algorithm using an unstructured triangular mesh for satisfying the no-slip conditions on the geometrically complex immersed boundaries. The flow past a solid sphere and some other particulate flows have been well predicted by these improved immersed-boundary methods. Recently the lattice-Boltzmann method has also been applied to simulate the particulate, especially multiparticle flows [17,18]. Feng and Michaelides [19] combined the lattice-Boltzmann method and the immersed-boundary method [14] to compute the sedimentation of single and even 1232 spheres successfully in three-dimensional coordinates. Kajishima and Takiguchi [20] applied a direct numerical simulation (DNS) based on a two-way-coupling finite-difference scheme for the simulation of the motion of solid particles through the introduction of the volume-weighted average velocity close to the solid-fluid interface.

Fedkiw and co-workers [21–25] designed the ghost fluid method (GFM) coupled with the level set approach and applied an "isobaric fix technique" to avoid a smearing jump of

*Electronic address: chaoyang@home.ipe.ac.cn
†Electronic address: zsmao@home.ipe.ac.cn

numerical quantities across the interface and to get continuous profiles for the density, viscosity, pressure, entropy, and velocity across the interface. This method combines a Lagrangian code for tracking the interface with a Eulerian scheme for advancing the fluid by populating cells next to the interface with "ghost values" (e.g., by extrapolation techniques). A "ghost cell" is defined everywhere in the computational domain and each node is designated values of variables for real fluid and ghost values for the other fluid that does not really exist at the point. The GFM coupled with the level set approach was originally developed for multiphase compressible flow [21], then extended successfully to treat shocks, deflagrations, detonations, and two-phase incompressible flames and flows [22–25].

Although significant efforts have been made to model multiphase flow, its numerical simulation is not fully successful and very often semiempirical methods have to be resorted to. Since the solid-fluid interfaces are translated along with the motion of particles and the coupling between the particles and fluid with jump interface conditions is rather complicated, the problem of simulating the unsteady motion and interaction of solid particles with a fluid is still very tough for presently available computational fluid dynamics methods. The level set approach and other numerical methods have been applied successfully for gas-liquid and liquid-liquid flows, typically by using a δ function or Heaviside function to couple the governing equations of the two phases. As the δ or Heaviside function formulations are applied for computing the Navier-Stokes equations, the numerical smearing of density, viscosity, or pressure across the interface will arise and some source terms for modeling the surface tension need to be added. However, these methods cannot be directly extended to solid-fluid systems for it is very difficult to assign suitable surface tension to solid-fluid interfaces.

Inspired by the fictitious domain method [3,4] and the ghost fluid method [21], it is proposed that the domain of solid particles is assigned suitable flow parameters (e.g., velocity and pressure) so that a surface segment is eventually subjected to the correct shear and normal forces on the fluid side of the segment. In this way, the sum of the real stress contributed by a real fluid and the fictitious stress by the mirror fluid (does not really exist) to the interface segment is kept zero, as if the segment is an artificial one immersed in the fluid. Otherwise, an infinite acceleration with no physical meaning will result onto the infinitely thin surface segment with zero mass. Thus, the solid surface seems just nonexistent and the whole domain including the real fluid and the inside mirror fluid can be solved altogether by a set of equations. At the same time, the mirror fluid should have the same values of density and viscosity as the real fluid, and the surface tension becomes irrelevant to the solution of the Navier-Stokes equations. The problem of interface boundary conditions including some jump conditions is thus replaced by that of specifying the fictitious parameters to the mirror fluid in the solid domain. In comparison to other ways of representing moving solids on a moving mesh, such as the finite-element method and the orthogonal boundary-fitted coordinate method, a usual fixed Eulerian grid in this method is used for the fluid without the need of remeshing with the subdomain occupied by the particle taken as a fictitious one. In this paper, we attempt to utilize the advantages of the algorithm—namely, the mirror fluid method—to give a comparatively robust and accurate finite-difference scheme for solid-liquid flows by solving the governing equations of the fluid phase without using the δ or Heaviside functions. The motion of rigid solid particles in the laminar flow regime is numerically simulated and compared with the reported experimental data to validate the new algorithm.

II. MATHEMATICAL FORMULATION

A. Governing equations

The unsteady motion of a solid particle in a Newtonian liquid is considered as a typical example for the validation of the mirror fluid method with following assumptions: (1) the fluid is viscous and incompressible, (2) the physical properties of the fluid and solid are constant, (3) the two-phase solid-fluid flow is axisymmetric or two dimensional, and (4) the flow is laminar.

The velocity $\mathbf{u}(\mathbf{x},t)$ and the pressure $P(\mathbf{x},t)$ subtracted, the contribution of gravity in the fluid are governed by the continuity and Navier-Stokes equations in the following forms:

$$\boldsymbol{\nabla}\cdot\mathbf{u}=0, \tag{1}$$

$$\rho_f\left(\frac{\partial\mathbf{u}}{\partial t}+\mathbf{u}\cdot\boldsymbol{\nabla}\mathbf{u}\right)=\boldsymbol{\nabla}\cdot\sigma, \tag{2}$$

where ρ_f is the density of the fluid and t is the time. Since the fluid is Newtonian, the stress tensor σ takes the form

$$\sigma=-P\mathbf{I}+\tau, \tag{3}$$

in which τ is the stress tensor defined as

$$\tau=\mu_f[\boldsymbol{\nabla}\mathbf{u}+(\boldsymbol{\nabla}\mathbf{u})^T], \tag{4}$$

where μ_f is the viscosity of the fluid.

Suppose a rigid sphere with radius R falls by gravity in a fluid. The equations of motion are nondimensionalized by introducing characteristic scales: $L=2R$ for length, $V_c=\sqrt{2Rg}$ for velocity, L/V_c for time, and $\rho_f V_c^2$ for pressure and stress. g is the gravitational acceleration (9.81 m s^{-2}). In a two-dimensional coordinate system, the equations for fluid mass and momentum conservation are then expressed in terms of dimensionless variables as

$$\frac{\partial u}{\partial x}+\frac{1}{r}\frac{\partial}{\partial y}(rv)=0, \tag{5}$$

$$\frac{\partial u}{\partial\theta}+\frac{\partial}{\partial x}\left(uu-\frac{1}{N_{\mathrm{Re}}}\frac{\partial u}{\partial x}\right)+\frac{1}{r}\frac{\partial}{\partial y}\left(rvu-r\frac{1}{N_{\mathrm{Re}}}\frac{\partial u}{\partial y}\right)=-\frac{\partial p}{\partial x}, \tag{6}$$

$$\frac{\partial v}{\partial \theta}+\frac{\partial}{\partial x}\left(uv-\frac{1}{N_{\mathrm{Re}}}\frac{\partial v}{\partial x}\right)+\frac{1}{r}\frac{\partial}{\partial y}\left(rvv-r\frac{1}{N_{\mathrm{Re}}}\frac{\partial v}{\partial y}\right)=-\frac{\partial p}{\partial y}-\left\{\frac{1}{N_{\mathrm{Re}}}\frac{v}{r^2}\right\}, \tag{7}$$

where $r\equiv 1$ for Cartesian coordinates, $r\equiv y$ for cylindrical coordinates, and curly brackets indicate the term present only in cylindrical coordinates. x, y, r, θ, u, v, and p are defined to be the dimensionless axial coordinate, radial or transverse coordinate, radial coordinate, time, axial velocity component, radial or transverse velocity component, and pressure, respectively. The dimensionless group N_{Re} is the Reynolds number and expressed as

$$N_{\mathrm{Re}}\equiv\frac{\rho_f L V_c}{\mu_f}=(2R)^{1.5}g^{0.5}\frac{\rho_f}{\mu_f}. \tag{8}$$

