Journal of Computational Physics 228 (2009) 1750–1769

Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

Smoothed profile method for particulate flows: Error analysis and simulations

Xian Luo, Martin R. Maxey, George Em Karniadakis *

Division of Applied Mathematics, Brown University, Providence, RI 02912, USA

ARTICLE INFO

Article history:
Received 11 March 2008
Received in revised form 30 September 2008
Accepted 5 November 2008
Available online 20 November 2008

Keywords:
SPM
Two-phase flow
Suspensions
Time-splitting scheme

ABSTRACT

We re-formulate and analyze a new method for particulate flows, the so-called "smoothed profile" method (SPM) first proposed in [Y. Nakayama, R. Yamamoto, Simulation method to resolve hydrodynamic interactions in colloidal dispersions, Phys. Rev. E 71 (2005) 036707], which uses a fixed computational mesh without conformation to the geometry of the particles. The method represents the particles by certain smoothed profiles to construct a body force term added into the Navier–Stokes equations. SPM imposes accurately and efficiently the proper conditions at the particle–fluid interface. In particular, while the original method employs a fully-explicit time-integration scheme we develop a high-order semi-implicit splitting scheme, which we implement in the context of spectral/*hp* element discretization. First, we show that the modeling error of SPM has a non-monotonic dependence on the time step size Δt; it is a function of $\sqrt{\nu\Delta t}/\xi$, where ν is the kinematic viscosity of fluid and ξ is the interface thickness of the smoothed profile. Subsequently, we present several steady and unsteady simulations, including flow past 3D complex-shaped particles, and compare against direct numerical simulations and the force coupling method (FCM).

1. Introduction

Particulate flow occurs in many natural and technological situations, such as colloidal sedimentation, polymer suspensions and lubricated transport. The methods for numerical simulation of particulate flows can be broadly divided into two categories: the first one involves a Lagrangian description in which the computational mesh follows the particles or the fluid; the second one employs an Eulerian description on a fixed grid with the interactions between the solvent and the particles represented by some form of body force or appropriate interaction equations. For example, the arbitrary Lagrangian–Eulerian (ALE) method [2,3] belongs to the first category, while the distributed Lagrange multiplier (DLM) method first proposed by Glowinski et al. [4,5] and further developed by Patankar et al. [6] falls into the second category.

We have studied extensively in the past the force coupling method (FCM) [7–10], which is also a method in the second category. It represents each particle in the flow by a low-order expansion of force multipoles, which is applied as a distributed body force on the flow. It captures the far-field solution accurately and can typically resolve body forces with better than a couple of percent accuracy, though it does not fully resolve the flow field close to the particle surfaces (in the pointwise sense).

One of the most widely used methods in the second category is the immersed boundary method (IBM) of Peskin, which was originally developed for flows with immersed flexible membranes [11,12]. The main idea of IBM is to employ a regular Eulerian mesh for the solvent over the entire domain, together with Lagrangian markers for the immersed boundary. The

* Corresponding author. Tel.: +1 401 863 1217; fax: +1 401 863 3369.
E-mail addresses: gk@cfm.brown.edu, gk@dam.brown.edu (G.E. Karniadakis).

0021-9991/$ - see front matter © 2008 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2008.11.006

immersed boundary moves with the local fluid velocity, and in return exerts a "singular" force to the nearby fluid. This force is usually obtained from the elastic properties of the immersed boundary, and then distributed into the surrounding fluid using a regularized Dirac delta function. IBM was later extended to flows with rigid bodies with certain treatment, e.g. in [13,14]. A new "formally" second-order scheme of IBM was proposed in [15], which has less numerical viscosity and might be more appropriate for high Reynolds number flows. In these references [13–15], the fluid–particle interaction force utilizes a feed-back formulation, namely the "virtual boundary method" proposed in [16]. An alternative expression for the force called "direct forcing" has been introduced to IBM by [17], which avoids introducing additional free parameters and alleviates the severe time step restrictions of the virtual boundary method. More recently, [18] proposed an improved direct forcing formulation which shows a smoother and less oscillatory boundary force for moving boundary problems. A similar approach to IBM called "immersed interface method" (IIM) has been proposed in [19], and it was applied to Stokes flow problems in [20]. In contrast to IBM adding the forcing function into the continuous governing equations, IIM incorporates the jump conditions across the immersed interfaces directly into the *discrete* equations. The IIM has been shown to be second-order accurate in the maximum norm for elliptic problems. Extension to incompressible Navier–Stokes equations has been presented in [21,22]. The entire vector or only the normal component of the singular force (from the immersed interface to the fluid) is incorporated into jump conditions for both velocity and pressure or only pressure, in [21] and [22], respectively. They both give sharp resolution of the pressure across the interface. Compared to [22], the method in [21] achieves better accuracy, but it is more complicated to implement in general.

More recently, a new method, the so-called "smoothed profile" method (SPM), was proposed in [1,23–28]; it falls into the second category as well, for it uses a fixed grid. It is conceptually similar to FCM in the sense that it represents the particles by certain smooth body forces in the Navier–Stokes equations instead of treating the particles as boundary conditions to the fluid. However, it has also features of IBM in that the force distribution effectively imposes constraints on the fluid motion that approximate the boundary conditions. As with FCM and IBM, SPM solves a single set of fluid dynamics equations in the entire domain including the particle volumes, without any internal boundary conditions. However, different to FCM and IBM, a smoothly spreading interface layer is used to represent the particle boundaries to have a transition from the rigid-body motion to the fluid motion. This treatment is similar to the method of PHYSALIS [29,30], which assumes a boundary layer to exist next to the particle surface, where the Stokes approximation is valid; PHYSALIS uses a local analytic solution of the Stokes equations near the particle. Both PHYSALIS and SPM smear out the sharp solid–fluid interface and replace it by a smooth layer, which should be sufficiently resolved by the spatial discretization.