The rotating, tilting, and deforming of a rigid particle falling in a Newtonian fluid are absent in an axisymmetric case. Therefore, the motion of a solid particle with the density ρ_s satisfies Newton's law:

$$M_s\mathbf{a}=\mathbf{F}+\mathbf{G}, \tag{9}$$

$$\mathbf{a}=\frac{d\mathbf{U}}{dt}, \tag{10}$$

$$\mathbf{U}=\frac{d\mathbf{x}_P}{dt}, \tag{11}$$

where $\mathbf{x}_P$, $\mathbf{U}$, and $\mathbf{a}$ are the position, velocity, and acceleration vectors of the particle. For a solid sphere, the mass of the particle is

$$M_s=\frac{4}{3}\pi R^3\rho_s. \tag{12}$$

$\mathbf{F}$ is the force vectors exerted on the particle by the fluid and $\mathbf{G}$ by gravity:

$$\mathbf{F}=\int\int_{\Gamma_P}\sigma\cdot\mathbf{n}\,d\Gamma, \tag{13}$$

$$\mathbf{G}=\frac{4}{3}\pi R^3(\rho_s-\rho_f)\mathbf{g}, \tag{14}$$

where $\mathbf{g}$ is the acceleration vector of gravity. In Eq. (13), Γ_P is the surface of the particle and $\mathbf{n}$ is the outward unit vector normal to the particle surface. In addition, the no-slip condition is imposed on the solid-fluid boundary—i.e., $\mathbf{u}_\Gamma=\mathbf{U}$—to couple the governing equations of motion of fluid and the rigid particle, where $\mathbf{u}_\Gamma$ is the velocity vector of fluid on the fluid-solid interface.

After the nondimensionalization of Eq. (9) using the same characteristic scales for the fluid, we have

$$\frac{\pi}{6}\rho_2\tilde{\mathbf{a}}=\int\int_{\Gamma_P}\left(-p\mathbf{I}+\frac{1}{N_{\mathrm{Re}}}\tilde{\tau}\right)\cdot\mathbf{n}d\Gamma+\frac{\pi}{6}\frac{1}{N_{\mathrm{Fr}}}(\rho_2-\rho_1)\tilde{\mathbf{g}}, \tag{15}$$

where $\tilde{\tau}$ is the dimensionless form of the stress tension defined in Eq. (4), ρ_1 is the dimensionless fluid density and equal to 1, ρ_2 is the dimensionless density of the particle ($\rho_2=\rho_s/\rho_f$), and the Froude number is expressed as $N_{\mathrm{Fr}}\equiv V_c^2/Lg$. In two-dimensional coordinates the dimensionless vectors $\tilde{\mathbf{a}}$ and $\tilde{\mathbf{g}}$ can be written explicitly as

$$\tilde{\mathbf{a}}=\begin{pmatrix}a_x\\a_y\end{pmatrix}=\begin{pmatrix}dU_x/d\theta\\dU_y/d\theta\end{pmatrix}, \tag{16}$$

$$\tilde{\mathbf{g}}=\begin{pmatrix}g_x\\g_y\end{pmatrix}, \tag{17}$$

where U_x and g_x are the dimensionless velocity and gravitational acceleration of the particle in the axial direction (x), and U_y and g_y those at the radial or transverse direction (y).

Given another case of sedimentation of an elliptic solid particle in a two-dimensional channel, the governing equations of rotation have to be included in the above equations. The torque imposed on an ellipse by the fluid is given by

$$\mathbf{F_m}=\int\int_{\Gamma P}(\mathbf{x}-\mathbf{x}_P)\times\sigma\cdot\mathbf{n}\,d\Gamma. \tag{18}$$

Then the angular velocity ω of the solid particle rotation and the orientation angle α of the ellipse can be calculated by the following equations:

$$I\frac{d\omega}{dt}=\mathbf{F_m}, \tag{19}$$

$$\omega=\frac{d\alpha}{dt} \tag{20}$$

where I is the rotational moment of inertia of the particle.

B. Mirror fluid method

The basic idea of the mirror fluid method is to take the whole domain as a Eulerian one for the fluid with a Lagrangian subdomain embedded; i.e., a solid-fluid flow problem with geometric complexity domain is resolved on a larger and regular domain, including a mirror fluid domain which is originally occupied by the solid. In this aspect, the mirror fluid method and the fictitious-domain methods have the same advantages that a fixed and regular mesh can be applied for the entire computation without repeated remeshing or using boundary-fitted coordinates. A simple finite-difference method used here instead of the finite-element method and a complicated weak formulation [3,4]. The no-slip boundary condition is enforced implicitly on solid-fluid surface segments by mirror relations. Therefore, to make a surface segment subjected to a zero net force, the key is to guarantee the shear rates across the surface be the same magnitude but with opposite direction. This may be implemented by taking the

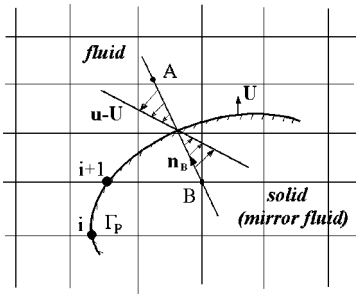

FIG. 1. Schematic diagram of the solid-fluid interface for the mirror fluid method.

inside flow in the subdomain occupied by the solid particle (i.e., the mirror fluid domain) as the flipped mirror image of the outside flow in the real fluid phase at the same surface segment or, in other words, by rotating the outside flow field ($\mathbf{u}-\mathbf{U}$) and pressure field by 180° around the surface segment (as shown in Fig. 1).

Unlike the linear extrapolation of the ghost fluid method [21,26], the flow parameters ($\mathbf{u}$ and P) are defined by the mirror relations. Thus, the key step of the mirror fluid method is to accurately specify the flow parameters inside the mirror fluid domain, in which the variable value at each node corresponds to that of a point in real fluid by the mirror relations. The following formulations are derived only in two-dimensional Cartesian or cylindrical coordinates for a single solid particle moving in a fluid as a typical example, while this is easy to be extended to a three-dimensional space and straightforward to multiple-particle–fluid systems.

As depicted in Fig. 1, we can find the mirror location of $A(x_A,y_A)$ in the real fluid region corresponding to node $B(x_B,y_B)$ in the mirror fluid domain in terms of a distance function defined as in the level set approach [27,28]. The signed algebraic distance function denoted as ϕ, being positive in the continuous fluid phase, negative in the solid phase, and zero at the solid-fluid interface, to facilitate the mirror calculations. After the particle is advanced by a time step, ϕ can be redesignated with the known center position of the particle. The unit normal vector to the interface in Eqs. (13) and (15) is denoted as

$$\mathbf{n} = \frac{\nabla \phi}{|\nabla \phi|}. \tag{21}$$

In two-dimensional Cartesian or cylindrical coordinates, the above equation is expanded as

$$\mathbf{n} = \begin{pmatrix} n_x \\ n_y \end{pmatrix} = \begin{pmatrix} \dfrac{\partial\phi/\partial x}{\sqrt{(\partial\phi/\partial x)^2 + (\partial\phi/\partial y)^2}} \\ \dfrac{\partial\phi/\partial y}{\sqrt{(\partial\phi/\partial x)^2 + (\partial\phi/\partial y)^2}} \end{pmatrix} \tag{22}$$

and can be calculated at the nodes immediately close to the solid-fluid interface. So the unit normal vector passing through B to the interface is thus calculated:

$$\mathbf{n}_B = \begin{pmatrix} n_{xB} \\ n_{yB} \end{pmatrix} = \left(\frac{\nabla \phi}{|\nabla \phi|} \right)_B. \tag{23}$$

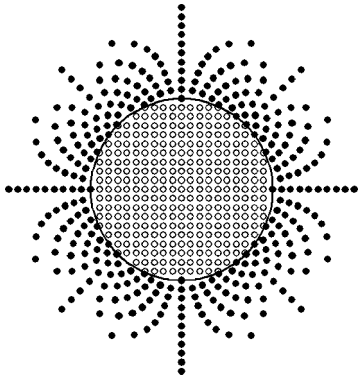

FIG. 2. Sketch map of a typical mirror relation between the nodes in a solid particle and the corresponding mirrored nodes in a real fluid.

In Fig. 1, the straight line along $\mathbf{n}_B$ passing through B is

$$\frac{x - x_B}{n_{xB}} = \frac{y - y_B}{n_{yB}}. \tag{24}$$

So the coordinates of A mirrored with B can be decided by solving the following equation set of Eqs. (25) and (26) coupled with the constraint in Eq. (27):

$$\frac{x_A - x_B}{n_{xB}} = \frac{y_A - y_B}{n_{yB}}, \tag{25}$$

$$(x_A - x_B)^2 + (y_A - y_B)^2 = (2\phi_B)^2, \tag{26}$$

$$\phi_A \phi_B \leqslant 0 \quad (\text{if } \phi_B \leqslant 0, \text{ then } \phi_A \geqslant 0). \tag{27}$$

As shown in Fig. 2, the points denoted by solid circle symbols (A) are one-to-one mirror images of the nodes in a solid particle denoted by open circle symbols (B). Then the fictitious velocity and pressure of node B in the mirror fluid are obtained easily:

$$\mathbf{u}_B = -(\mathbf{u}_A - \mathbf{U}) + \mathbf{U} = 2\mathbf{U} - \mathbf{u}_A, \tag{28}$$

$$P_B = P_A. \tag{29}$$

Be aware that for a rotating solid particle, the velocity at the particle-fluid interface—i.e., $\mathbf{U}$—should be replaced by $\mathbf{U} + \omega \times (\mathbf{x} - \mathbf{x}_P)$.

Such specification ensures that the shear and normal stresses on the two sides of the solid-fluid interface are with the same magnitude but opposite direction. The density and viscosity of the mirror fluid are designated as simply equal to those of the real fluid. As shown in Fig. 1, in the mirror relation the extension of velocity across the curved solid-fluid interface is continuous. The updated velocity field remains continuous across the particle surface; i.e., if ϕ_B is unlimitedly close to zero, $\mathbf{u}_A \approx \mathbf{U}$ can be derived from Eq. (28). Therefore, if the $\mathbf{u}$ at $\phi = 0$ (interface) is essentially the constant $\mathbf{U}$ within numerical error, then the sphere remains nondeformable and the no-slip boundary condition is automatically satisfied. In this way, we can update the fluid velocity field including the mirror fluid by the Navier-Stokes equation (2) at every node in the entire computational do-

TABLE I. Comparison of the relative error between the mirror fluid method and the method in [26]. E_P is for Poiseuille flow at $x=0$ and E_C for Couette flow at $r=4.0$ mm.

Grid spacing		1 mm	0.5 mm	0.25 mm
E_P	Yokoi [26]	2.16×10^{-2}	1.03×10^{-2}	4.67×10^{-3}
	This work	8.35×10^{-3}	1.99×10^{-3}	5.43×10^{-4}
E_C	Yokoi [26]	3.75×10^{-2}	2.00×10^{-2}	1.23×10^{-2}
	This work	1.79×10^{-2}	6.50×10^{-3}	2.81×10^{-3}

main with the solid-fluid interface boundary condition enforced implicitly. The motion of the solid particle is then computed by Eq. (9) using a Lagrangian scheme. Thus, one can compute solutions to solid-fluid two-phase flow problems replaced by solving two single-phase flows. Last, it is noted that only a band of several mirror nodes close to the interface is actually necessary by this novel algorithm. All the nodes in the particle domain whose neighbor nodes are all in the same domain are actually irrelevant to the solution and can be blocked out from the computation as suggested by Patankar [29].

In [26], the ghost fluid method [21] was used to impose the boundary condition of velocity on a solid object without including the pressure boundary condition. A distance function ψ (i.e., level set function) was also defined to track the interface, and the ghost velocity in the solid domain was estimated by a linear extrapolation and usually $|\psi_{\text{fluid}}| \neq |\psi_{\text{ghost}}|$. But in the mirror fluid method we set $|\phi_A|=|\phi_B|$ and use a definite one-to-one mirror relation; thus, the location of A in the fluid mirrored with B in the solid can be easily found by solving the explicit formulation of Eqs. (25) and (26) instead of the more costly iteration equation $\partial\mathbf{u}/\partial t_g \pm \mathbf{n}\cdot\nabla\mathbf{u}=0$ with Courant-Friedrichs-Lewey (CFL) number >1. In [26], only a first-order accuracy for the iteration equation was obtained. Even a high-order semi-Lagrangian scheme (the CIP method) was adopted, so the accuracy of all numerical simulations is first order. In fact, if we choose $|\psi_{\text{fluid}}|=|\psi_{\text{ghost}}|$, Eq. (7) in [26] is equivalent to Eq. (28) of this paper.

In order to further compare the difference of both methods, the same numerical samples—i.e., two-dimensional Poiseuille and Couette flow problems—are computed by the mirror fluid method. The relative deviation is also defined as $E=|u_{\text{sim}}-u_{\text{exact}}|/|u_{\text{exact}}|$. The theoretical solution of the Poiseuille flow is $u_{\text{exact}}=(1/2\mu)(-dp/dy)(ax-x^2)$, where a is the wall width ($0.5a=3.8$ mm). The exact solution of velocity for the Couette flow is $u_{\text{exact}}=\omega r$, where ω is a constant angular velocity of rotation ($\omega=1.0\ \text{s}^{-1}$) and r is the radius (the radius of the cylinder is 4.3 mm). As shown in Table I, for a flat boundary (Poiseuille flow), the computational accuracy of velocity for the mirror fluid method is nearly second order and between first and second order for a curved boundary (Couette flow), both being better than the first-order accuracy in [26]. Since the divergence-free condition can be satisfied more easily, a higher accuracy is obtained for the Poiseuille flow.

It is necessary here to comment on the immersed-boundary method [14–16,19], by which a body force term in different forms is added to the Navier-Stokes equations *explicitly* to make the solid-fluid interface boundary conditions enforced. A linear interpolation procedure with lower than second-order accuracy is usually adopted and a substantial task for computing the intersection points nearby the moving interface is usually needed. In [20], another interpolation method by the volume-weighted average was employed to determine the velocity in the interface field. All the above interpolation schemes are based on the assumption that the velocity across the boundary is linear. However, the mirror fluid relations rather than interpolation introduced in this work need no such *a priori* assumption. Moreover, the boundary conditions are *implicitly* satisfied in the mirror fluid method without modifying the physical momentum balance equations. The advantages of our method seem to be compensated for somewhat because a few iterations on the mirror relations are needed for each computational time step, when the solid body velocity is also part of the solution for the implicit time scheme (see Sec. II C 2).

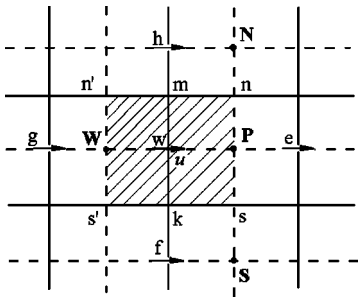

FIG. 3. The control volume of u at node w.