The key point of SPM is to update the velocity inside each particle through the integration of a "penalty" body force to ensure the rigidity of the particles. It imposes the no-slip boundary condition implicitly without any special treatment on the solid–fluid boundaries. This feature is, of course, appealing as it corrects the near-particle behavior of FCM in an efficient manner.

SPM has been verified by several test problems, and has also been applied to the sedimentation of hundreds of 2D colloidal disks in [1,24]. More recently, the method was extended to multi-component fluids, such as charged colloids in electrolyte solutions in [25,26,31] and the electrophoresis of dense dispersions in [27,32]. However, this method lacks any rigorous theoretical foundation and it has only been presented as a fully-explicit scheme, hence limiting the size of time step to very small values. It has also been used in conjunction with uniform grids and simple particle shapes. To this end, we have worked on an error analysis of this method and, in particular, its semi-implicit extension that we propose in the current work. In addition, we critically evaluate SPM, for first time, against direct numerical simulations (DNS) and FCM solutions for a number of problems, including steady and unsteady particulate flows, simple- and complex-shaped particle evolutions. All the results we present in this paper are based on *three-dimensional* simulations.

The paper is organized as follows. In Section 2 we present the main steps of SPM in the context of high-order discretizations. Specifically, a third-order stiffly-stable splitting scheme is employed for temporal discretization and spectral elements for spatial discretization. In Section 3 we quantify the accuracy of SPM for several prototype flows. In Section 4 a further verification of the method is presented through comparisons with other methods for complex-shaped particles. Section 5 presents a SPM simulation of two interacting spheres in low Reynolds number. We conclude in Section 6 with a brief summary.

2. Formulation

2.1. Smoothed representation of particles

SPM represents each particle by a smoothed profile (or in other words an indicator/concentration function), which equals *unity* in the particle domain, *zero* in the fluid domain, and varies smoothly between one and zero in the solid–fluid interfacial domain. In [1], several *analytical* forms of the smoothed profile for spherical particles are presented. Among them, the following profile is mostly used:

$$\phi_i(\mathbf{x},t) = \frac{1}{2}\left[\tanh\left(\frac{a_i - |\mathbf{x}-\mathbf{R}_i|}{\xi_i}\right) + 1\right], \tag{1a}$$

where index i refers to the ith sphere, and a_i, $\mathbf{R}_i$, ξ_i are respectively the radius, the position vector, and the interface thickness parameter for the ith sphere. The indicator function ϕ_i is a function of spatial coordinates $\mathbf{x}$ and time t.

Here we extend the above formula to a more *general* form, which is effective for *any* particle shape:

$$\phi_i(\mathbf{x},t) = \frac{1}{2}\left[\tanh\left(\frac{-d_i(\mathbf{x},t)}{\xi_i}\right)+1\right], \tag{1b}$$

where $d_i(\mathbf{x},t)$ is the *signed* distance to the ith particle surface with positive value outside the particle and negative value inside the particle. The calculation of $d_i(\mathbf{x},t)$ is rather straightforward if particles are in simple shapes (boxes, spheres, etc.). If, however, we deal with more complex shapes which are represented by many surface points coordinates, a closest-point searching procedure should be used to find $d_i(\mathbf{x},t)$ and thus $\phi_i(\mathbf{x},t)$.

For other cases where we know the *analytical* form of the particle surface functions, or where we can construct an interpolating spline function based on the surface points, we propose to use a slightly different form of the indicator function due to its convenience and efficiency:

$$\phi_i(\mathbf{x},t) = \frac{1}{2}\left[\tanh\left(\frac{f_i(\mathbf{x},t)}{\xi_i}\tau_i\right)+1\right], \tag{2}$$

where $f_i(\mathbf{x},t)$ is the surface function of the ith particle, which is zero on the surface, positive inside the particle, and negative outside; here τ_i is a scaling factor. We implement this form in most of the numerical simulations of flows involving spherical, ellipsoidal and biconcave particles, which are presented in the next two sections.

A smoothly spreading *concentration field* is achieved by summing up the concentration functions of all the N_p non-overlapping particles:

$$\phi(\mathbf{x},t) = \sum_{i=1}^{N_p}\phi_i(\mathbf{x},t). \tag{3}$$

Fig. 1 shows a schematic of the smoothed profiles for an example involving two particles. The entire domain D can be subdivided into three domain types: D_{pi} is the particle domain of the i_{th} particle, where $\phi \sim 1.0$; D_f is the fluid domain, where $\phi \sim 0.0$; and D_s is the interface domain, where $0.0 < \phi < 1.0$.

Based on this concentration field, the *particle velocity field*, $\mathbf{u}_p(\mathbf{x},t)$, is constructed from the rigid motions of the N_p particles:

$$\phi(\mathbf{x},t)\mathbf{u}_p(\mathbf{x},t) = \sum_{i=1}^{N_p}\{\mathbf{V}_i(t)+\boldsymbol{\Omega}_i(t)\times[\mathbf{x}-\mathbf{R}_i(t)]\}\phi_i(\mathbf{x},t), \tag{4}$$

where $\mathbf{R}_i$, $\mathbf{V}_i = \frac{d\mathbf{R}_i}{dt}$ and $\boldsymbol{\Omega}_i$ are spatial positions, translational velocity and angular velocity of the ith particle, respectively. We can verify that the divergence of the particle velocity field is identically zero, i.e.,

$$\nabla\cdot\mathbf{u}_p = \sum_{i=1}^{N_p}\{\mathbf{V}_i(t)+\boldsymbol{\Omega}_i(t)\times[\mathbf{x}-\mathbf{R}_i(t)]\}\cdot\nabla\left(\frac{\phi_i}{\phi}\right) = 0,$$

where we have used: $\nabla(\frac{\phi_i}{\phi}) = \frac{(\nabla\phi_i)\phi-\phi_i(\nabla\phi)}{\phi^2} = \frac{\sum_{j=1}^{N_p}[(\nabla\phi_i)\phi_j-\phi_i(\nabla\phi_j)]}{\phi^2} = 0$, by the assumption of non-overlapping envelopes of different particles, i.e., $\phi_i(\nabla\phi_j) = 0$ for $\forall i\neq j$.