C. Computational scheme

1. Method for numerical solution

The control volume formulation with the power-law scheme described by Patankar [29] is adopted to solve Eqs. (6) and (7) for the fluid. A typical control volume for velocity component u in the x direction in a staggered grid is denoted by the shaded area in Fig. 3. Equation (6) is integrated over the control volume centered at w, and an algebraic equation for the central node w involved with four neighboring nodes e, f, g, and h is obtained. The momentum equation (7) for velocity component v is handled in a similar manner. In order to improve convergence and reduce solution costs, the SIMPLE algorithm [29] is adopted to compute the velocity and pressure fields. The discrete momentum and pressure-correction equations are left out for conciseness of this paper.

In order to ensure variables more accurately interpolated and revolved, a double fine grid detailed by Yang and Mao [27] is applied. When the motion of a bubble or droplet was simulated by a level set approach the inevitable "parasitic" surface currents that arise from errors in the surface force calculation were suppressed by this improvement [27]. As depicted in Fig. 3, the coarse-grid cells are shown with solid lines and the double fine-grid cells are demarcated by both solid and dashed lines. If a $N_x\times N_y$ grid is used for the velocity and pressure solution, a $2N_x\times 2N_y$ one is used for ϕ specification.

As suggested by Figs. 1 and 3, the solid-fluid surface Γ_P consists of surface segments formed by the intersection of the particle surface with the grid lines, and the evaluation of surface forces may be done by integration over these segments. For this purpose, the location and variables of two end points i and $i+1$ of segment i are determined by linear interpolation between grid nodes. In two-dimensional Cartesian or cylindrical coordinates, the dimensionless force vector $\tilde{\mathbf{F}}$ imposed on the particle by the fluid stress can be integrated along the solid-fluid interface as

$$\tilde{\mathbf{F}} = \int\int_{\Gamma_P} \begin{pmatrix} \left(-p + \frac{2}{N_{\mathrm{Re}}}\frac{\partial u}{\partial x}\right)n_x + \frac{1}{N_{\mathrm{Re}}}\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right)n_y \\ \frac{1}{N_{\mathrm{Re}}}\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right)n_x + \left(-p + \frac{2}{N_{\mathrm{Re}}}\frac{\partial v}{\partial y}\right)n_y \end{pmatrix} d\Gamma = \int\int_{\Gamma_P} \mathbf{E}\, d\Gamma, \tag{30}$$

where the vector $\mathbf{E}$ is the average of the stress at two ends of segment i when Eq. (30) is applied for surface integration of fluid stress—i.e., $\mathbf{E}=(\mathbf{E}_i+\mathbf{E}_{i+1})/2$. The surface area element $\Delta\Gamma$ in discrete form is

$$\Delta\Gamma = \frac{r_{i+1}+r_i}{2}\sqrt{(x_{i+1}-x_i)^2+(y_{i+1}-y_i)^2}. \tag{31}$$

If the grid is adequately fine, the error of integrated forces would be acceptable. A similar integration method is also applied to calculate the torque $\mathbf{F_m}$.

The dimensionless time step $\Delta\theta$ must satisfy the Courant-Friedrich-Lewy conditions and also the restrictions due to gravity and viscous terms to make the numerical procedure stable and convergent [28,30]:

$$\Delta\theta_v = \frac{3}{14}h^2 N_{\mathrm{Re}} \min\left(\frac{\rho(\mathbf{x})}{\mu(\mathbf{x})}\right), \tag{32}$$

$$\Delta\theta_c = \min\left(\frac{h}{|\mathbf{u}|}\right), \tag{33}$$

$$\Delta\theta = 0.5 \min(\Delta\theta_v, \Delta\theta_c), \tag{34}$$

where h is the mesh size of velocity-pressure grid and usually we take $\Delta x=\Delta y=h$.

2. Numerical procedure

By the mirror fluid method, the main calculation steps of solid-fluid flows are summarized as follows:

Step 1. Initialize the flow field $(\mathbf{u},\mathbf{U},P)$, $\mathbf{a}$, ω, α, physical parameters (density and viscosity), and ϕ as the signed normal distance to the interface.

Step 2. Assume the velocity of the solid particle be $\mathbf{U}$ at the end of a time increment Δt:

$$\mathbf{U} = \mathbf{U}_0 + \mathbf{a}_0\Delta t, \tag{35}$$

where $\mathbf{U}_0$ and $\mathbf{a}_0$ are the velocity and acceleration of the solid particle at the end of the former time step.

Step 3. Calculate the location (A) in the real fluid mirrored with any node (B) in the subdomain occupied by the solid particle through solving the equation set of Eqs. (25) and (26). Obtain ϕ_A, $\mathbf{u}_A$, and P_A by interpolation and assign values of $\mathbf{u}_A$ and P_A outside the particle to the mirror fluid ($\mathbf{u}_B$ and P_B) inside the particle with the surface values as the pivots by Eqs. (28) and (29).

Step 4. Solve the mass and momentum conservation equations of fluid in the whole domain by the SIMPLE algorithm with the interface boundary condition implicitly enforced by the mirror fluid method.

Step 5. Calculate the total drag force of $\mathbf{F}$ exerted by the liquid on the particle and the body force of $\mathbf{G}$ by the external field. Then calculate the acceleration $\mathbf{a}$ by solving the force balance Eq. (9) of the particle.

Step 6. Estimate the particle velocity $\mathbf{U}^*$ at the end of current time step according to the new acceleration value:

$$\mathbf{U}^* = \mathbf{U}_0 + \tfrac{1}{2}(\mathbf{a}_0+\mathbf{a})\Delta t. \tag{36}$$

Step 7. Compare $\mathbf{U}^*$ and $\mathbf{U}$. If $\mathbf{U}^*=\mathbf{U}$ (meaning $\mathbf{a}=\mathbf{a}_0$), and go on to the next time step of the simulation. If not equal, go back to step 2 with suitably adjusted $\mathbf{a}_0$, typically as

$$\mathbf{a}_0^{(n+1)} = \mathbf{a}_0^{(n)} + \tfrac{1}{2}(\mathbf{a}-\mathbf{a}_0^{(n)}), \tag{37}$$

until the value of $\mathbf{U}$ converged during the iterations (i.e., $\mathbf{U}^*$ approaches a convergent value). If the solid particle rotates, use the converged velocity field $\mathbf{u}$ and $\mathbf{U}$ to calculate the angular velocity ω and orientation angle α of the particle by Eqs. (18), (19), and (20).

Step 8. Move the particle center to a new position with the converged surface velocities and respecify ϕ of fluid and solid particle. Keep the $\mathbf{u}$ and P values for nodes in the fluid and discard these in the particle according to the renewed signed distance function ϕ.

Step 9. Let $\mathbf{U}_0=\mathbf{U}$, $\mathbf{u}_0=\mathbf{u}$, $\mathbf{a}_0=\mathbf{a}$, $\omega_0=\omega$, and $\alpha_0=\alpha$, and repeat steps 2–8 for the next time step.

III. NUMERICAL RESULTS AND DISCUSSION

To demonstrate and validate the applicability and robustness of the above-mentioned numerical algorithm, we consider an impressible Newtonian fluid bounded in an adequately wide cylindrical column with free-slip boundary conditions imposed at the column wall and a rigid spherical particle that moves freely in the fluid. Typically we solve the solid-fluid flow in a computational domain $\Omega=\{(x,y)\,|\,0\leqslant x\leqslant 45R,\ 0\leqslant y\leqslant 36R\}$ to assure no wall effect. In case of particle rotation, the sedimentation of an elliptic particle in a two-dimensional channel is simulated.

A. Convergence test

A number of numerical experiments with different particle radius, viscosity, and density in wide range of Reynolds number in axisymmetrical and laminar two-phase flow are carried out. The effects of grid on calculated velocities of the spherical particle and fluid are presented in Figs. 4, 5, and 6. Grids of 50×40, 100×80, 150×120, 200×160, 250×200, and 300×240 are used to test the convergence of the

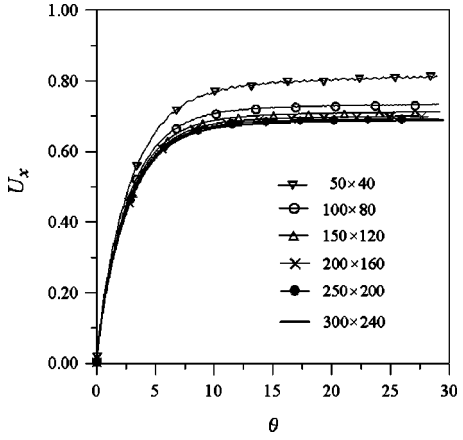

FIG. 4. Grid independence test: dimensionless velocity of particle as a function of dimensionless time ($R=0.21$ mm, $\rho_f=1000$ kg m^{-3}, $\rho_s=2000$ kg m^{-3}, $\mu_f=1.0\times10^{-3}$ Pa s).

particle motion as a function of time (see Fig. 4). Figure 5 depicts the distribution of interfacial velocity along the particle surface against the azimuthal angle β measured from the particle rear stagnant point. The real axial velocity is a constant at the whole surface, and the simulation velocity

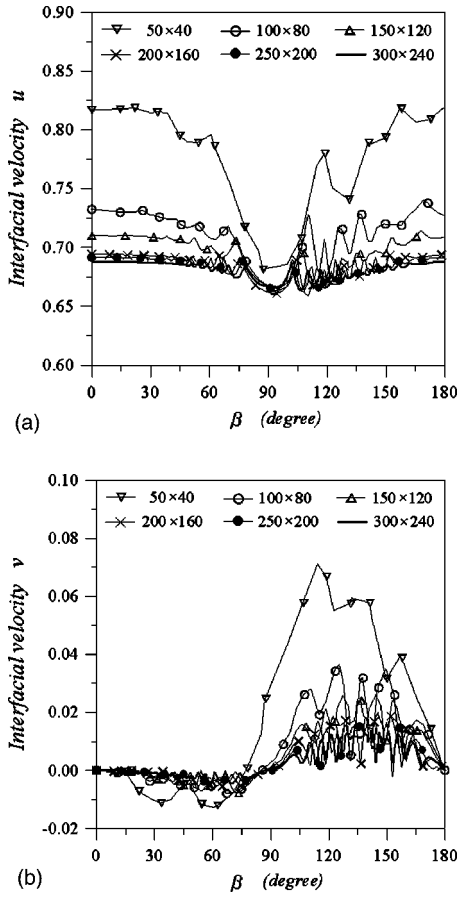

FIG. 5. Profiles of the dimensionless interfacial velocity along the particle surface with different computational grids (the same simulation conditions as in Fig. 4). (a) Interfacial velocity at axial direction (u). (b) Interfacial velocity at radial or transverse direction (v).

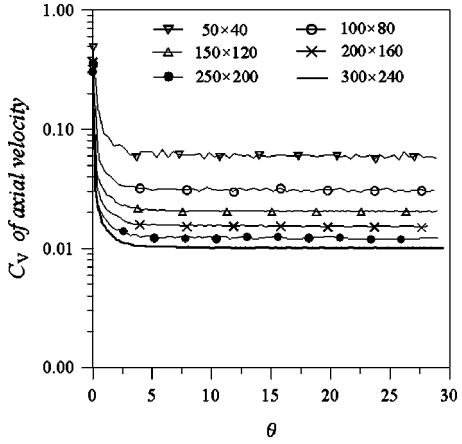

FIG. 6. Coefficient of the variation of the axial velocity (u) versus dimensionless time with different computational grids (the same simulation conditions as in Fig. 4).

component shows certain fluctuations. We use the coefficient of variation C_V to characterize quantitatively the deviation of the calculated interfacial velocity from the ideal value as shown in Fig. 6. C_V is defined as

$$C_V=\frac{1}{|\overline{w}|}\sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(w_i-\overline{w})^2}, \tag{38}$$

where w_i is the interfacial velocity either in axial direction (u) or in radial or transverse direction (v) at every intersecting point of the particle surface with the grid lines, and the average velocity $\overline{w}$ with total number n of interfacial velocities is computed by

$$\overline{w}=\frac{1}{n}\sum_{i=1}^{n}w_i. \tag{39}$$

We can find the simulation velocities at the solid-fluid interface gradually approach to constant values with the increase of the total number of nodes. The constant velocity **u** at the curved solid-fluid interface also validates the proposed mirror relation can guarantee the no-slip condition and no deformation of the rigid particle.

Table II compares the predicted Reynolds numbers ($N^*_{\mathrm{Re_{sim}}}$) of the sphere at steady-state motion with $N^*_{\mathrm{Re_{cor}}}$ calculated from a correlation recommended by Clift *et al.*

TABLE II. Convergence study (the same simulation conditions as in Fig. 4).

Grid	$N^*_{\mathrm{Re_{sim}}}$	$N^*_{\mathrm{Re_{cor}}}$	D	C_V of u	C_V of v
50×40	22.03	18.55	18.76%	5.79%	3.49%
75×60	20.37	18.55	9.81%	4.01%	2.59%
100×80	19.71	18.55	6.25%	3.02%	2.06%
150×120	19.14	18.55	3.18%	2.02%	1.45%
200×160	18.82	18.55	1.46%	1.51%	1.11%
250×200	18.62	18.55	0.38%	1.20%	0.90%
300×240	18.57	18.55	0.11%	1.00%	0.76%

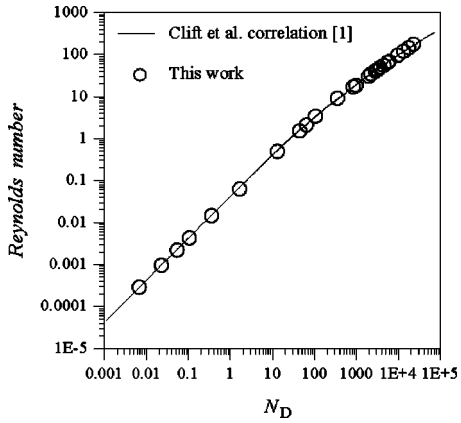

FIG. 7. Plot of the Reynolds number against N_D for the motion of a spherical particle ($R=0.21$ mm, $\rho_f=1000$ kg m^{-3}, $\rho_s=2000$ kg m^{-3}, $\mu_f=1.0\times10^{-3}$ Pa s).

(Table 5.3, p. 114) [1]. The real Reynolds number N^*_{Re} of the particle is defined as

$$N^*_{\mathrm{Re}}=\frac{\rho_f LU}{\mu_f}, \tag{40}$$

where U is the real terminal velocity of the spherical particle.