The *total velocity field* is then defined by a smooth combination of both the particle velocity field $\mathbf{u}_p$ and the fluid velocity field $\mathbf{u}_f$:

$$\mathbf{u}(\mathbf{x},t) = \phi(\mathbf{x},t)\mathbf{u}_p(\mathbf{x},t)+(1-\phi(\mathbf{x},t))\mathbf{u}_f(\mathbf{x},t). \tag{5}$$

We see that inside the particle domain ($\phi = 1$), we have $\mathbf{u} = \mathbf{u}_p$, i.e., the total velocity equals the particle velocity. In the interfacial domain ($0 < \phi < 1$) the total velocity changes smoothly from the particle velocity $\mathbf{u}_p$ to the fluid velocity $\mathbf{u}_f$.

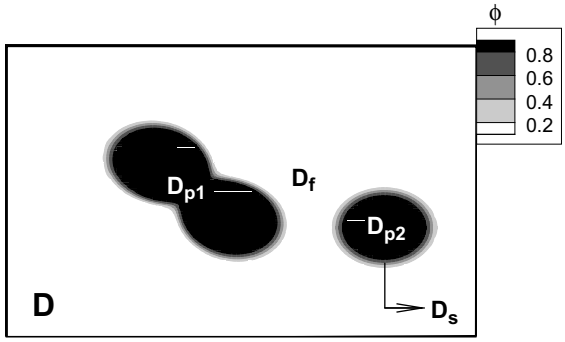


Fig. 1. An example of two particles: contours of the concentration field ϕ and computational domain partitioning ($D = \sum_{i=1}^{N_p} D_{pi} + D_f + D_s$).

SPM imposes indirectly the no-slip constraint on the particle boundaries. This imposition can be verified by taking the curl of Eq. (5):

$$\nabla \times \mathbf{u} = \nabla \times [\mathbf{u}_f + \phi(\mathbf{u}_p - \mathbf{u}_f)] = (\nabla\phi) \times (\mathbf{u}_p - \mathbf{u}_f) + \phi(\nabla \times \mathbf{u}_p) + (1-\phi)(\nabla \times \mathbf{u}_f).$$

As $\nabla\phi$ is non-zero in the interfacial domain and is perpendicular to the particle surface, the no-slip constraint can be expressed as $(\nabla\phi) \times (\mathbf{u}_p - \mathbf{u}_f) = 0$. So the imposition of the no-slip constraint comes from the assumption that the vorticity of the total flow field consists of the vorticities from both the fluid velocity and the particle velocity, i.e.,

$$\nabla \times \mathbf{u} = \phi(\nabla \times \mathbf{u}_p) + (1-\phi)(\nabla \times \mathbf{u}_f). \tag{6}$$

The validity of this equation will be checked later in a numerical simulation example.

SPM also imposes the no-penetration constraint on particle surfaces. If we check the divergence of the total velocity, taking into account that the fluid solvent is incompressible and the particle velocity field is divergence-free (derived by assuming non-overlapping particle envelopes), we obtain:

$$\nabla \cdot \mathbf{u} = \nabla \cdot [\mathbf{u}_f + \phi(\mathbf{u}_p - \mathbf{u}_f)] = (\nabla\phi) \cdot (\mathbf{u}_p - \mathbf{u}_f). \tag{7}$$

Since $(\nabla\phi) \cdot (\mathbf{u}_p - \mathbf{u}_f) = 0$ represents the no-penetration constraint on particle surfaces, imposing the incompressibility condition of the total velocity $\nabla \cdot \mathbf{u} = 0$ ensures the no-penetration surface condition, and vice versa.

SPM solves for the total velocity, $\mathbf{u}$, in the entire domain D, including the particle domain, using the Navier–Stokes equations with an extra force density term along with the incompressibility constraint:

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} = -\frac{1}{\rho}\nabla p + \nu\nabla^2\mathbf{u} + \mathbf{g} + \mathbf{f}_s \quad \text{in} \quad D, \tag{8a}$$

$$\nabla \cdot \mathbf{u} = 0 \quad \text{in} \quad D, \tag{8b}$$

where ρ is the density of the fluid, p is the pressure field, ν is the kinematic viscosity of the fluid, $\mathbf{g}$ is the gravity (and other external forces on the fluid), and $\mathbf{f}_s$ is the body force density term representing the interactions between the particles and the fluid. Here, the fluid solvent is assumed to be Newtonian with constant viscosity for simplicity.

The velocity and pressure fields are solved through the following two-step semi-discrete form proposed in [1,25]. First, SPM solves for an *intermediate* velocity and pressure fields as $\mathbf{u}^*, p^*$ from the previous step solution $\mathbf{u}^n$, by integrating the advection and viscous stress:

$$\mathbf{u}^* = \mathbf{u}^n + \int_{t^n}^{t^{n+1}} dt[-(\mathbf{u} \cdot \nabla)\mathbf{u} - \frac{1}{\rho}\nabla p^* + \nu\nabla^2\mathbf{u} + \mathbf{g}], \tag{9a}$$

$$\approx \mathbf{u}^n + \Delta t\left[-(\mathbf{u}^n \cdot \nabla)\mathbf{u}^n - \frac{1}{\rho}\nabla p^* + \nu\nabla^2\mathbf{u}^n + \mathbf{g}^n\right] \quad \text{in} \quad D, \tag{9b}$$

where Δt is the time increment $t^{n+1} - t^n$, and a forward Euler integration is used. This is solved in conjunction with the incompressibility constraint on $\mathbf{u}^*$, i.e., $\nabla \cdot \mathbf{u}^* = 0$, and hence $\frac{1}{\rho}\nabla^2 p^* = \nabla \cdot [-(\mathbf{u}^n \cdot \nabla)\mathbf{u}^n + \mathbf{g}^n]$.

Then SPM updates the total velocity and pressure fields from $\mathbf{u}^*, \mathbf{p}^*$:

$$\mathbf{u}^{n+1} = \mathbf{u}^* + \int_{t^n}^{t^{n+1}} dt\left[\mathbf{f}_s - \frac{1}{\rho}\nabla p_p\right] = \mathbf{u}^* + \left[\phi(\mathbf{u}_p - u^*) - \frac{\Delta t}{\rho}\nabla p_p\right] \quad \text{in} \quad D, \tag{10}$$

where p_p is the *extra pressure* field and $\mathbf{u}_p$ is calculated from Eq. (4). Note that the particle motions ($\mathbf{V}_i$ and $\mathbf{\Omega}_i$) could either be given or calculated from Newton's equations using the exerted forces on the particle. SPM assigns $\int_{t^n}^{t^{n+1}} \mathbf{f}_s dt = \phi(\mathbf{u}_p - u^*)$ to denote the momentum change (per unit mass) due to the presence of the particle. Thus at each time step the flow is corrected by a momentum impulse to ensure that the total velocity matches that of the rigid particles within the particle volume, and hence to enforce the rigidity constraint. From the above expression we have that $\mathbf{f}_s \sim \frac{1}{\Delta t}\phi(\mathbf{u}_p - u^*)$, suggesting that SPM is similar to penalty methods [33,34], which incorporate constraints into the governing equations. Here $1/\Delta t$ serves as a penalty parameter; thus the smaller Δt is, the tighter the control of the rigidity constraint is. However, in contrast to the delta functions used in [33,34], the smooth indicator function ϕ is used here by SPM.

The total pressure field is split into two parts as: $p = p^* + p_p$ to ensure the divergence-free of the total velocity field ($\nabla \cdot \mathbf{u} = 0$) and hence the imposition of the impermeable boundary condition on the particle boundaries. Specifically, the following equation for p_p:

$$\frac{\Delta t}{\rho}\nabla^2 p_p = \nabla \cdot [\phi(\mathbf{u}_p - \mathbf{u}^*)] \quad \text{in} \quad D, \tag{11}$$

is derived by taking the divergence of Eq. (10) and using the divergence-free constraints $\nabla \cdot \mathbf{u}^* = 0$, $\nabla \cdot \mathbf{u}^n = 0$ and $\nabla \cdot \mathbf{u}^{n+1} = 0$.

2.2. Semi-discrete system: temporal integration

The original SPM [1,25] used a fully-explicit time marching scheme for Eq. (9a). In the current paper, we introduce a semi-implicit treatment, using a stiffly-stable high-order *splitting* (velocity-correction) scheme [35], in order to enhance stability and increase temporal accuracy.

In the velocity-correction splitting scheme, the fields at time level n: $\{\mathbf{u}^n, p^n, \mathbf{R}_i^n, \mathbf{O}_i^n, \phi^n, \mathbf{V}_i^n, \mathbf{\Omega}_i^n, \forall i = 1,\ldots,N_p\}$, are advanced over a time step Δt to determine the fields $\{\mathbf{u}^{n+1}, p^{n+1}, \mathbf{R}_i^{n+1}, \mathbf{O}_i^{n+1}, \phi^{n+1}, \mathbf{V}_i^{n+1}, \mathbf{\Omega}_i^{n+1}, \forall i = 1,\ldots,N_p\}$, through the four substeps outlined below. Note that $\mathbf{R}_i^n, \mathbf{O}_i^n$ are position vector and orientation vector of the ith particle respectively, which contribute to the concentration function ϕ^n.

As a preliminary step before all substeps, we move particles from $\mathbf{R}_i^n, \mathbf{O}_i^n$ to $\mathbf{R}_i^{n+1}, \mathbf{O}_i^{n+1}$ by $\mathbf{V}_i^n, \mathbf{\Omega}_i^n$ using an explicit integration scheme, e.g. a high-order Adam–Bashforth method as

$$\mathbf{R}_i^{n+1} = \mathbf{R}_i^n + \Delta t \sum_{q=0}^{Je} a_q \mathbf{V}_i^{n-q}, \tag{12a}$$

$$\mathbf{O}_i^{n+1} = \mathbf{O}_i^n + \Delta t \sum_{q=0}^{Je} a_q \mathbf{\Omega}_i^{n-q}, \tag{12b}$$

where J_e is the order of the explicit time-integration scheme and a_q are the coefficients in the Adam–Bashforth scheme. Then, the concentration field ϕ^{n+1} is calculated.

2.2.1. Velocity update due to advection

We compute the intermediate velocity field $\mathbf{u}^s$ by first integrating the non-linear (advection) term and body force term:

$$\frac{\mathbf{u}^s - \sum_{q=0}^{J_e-1} \alpha_q \mathbf{u}^{n-q}}{\Delta t} = \sum_{q=0}^{J_e-1} \beta_q [-((\mathbf{u}\cdot\nabla)\mathbf{u})^{n-q} + \mathbf{g}^{n-q}] \quad \text{in} \quad D, \tag{13}$$

where α_q, β_q are the coefficients derived from the standard stiffly-stable scheme formulation (see [36]). Also, $\mathbf{u}^{n-q}$ and $\mathbf{g}^{n-q}$ are the velocity and body force fields at previous time steps.