The correlation for $12.2<N^*_{\mathrm{Re}}\leqslant 6.35\times10^3$ and $580<N_D\leqslant 1.55\times10^7$ is given by

$$\log_{10} N^*_{\mathrm{Re}}=-1.81391+1.34671A-0.12427A^2+0.006344A^3, \tag{41}$$

where $A=\log_{10} N_D$ and the dimensionless number N_D is

$$N_D=\frac{4}{3\mu_f^2}\rho_f|\rho_s-\rho_f|gL^3. \tag{42}$$

For other two ranges of N_D, the correlations that will be involved in the following sections are

$$N^*_{\mathrm{Re}}=\frac{N_D}{24}-1.7569\times10^{-4}N_D^2+6.9252\times10^{-7}N_D^3-2.3027\times10^{-10}N_D^4, \tag{43}$$

used for the range of $N^*_{\mathrm{Re}}\leqslant 2.37$ and $N_D\leqslant 73$, and

$$\log_{10} N^*_{\mathrm{Re}}=-1.7095+1.33438A-0.11591A^2, \tag{44}$$

for the range of $2.37<N^*_{\mathrm{Re}}\leqslant 12.2$ and $73<N_D\leqslant 580$. The relative deviation between the predicted Reynolds number by the mirror fluid method and the value from correlation (41) and (43), or (44) is calculated by

$$D=(N^*_{\mathrm{Re}_{\mathrm{sim}}}-N^*_{\mathrm{Re}_{\mathrm{cor}}})/N^*_{\mathrm{Re}_{\mathrm{cor}}}\times 100\,\%\,. \tag{45}$$

As the grid is refined, the deviation of the Reynolds number of the spherical solid particle at steady-state motion decreases gradually (see Table II) and a grid with 250×200 nodes is sufficient for spatial computational accuracy and is adopted for the subsequent simulations.

TABLE III. Comparison of the Reynolds numbers of solid particles.

N_D	$N^*_{\mathrm{Re}_{\mathrm{sim}}}$	$N^*_{\mathrm{Re}_{\mathrm{cor}}}$	D
6.695×10^{-3}	2.887×10^{-4}	2.789×10^{-4}	3.51%
2.259×10^{-2}	9.599×10^{-4}	9.413×10^{-4}	1.98%
5.356×10^{-2}	2.209×10^{-3}	2.231×10^{-3}	−0.986%
0.1046	4.286×10^{-3}	4.356×10^{-3}	−1.61%
0.3530	1.455×10^{-2}	1.469×10^{-2}	−0.953%
13.08	0.4969	0.5163	−3.76%
44.13	1.522	1.555	−2.12%
64.24	2.110	2.131	−0.985%
104.6	3.406	3.255	4.64%
353.0	9.129	8.667	5.33%
836.8	17.13	16.55	3.50%
968.7	18.52	18.55	−0.162%
1.947×10^3	30.90	31.41	−1.62%
2.175×10^3	34.73	34.07	1.94%
2.824×10^3	40.75	41.14	−0.95%
3.116×10^3	43.67	44.13	−1.04%
3.591×10^3	49.49	48.78	1.46%
4.485×10^3	56.22	56.98	−1.33%
5.516×10^3	65.98	65.70	0.426%
5.969×10^3	69.26	69.33	−0.101%
9.532×10^3	97.00	94.83	2.29%
1.308×10^4	120.6	116.6	3.43%
1.740×10^4	146.7	140.0	4.79%
2.259×10^4	177.1	165.0	7.33%

B. Comparison of the Reynolds number and drag coefficient

In this section, we compare the predicted Reynolds numbers and drag coefficients of falling spherical particles in stagnant fluid with the experimental data to examine the robustness of the novel mirror fluid method. As shown in Fig. 7 and Table III, the numerical simulated Reynolds numbers of falling particles are in good agreement with the values calculated from the typical empirical correlations of Eqs. (41), (43), and (44) proposed by Clift *et al.* [1] with relative deviation mostly below 5%. As shown in Fig. 8 and Table IV, the comparison of drag coefficients of a sphere in steady-state motion is also rather satisfactory. The values of drag coefficient $C_{d,\mathrm{cor}}$ are calculated from three experimental correlations also recommended by Clift *et al.* (Table 5.2, p. 112) [1]:

$$C_d=\frac{3}{16}+\frac{24}{N^*_{\mathrm{Re}}},\quad \text{if } N^*_{\mathrm{Re}}<0.01, \tag{46}$$

$$C_d=\frac{24}{N^*_{\mathrm{Re}}}[1+0.1315(N^*_{\mathrm{Re}})^{(0.82-0.05\log_{10}N^*_{\mathrm{Re}})}],\quad \text{if } 0.01<N^*_{\mathrm{Re}}\leqslant 20, \tag{47}$$

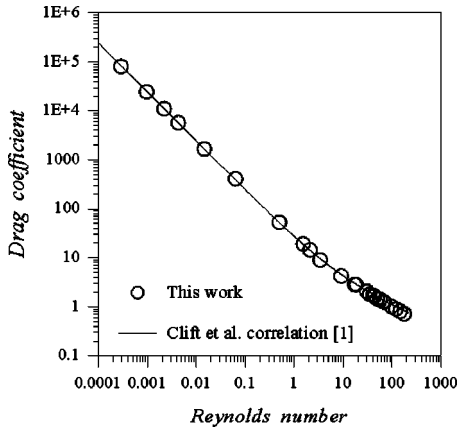

FIG. 8. Plot of the drag coefficient against Reynolds number of a falling spherical particle (the same simulation conditions as in Fig. 7).

$$C_d = \frac{24}{N^*_{\mathrm{Re}}}[1 + 0.1935(N^*_{\mathrm{Re}})^{0.6305}], \quad \text{if } 20 \leqslant N^*_{\mathrm{Re}} \leqslant 260. \tag{48}$$

The relative deviation between the predicted drag coefficient $\left(C_{d,\mathrm{sim}} = \frac{4}{3} LgU^{-2}\rho_f^{-1}\left|\rho_s - \rho_f\right|\right)$ by the mirror fluid method and the value from the above correlations is calculated by

$$D_c = (C_{d,\mathrm{sim}} - C_{d,\mathrm{cor}})/Cd_{d,\mathrm{cor}} \times 100\,\% . \tag{49}$$

TABLE IV. Comparison of the drag coefficients of solid particles.

$N^*_{\mathrm{Re}_{\mathrm{sim}}}$	$C_{d,\mathrm{sim}}$	$C_{d,\mathrm{cor}}$	D_c
2.887×10^{-4}	8.031×10^{4}	8.312×10^{4}	−3.38%
9.599×10^{-4}	2.452×10^{4}	2.500×10^{4}	−1.92%
2.209×10^{-3}	1.097×10^{4}	1.086×10^{4}	1.01%
4.286×10^{-3}	5.694×10^{3}	5.600×10^{3}	1.68%
1.455×10^{-2}	1.668×10^{3}	1.654×10^{3}	0.846%
0.4969	52.96	51.84	2.16%
1.522	19.04	18.68	1.93%
2.110	14.43	14.10	2.34%
3.406	9.017	9.497	−5.05%
9.129	4.236	4.535	−6.59%
17.13	2.853	2.990	−4.58%
18.52	2.823	2.847	−0.843%
30.90	2.039	2.084	−2.16%
34.73	1.804	1.943	−7.15%
40.75	1.701	1.769	−3.84%
43.67	1.634	1.700	−3.88%
49.49	1.466	1.583	−7.39%
56.22	1.419	1.475	−3.80%
65.98	1.267	1.351	−6.22%
69.26	1.244	1.317	−5.54%
97.00	1.013	1.104	−8.24%
120.6	0.8998	0.9894	−9.06%
146.7	0.8087	0.8988	−10.0%
177.1	0.7200	0.8212	−12.3%

The numerical simulation results of $C_{d,\mathrm{sim}}$ coincide with $C_{d,\mathrm{cor}}$ rather well except the cases of $N_{\mathrm{Re}} > 100$, for which the relative error is about 10%. A similar trend is observed for the comparison of Reynolds numbers of falling particles with different radii. In [4] (Tables 8.3–8.6), Glowinski *et al.* have also simulated the motion of a ball falling in an incompressible viscous fluid by a fictitious domain approach and the finite-element method and compared with the experimental data from Clift *et al.* [1]. The relative errors of terminal velocities for Reynolds numbers of 38.1, 78.3, 84, and 118 are 12%, 17%, 13.4%, and 15.2%, respectively. The possible reason for the larger prediction deviation from the experimental data is that the solid particle may oscillate at high Reynolds numbers, typically when the Reynolds number of a falling sphere is about larger than 130. In this case, the axisymmetric assumption is not suitable and three-dimensional simulation should be performed for more accurate prediction.