2.2.2. Velocity update due to pressure influence

We then update the velocity from $\mathbf{u}^s$ to $\mathbf{u}^{ss}$ to account for the pressure influence:

$$\frac{\mathbf{u}^{ss} - \mathbf{u}^s}{\Delta t} = -\nabla p^* \quad \text{in} \quad D, \tag{14}$$

where the intermediate pressure field p^* is solved from:

$$\nabla^2 p^* = \nabla \cdot \left(\frac{\mathbf{u}^s}{\Delta t}\right) \quad \text{in} \quad D. \tag{15}$$

This equation is derived from Eq. (14) by imposing the incompressibility constraint $\nabla \cdot \mathbf{u}^{ss} = 0$, following the velocity-correction splitting scheme described in [36].

The following Neumann boundary condition for p^* is used at any velocity Dirichlet boundary:

$$\frac{\partial p^*}{\partial n} = \sum_{q=0}^{Je-1} \beta_q [-(\mathbf{u}\cdot\nabla)\mathbf{u} + \mathbf{g} - \nu\nabla\times(\nabla\times\mathbf{u})]^{n-q} \cdot \mathbf{n}, \tag{16}$$

assuming time-independent normal velocities at the boundary (see [36]).

2.2.3. Viscous effects

We obtain the intermediate velocity $\mathbf{u}^*$ based on $\mathbf{u}^{ss}$ by treating the viscous term implicitly:

$$\left(\nabla^2 - \frac{\gamma_0}{\nu\Delta t}\right)\mathbf{u}^* = -\frac{\mathbf{u}^{ss}}{\nu\Delta t} \quad \text{in} \quad D, \tag{17}$$

where γ_0 is the scaled coefficient of the stiffly-stable scheme with $\gamma_0 = \sum_{q=0}^{Je-1} \alpha_q$. Here, the velocity $\mathbf{u}^*$ on the boundary takes the same values prescribed as boundary conditions.

2.2.4. Velocity update due to rigid-body motion constraint

First, the hydrodynamic force and torque on the particles exerted by the surrounding fluid are derived from the momentum conservation between the particles and the fluid. In other words, the momentum change in the particle and interfacial domain equals the time integral of the hydrodynamic force and the external force. Hence, we assign

$$\mathbf{F}_{h_i}^n = \frac{1}{\Delta t}\int_D \rho\phi_i^{n+1}(\mathbf{u}^* - \mathbf{u}_p^n)d\mathbf{x} - \int_D \phi_i^{n+1}\rho\mathbf{g}d\mathbf{x}, \tag{18a}$$

$$\mathbf{N}_{h_i}^n = \frac{1}{\Delta t}\int_D \mathbf{r}_i^{n+1}\times[\rho\phi_i^{n+1}(\mathbf{u}^* - \mathbf{u}_p^n)] - \int_D \phi_i^{n+1}\mathbf{r}_i^{n+1}\times(\rho\mathbf{g})d\mathbf{x}, \tag{18b}$$

where $\mathbf{r}_i^{n+1}$ is the distance vector from the rotational reference point on the ith particle to any spatial point $\mathbf{x}$. Here, a first order approximation is used in the force integration.

Second, the particle translational and angular velocities are updated using Newton's equations:

$$\mathbf{V}_i^{n+1} = \mathbf{V}_i^n + M_i^{-1}\cdot\Delta t\cdot\sum_{q=0}^{Je} a_q\cdot(\mathbf{F}_{h_i}^{n-q} + \mathbf{F}_{ext_i}^{n-q}), \tag{19a}$$

$$\mathbf{\Omega}_i^{n+1} = \mathbf{\Omega}_i^n + I_i^{-1}\cdot\Delta t\cdot\sum_{q=0}^{Je} a_q\cdot(\mathbf{N}_{h_i}^{n-q} + \mathbf{N}_{ext_i}^{n-q}), \tag{19b}$$

where M_i and I_i are the mass and the moment of inertia of the i_{th} particle, a_q are the coefficients from a Adam–Bashforth scheme. Here, $\mathbf{F}_{ext_i}^{n-q}$ and $\mathbf{N}_{ext_i}^{n-q}$ are, respectively the external force and torque on the ith particle at previous time steps.

Third, the particle velocity field is calculated from:

$$\phi^{n+1}\mathbf{u}_p^{n+1} = \sum_{i=1}^{N_p}\phi_i^{n+1}\{\mathbf{V}_i^{n+1} + \mathbf{\Omega}_i^{n+1}\times[\mathbf{x} - \mathbf{R}_i^{n+1}]\}. \tag{20}$$

For free motion of particles, $\mathbf{u}_p$ is coupled with the forces and torques on particles $\mathbf{F}_{hi}, \mathbf{F}_{exti}, \mathbf{N}_{hi}$ and $\mathbf{N}_{exti}$. To decouple the system, we employ Eqs. (18),(19) and (20) using an explicit numerical scheme.

Fourth, we solve for the extra pressure field p_p due to the rigidity of the particle as

$$\nabla^2 p_p = \gamma_0\nabla\cdot\left(\frac{\phi^{n+1}(\mathbf{u}_p^{n+1} - \mathbf{u}^*)}{\Delta t}\right) \quad \text{in} \quad D. \tag{21a}$$

The following is used as the boundary conditions for p_p at any velocity Dirichlet boundary (with time-independent normal velocities):

$$\frac{\partial p_p}{\partial n} = \frac{\gamma_0\phi^{n+1}(\mathbf{u}_p^{n+1} - \mathbf{u}^*)}{\Delta t}\cdot\mathbf{n}. \tag{21b}$$

Finally, we update the total velocity field by enforcing the rigid-body motion constraint using the particle velocity field $\phi^{n+1}\mathbf{u}_p^{n+1}$:

$$\frac{\gamma_0\mathbf{u}^{n+1} - \gamma_0\mathbf{u}^*}{\Delta t} = \frac{\gamma_0\phi^{n+1}(\mathbf{u}_p^{n+1} - \mathbf{u}^*)}{\Delta t} - \nabla p_p \quad \text{in} \quad D. \tag{22}$$

Eqs. (21a) and (21b) are both derived from Eq. (22). The total pressure field is calculated by summing the intermediate pressure and the extra pressure field, i.e., $p^{n+1} = p^* + p_p$.