C. Comparison of the wake length

Besides the investigation of the drag for the motion of a rigid particle in a fluid, the flow pattern at different Reynolds numbers has also been concerned and the wake length is one of the important parameters to characterize the solid-fluid flow.

The velocity field around a falling solid sphere computed by the mirror fluid method is shown in Fig. 9 as a typical example. The velocity vector map in Fig. 9(a) is plotted in the stationary frame, but that in Fig. 9(b) is in the coordinates fixed at the center of the falling spherical particle, where the wake behind the particle is obvious. According to the velocity field in Fig. 9, we can also explain the reason of the fluctuation of the interfacial velocity along the particle surface as shown in Fig. 5. The interfacial velocity of the node nearby the rear of the particle is close to a constant value for the nearly same magnitude of velocities of the neighboring nodes. However, the velocity gradients at the interface nodes near to the front nose of the particle are much larger than other nodes, so the corresponding interfacial velocities at these nodes fluctuate relatively seriously.

Figure 10 presents the computed stream function contours around the falling particle with different Reynolds numbers reaching steady state at sufficiently large θ. When N^*_{Re} is very low (18.52), the flow around the particle is perfectly laminar and no vortex ring appears in the wake. As the Reynolds number is increased, the stream function contours at the rear of the particle become more flexuous. When $N^*_{\mathrm{Re}_{\mathrm{sim}}}$ is 69.26, the circulating vortex emerges in the neighborhood of the rear stagnation point. With the further increase of the Reynolds number, the vortex ring grows in size and becomes more and more elongated.

In Fig. 11, the wake lengths at different Reynolds numbers obtained from the simulations by the mirror fluid method are compared with Taneda's experimental data [31]. The agreement seems satisfactory except the cases of very

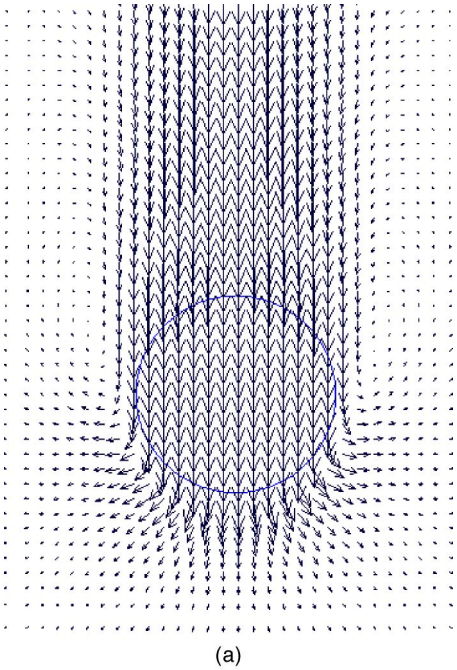

FIG. 9. Velocity field around a falling sphere in a fluid ($N_D = 1.740 \times 10^4$, $N^*_{\mathrm{Re}_{\mathrm{sim}}} = 146.7$, $\theta = 21.04$).

high Reynolds numbers when the zigzag motion of falling particles with an unsteady wake needs to be dealt with three-dimensional simulation. The predicted wake lengths are also very close to the simulated results by Mao and Chen [12] using an orthogonal, boundary-fitted coordinate system.

D. Sedimentation of an elliptic particle with rotation

To further evaluate the reliability of the mirror fluid method, a single ellipse settling in a Newtonian fluid under gravity in a two-dimensional vertical channel is simulated under the same conditions as used by Huang *et al.* [5]. The

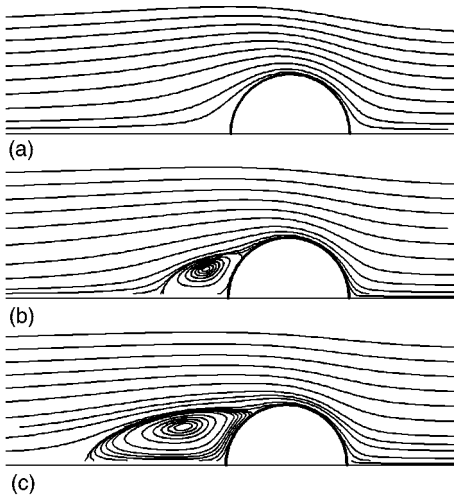

FIG. 10. Influence of the Reynolds numbers (N^*_{Re}) on stream function contour maps for the motion of a rigid sphere in a fluid. (a) $N_D = 968.7$, $N^*_{\mathrm{Re}_{\mathrm{sim}}} = 18.52$, $\theta = 18.05$. (b) $N_D = 5.969 \times 10^3$, $N^*_{\mathrm{Re}_{\mathrm{sim}}} = 69.26$, $\theta = 13.44$. (c) $N_D = 1.740 \times 10^4$, $N^*_{\mathrm{Re}_{\mathrm{sim}}} = 146.7$, $\theta = 21.04$.

elliptic particle with the aspect ratio $R_a/R_b = 1.5$, where R_a is the major axis and R_b the minor axis, rests initially at the channel center and the major axis tilts 135° with respect to the horizontal direction. For the close wall sedimentation, the width W of the channel is 5 times the length of the major axis. The inflow boundary of the computational domain is placed $30R_a$ ahead of the ellipse and the outflow boundary is $40R_a$ behind the particle. In [5], the orientation angle and trajectory of an ellipse with two Reynolds numbers of $N^*_{\mathrm{Re}} = 0.31$ and $N^*_{\mathrm{Re}} = 0.82$ were simulated directly using the Galerkin finite-element method and an arbitrary Lagrangian-Eulerian moving mesh technique. The authors found that in a narrow channel the lubrication pressure turns the ellipse vertical and executes a damped oscillation as it drifts to the channel center for the case of lower Reynolds number ($N^*_{\mathrm{Re}} = 0.31$), while for the case of $N^*_{\mathrm{Re}} = 0.82$ the stagnation pressure dominates and the ellipse turns horizontal as it migrates to the channel center.

In this computation, a nonuniform grid with 82×97 nodes is adopted. As shown in Fig. 12, very reasonable

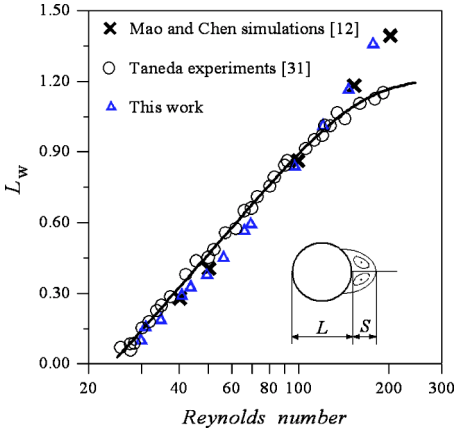

FIG. 11. Comparison of predicted wake length with experimental data ($L_w = S/L$).