2.3. Fully-discrete system: spatial discretization

For spatial discretization of the above equations, we apply the spectral/*hp* element method (see [36]). This hybrid method benefits from both finite element and spectral discretization: on one hand, for domains with complex geometry, we can increase the number of subdomains/elements (h-refinement) with the error in the numerical solution decaying algebraically. On the other hand, with fixed elemental size we can increase the interpolation order within the elements (p-refinement) to achieve an exponentially decaying error, provided the solutions are sufficiently smooth throughout the domain. Hence, the use of smoothed profiles in SPM preserves the high-order numerical accuracy of the spectral/*hp* element method.

The solution domain is partitioned into non-overlapping sub-domains/elements (with characteristic size of an element to be h): $\Omega = \bigcup_{e=1}^{N_{el}}\Omega^e$. We use hexahedral or tetrahedral elements in our present numerical simulations. By constructing the standard element Ω^{st}, the local coordinates in the standard element ($\zeta\in\Omega^{st}$) can be mapped to the global coordinate in any elemental domain ($\mathbf{x}\in\Omega^e$) by an isoparametric transformation $\mathbf{x} = \chi^e(\zeta)$. Then, a polynomial expansion (with polynomial order up to P) is employed within the standard element to construct the approximate solution u^δ:

$$u^\delta(\mathbf{x}) = \sum_{e=1}^{N_{el}}\sum_{j=1}^{P}\hat{u}_j^e\psi_j^e(\zeta) = \sum_{i=0}^{N_{dof}}\hat{u}_i\Psi_i(\mathbf{x}), \tag{23}$$

where $\zeta = [\chi^e]^{-1}(\mathbf{x})$ is the local coordinate, $\psi_j^e(\zeta)$ denotes the local expansion modes, and $\Psi_i(\mathbf{x})$ are the global modes derived from the global assembly of the local modes [36].

For the expansion basis $\psi_j^e(\zeta)$, both *modal* and *nodal* basis can be used. A modal expansion is used in conjunction with Galerkin projection to solve Eqs. (15), (17) and (21a), while a nodal expansion is used in conjunction with collocation projection of Eq. (13). The updating step (22) could be done by either collocation or Galerkin projection. The former is more efficient but the numerical solution might be discontinuous at some element boundaries, whereas the latter guarantees C^0 continuity of the approximate solution across the elements through an appropriate choice of the boundary modes. For the modal expansion, we use a semi-orthogonal basis written in terms of the Jacobi polynomials. For the nodal basis, we employ the Lagrange–Jacobi polynomials through the Gauss–Legendre–Lobatto quadrature points.

The spectral/*hp* element method allows us to accurately represent arbitrary fixed rigid boundaries of the flow domain while using SPM to represent the particles.

3. Prototype flows: error analysis and verification

In order to verify SPM and analyze its error, we test several prototype flow problems.

3.1. Planar Couette flow

We simulate the steady laminar flow of a viscous fluid between two parallel plates, one moving relatively to the other. The plates are separated by a distance h in the y-direction with the lower plate aligned with the x-axis and being stationary while the upper plate moving with constant velocity $\mathbf{V} = (V, 0, 0)$. At first, we want to see how accurately SPM captures the exact analytical solution $\mathbf{u}_{exact} = (V \cdot y/h, 0, 0)$ for $y \in [0, h]$ and $(V, 0, 0)$ for $y \in [h, 2h]$; here we specify $V = 1, h = 3$.

Specifically, using this problem, we want to analyze SPM using two different time-discretization implementations: one is the original fully-explicit scheme [1,25], while the other is the semi-implicit scheme we proposed in the previous section. The fully-explicit SPM uses two steps as in Eqs. (9a) and (10), while the semi-implicit SPM uses four steps given by Eqs. (13), (14), (17) and (22). The top wall ($h \leqslant y \leqslant 2h$) is treated by a "smoothed profile", with the concentration field of $\phi(\mathbf{x}, t) = \frac{1}{2}\left[\tanh\left(\frac{y-h}{\xi}\right) + 1\right]$ derived from Eq. (1b). The simulation domain D is a box of $h/3 \times 2h \times h/3$, with periodic boundaries in the x- and z-directions and Dirichlet boundaries in the y-direction (i.e., $\mathbf{u}|_{y=0} = (0, 0, 0), \mathbf{u}|_{y=2h} = \mathbf{V} = (1, 0, 0)$). We ignore both the gravity and the advection $(\mathbf{u} \cdot \nabla)\mathbf{u}$ in both implementations. The velocity and pressure fields are solved by a first-order temporal scheme by marching through pseudo-time to the steady state. The spectral/*hp* element method is used for spatial discretization, with six non-uniform elements and a polynomial order of $P = 11$. Upon convergence to steady state, we calculate the L_2 error ($L2_{er}$) of the streamwise velocity u in the flow region $0 \leqslant y \leqslant h$.