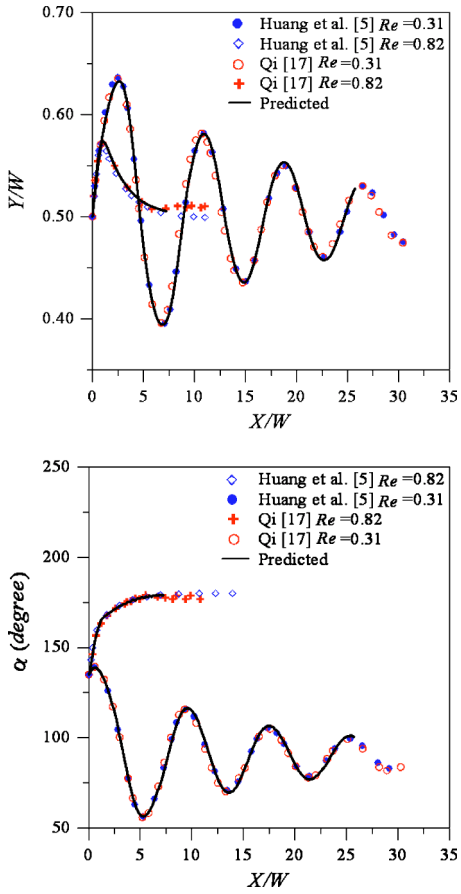

FIG. 12. Comparison of the predicted trajectory Y/W and orientation angle α of an ellipse with the results of Huang *et al.* [5] and Qi [17] (X is the displacement at the gravity or vertical direction and Y at the horizontal direction).

agreements of the particle orientation angles and trajectories between the mirror fluid method and the finite-element method by Huang *et al.* are found. Similar simulated results by the lattice-Boltzmann method [17] with a finer grid of 101×700 are also compared in Fig. 12.

IV. CONCLUSIONS AND FUTURE WORKS

We have presented a mirror fluid method for numerical simulation of solid-fluid flows. In this method we take the entire computational domain as a fixed Eulerian one without using a body-fitted coordinate system and assign suitable flow parameters to the solid particle domain by mirror relations to ensure the surface segment subjected to zero net shear and normal forces and the fluid-solid boundary conditions enforced implicitly; meanwhile, the motion of solid particles is calculated using a Lagrangian numerical scheme. This algorithm has the advantages of solving multiphase flow by a set of coupled equations, eliminating smear of density and other variables profiles across the interfaces, using a simple finite-difference method for discretization and adopting a fixed Cartesian grid without need of remeshing.

A number of numerical examples were studied in a two-dimensional coordinate system. The SIMPLE scheme and the control volume formulation are used for solving the governing equations for the motion of a rigid spherical or elliptic particle in a fluid. The Reynolds number, drag coefficient, and wake length of real spherical particles are well predicted as verified against the acknowledged experimental data. The orientation angle and trajectory of an ellipse settling in a fluid are also well predicted as compared with the reported simulation results. These numerical tests indicate that the mirror fluid method is simple, robust, and effective in simulating real solid-fluid flows.

The mirror fluid method can be extended to other more challenging particle flow problems with less restriction, especially when complicated and irregular solid-fluid interfaces are involved. In future work, we will use the mirror fluid method to compute the effects of irregular side walls, motion of other nonspherical (e.g., slender) solid particles with rotation or tilting in a Newtonian or non-Newtonian fluid. Extension to simulate three-dimensional solid-fluid flow at high Reynolds numbers will also be tested for further validity and application of this novel numerical algorithm.

ACKNOWLEDGMENTS

This work is supported by the National Natural Science Foundation of China (Nos. 20236050 and 20106016) and the National Key Basic Research and Development Program (No. 2004CB217604). The authors are also very grateful to Professor Jiayong Chen in our institute for his valuable support and advice.

[1] R. Clift, J. R. Grace, and M. E. Weber, *Bubbles, Drops, and Particles* (Academic Press, New York, 1978).
[2] J. Feng, H. H. Hu, and D. D. Joseph, J. Fluid Mech. **277**, 271 (1994).
[3] R. Glowinski, T.-W. Pan, T. I. Hesla, and D. D. Joseph, Int. J. Multiphase Flow **25**, 755 (1999).
[4] R. Glowinski, T. W. Pan, T. I. Hesla, D. D. Joseph, and J. Périaux, J. Comput. Phys. **169**, 363 (2001).
[5] P. Y. Huang, H. H. Hu, and D. D. Joseph, J. Fluid Mech. **362**, 297 (1998).
[6] P. Singh, D. D. Joseph, T. Hesla, R. Glowinski, and T.-W. Pan, J. Non-Newtonian Fluid Mech. **91**, 165 (2000).
[7] G. Ryskin and L. G. Leal, J. Comput. Phys. **50**, 71 (1983).
[8] G. Ryskin and L. G. Leal, J. Fluid Mech. **148**, 19 (1984).
[9] Z.-S. Mao, T. W. Li, and J. Y. Chen, Int. J. Heat Mass Transfer **44**, 1235 (2001).
[10] T. W. Li, Z.-S. Mao, J. Y. Chen, and W. Y. Fei, Chin. J. Chem. Eng. **10**, 1 (2002).
[11] Z.-S. Mao, Chin. J. Chem. Eng. **10**, 149 (2002).
[12] Z.-S. Mao and J. Y. Chen, Chin. J. Chem. Eng. **5**, 105 (1997).
[13] S. O. Unverdi and G. Tryggvason, J. Comput. Phys. **100**, 25 (1992).

[14] E. A. Fadlun, R. Verzicco, P. Orlandi, and J. Mohd-Yusof, J. Comput. Phys. **161**, 35 (2000).
[15] J. Kim, D. Kim, and H. Choi, J. Comput. Phys. **171**, 132 (2001).
[16] A. Gilmanov, F. Sotiropoulos, and E. Balaras, J. Comput. Phys. **191**, 660 (2003).
[17] D. Qi, J. Fluid Mech. **385**, 41 (1999).
[18] Z.-G. Feng and E. E. Michaelides, Int. J. Multiphase Flow **29**, 943 (2003).
[19] Z.-G. Feng and E. E. Michaelides, J. Comput. Phys. **202**, 20 (2005).
[20] T. Kajishima and S. Takiguchi, Int. J. Heat Fluid Flow **23**, 639 (2002).
[21] R. P. Fedkiw, T. Aslam, B. Merriman, and S. Osher, J. Comput. Phys. **152**, 457 (1999).
[22] S. Osher and R. P. Fedkiw, J. Comput. Phys. **169**, 463 (2001).
[23] R. P. Fedkiw, J. Comput. Phys. **175**, 200 (2002).
[24] R. Caiden, R. P. Fedkiw, and C. Anderson, J. Comput. Phys. **166**, 1 (2001).
[25] D. Q. Nguyen, R. P. Fedkiw, and M. Kang, J. Comput. Phys. **172**, 71 (2001).
[26] K. Yokoi, Phys. Rev. E **67**, 045701 (2003).
[27] C. Yang and Z.-S. Mao, Chin. J. Chem. Eng. **10**, 263 (2002).
[28] M. Sussman, P. Smereka, and S. Osher, J. Comput. Phys. **114**, 146 (1994).
[29] S. V. Patankar, *Numerical Heat Transfer and Fluid Flow* (Hemisphere, Washington, D.C., 1980).
[30] W. Mulder, S. Osher, and J. A. Sethian, J. Comput. Phys. **100**, 209 (1992).
[31] S. Taneda, J. Phys. Soc. Jpn. **11**, 1104 (1956).