We can also obtain solutions directly from the semi-discrete equations by assuming that: (1) all the field quantities are only y-dependent, i.e., $\mathbf{u}(x, y, z) = \mathbf{u}(y), p(x, y, z) = p(y)$, etc.; (2) the velocity field has a nontrivial component only in the streamwise direction, i.e., $\mathbf{u}(y) = (u(y), 0, 0)$; (3) steady state is achieved, i.e., $u^{n+1} = u^n$; and (4) the pressure is uniform everywhere, i.e., $\nabla p^* = (0, 0, 0), \nabla p_p = (0, 0, 0)$. Then for the *fully-explicit* implementation, from (9a) and (10), we obtain the 1D reduced system for the streamwise velocity u^n and the streamwise intermediate velocity u^*:

$$\begin{cases} (1-\phi)\nu\Delta t \frac{\partial^2 u^n}{\partial y^2} - \phi u^n = -\phi u_p, \\ u^* = u^n + \nu\Delta t \frac{\partial^2 u^n}{\partial y^2}. \end{cases} \tag{24}$$

Applying the same assumptions to the *semi-implicit* scheme, a *different* reduced system for u^n and u^* is derived from Eqs. (13), (14), (17) and (22):

$$\begin{cases} \nu\Delta t \frac{\partial^2 u^*}{\partial y^2} - \phi u^* = -\phi u_p, \\ u^n = u^* + \phi(u_p - u^*). \end{cases} \tag{25}$$

From the above reduced systems, for both schemes we obtain the velocity to be: $u^n = u_p$ when $\phi = 1$ (particle domain) and $u^n = u^* = ay$ when $\phi = 0$ (fluid domain), where the constant a needs to be determined by matching the velocity through the interface domain. In the limit of employing a *step function* for ϕ (i.e., $\xi = 0$), and by imposing C^0 continuity of u^n at the interface $y = h$, we find that $a = y/h$, which means we recover the exact solution using both schemes. However, if ϕ is a smooth function with $\xi > 0$, the above two reduced systems are solved by a second-order finite difference method using a fine uniform grid (with 4001 grid points and mesh size $\Delta y = 2h/4000$).

Furthermore, in both 1D systems (24) and (25), an artificial boundary layer (scaling as $\sqrt{\nu\Delta t}$) is present. In particular, both terms $\exp(\pm y/\sqrt{\nu\Delta t})$ will be present in the velocity solution and this will lead to a non-monotonic error behavior. Our results from solving the full 3D system further verify this.

Fig. 2 shows that the L_2 error in both implementations has a non-monotonic dependence on the time step Δt, similar in fact to the error structure of semi-Lagrangian schemes [37]. We note that the spatial discretization error is negligible since increasing the polynomial order in each element from $P = 11$ to $P = 13$ showed no change in the L_2 error. The true temporal discretization error should also vanish for the high-order splitting scheme (see [36]). The agreement of the 4-step full 3D system results with the 4-step reduced 1D system results in Fig. 2 indicates a negligible error of the pressure solver. So the L_2 error shown in the figures consists of the SPM modeling error from the smoothed particle representation in the

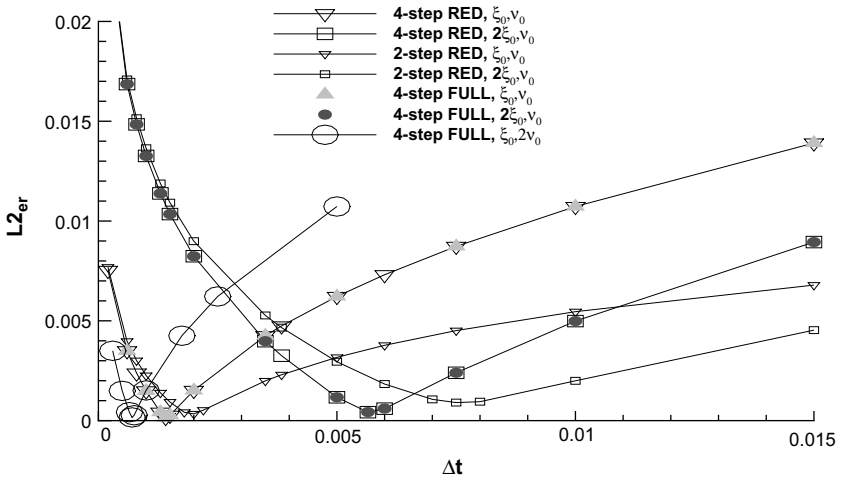


Fig. 2. Couette flow: L_2 error of the streamwise velocity versus Δt for two different interface thickness ξ and two different kinematic viscosity v. (LEGEND NOTE: '4-step': 4-step semi-implicit scheme; '2-step': 2-step full-explicit scheme; 'RED': reduced 1D system discretized by finite difference; 'FULL': full 3D system discretized by spectral/hp element; 'ξ_0': $\xi_0 = 0.05 = 0.0167h$; 'v_0': $v_0 = 1.0$.)

Navier–Stokes equation by Eqs. (9a) and (10). Simple scaling arguments of the 1D systems (24) and (25) show that the error is, in fact, a function of $\sqrt{v\Delta t}/\xi$, and the results from the solution of the full 3D system shown in Fig. 3 verify this.

Fig. 3 also indicates that the optimum time step size $(\Delta t)_o$ depends only on $\sqrt{v}/\xi$, i.e., for the 2-step fully-explicit scheme $\sqrt{v(\Delta t)_o}/\xi \sim 0.88$, while for the 4-step semi-implicit scheme $\sqrt{v(\Delta t)_o}/\xi \sim 0.75$. A physical explanation of this error behavior can be put forward by considering the following two aspects. On the one hand, recall that SPM essentially adds a penalty term $\mathbf{f}_s \sim \phi(\mathbf{u}_p - \mathbf{u}^*)/\Delta t$ in the Navier–Stokes equations. Hence, smaller Δt would have a tighter control of the flow and thus achieve better accuracy. But on the other hand, physically SPM simulates the particle by impulsively applying a momentum change to the particle domain at each time step, which in turn creates an impulse on the flow as in the Stokes' first problem. The thickness of the diffusive layer (within which the impulsive momentum change has an effect) is thus scaled as $\delta = 2.76\sqrt{v\Delta t}$ [38]. So a smaller Δt will produce a thinner Stokes layer thickness δ; in particular, if Δt is so small that δ cannot be spatially resolved by the smoothed profile used, it will lead to an increase in error. Let us introduce the "effective" smooth interface thickness l_e, which is expected to be scaled as 2ξ. The optimum time step size $(\Delta t)_o$, which produces the smallest error, comes from the balance of the Stokes layer scale δ and the effective interface thickness l_e. The numerical results in Fig. 3 verify this by showing that $l_e = 2.07\xi$ for the 4-step semi-implicit implementation. But for the 2-step fully-explicit scheme, a larger $l_e = 2.42\xi$ is indicated. By treating the viscous term implicitly in the 4-step scheme, we have a stricter control of the viscous layer throughout the interface domain, thus the intermediate velocity $\mathbf{u}^*$ is more sharply defined and hence the effective smooth interface thickness l_e is thinner.

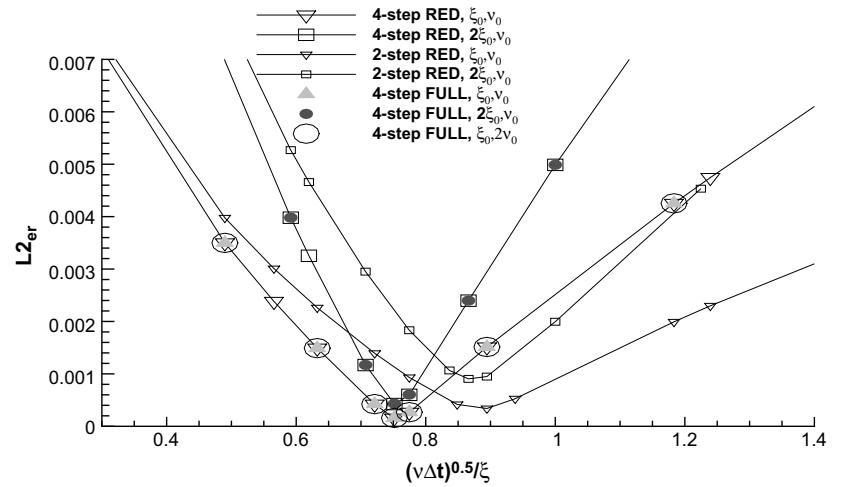


Fig. 3. Couette flow: L_2 error of the streamwise velocity versus $\sqrt{v\Delta t}/\xi$ for the same SPM simulation results as shown in Fig. 2.

3.2. Periodic array of spheres

Next we apply SPM to another benchmark problem describing the sedimentation of spheres (of radius a) arranged in a periodic cubic lattice (of length L). The particle concentration c is defined by the ratio of the particle volume (Ω_p) over the domain volume (Ω)) : $c = \Omega_p/\Omega = \frac{4\pi a^3}{3}/L^3$. For low c, Hasimoto [39] has derived a relationship between the external force ($\mathbf{F}$) on each sphere and the sedimentation velocity ($\mathbf{V}$) as follows:

$$\mathbf{V} = \left(1 - 1.7601c^{\frac{1}{3}} + c - 1.5593c^2 + O\left(c^{\frac{8}{3}}\right)\right)\frac{\mathbf{F}}{6\pi a\rho\nu}, \tag{26}$$

where $\frac{\mathbf{F}}{6\pi a\rho\nu}$ is the Stokes velocity of a particle settling in a unbounded fluid under the external force $\mathbf{F}$.

The simulation here adopts a configuration of $L = 12$ and $a = 2$. Hence, $c = 0.0194$, and Eq. (26) can be used to compare with the settling velocity results from the numerical simulations. In order to maintain the system in equilibrium, a uniform pressure gradient is applied to balance the net force exerted on the spheres: $\nabla p \cdot \Omega = -\mathbf{F}$. The net volume flux is zero in this frame of reference. Alternatively, if we change the reference frame to move with the particle (with velocity $\mathbf{V}$), the particles can be viewed as a fixed bed flushed by fluid driven by a pressure gradient. The superficial velocity is defined by the net displacement or volume flux of flow throughout the domain D:

$$\mathbf{V_s} = \frac{1}{\Omega}\int_D \mathbf{u}d\mathbf{x}, \tag{27}$$

where $\mathbf{u}$ is the flow velocity at any spatial point $\mathbf{x}$. As this reference frame is moving with velocity $\mathbf{V}$, from the zero net volume flux under the original fixed frame, we have $\Omega\mathbf{V_s} + \Omega\mathbf{V} = 0$, thus $\mathbf{V_s} = -\mathbf{V}$, i.e., the superficial velocity in the second reference frame equals the negative settling velocity in the first reference frame. This, in fact, is the result of Galilean invariance, which we have verified also numerically.

In addition to the settling velocity, we want to examine the accuracy of SPM in the *pointwise* sense with comparisons against *well-resolved* Direct Numerical Simulations (DNS) using the spectral/*hp* element code [40], and also against the force coupling method (FCM) with spectral/*hp* element discretizations as presented in [7,10]. For the SPM simulation, flows in both reference frames are resolved with an identical grid of 1728 uniform hexahedral elements and $P = 7$ polynomial order. Also, the same uniform pressure gradient $\nabla p = (-5.88 \times 10^{-4}, 0, 0)$ is applied in both reference frames. Within the first reference frame, an external force $\mathbf{F} = -\nabla p \cdot \Omega$ is applied on each sphere, and the sedimentation velocity $\mathbf{V}$ is calculated. In the second reference frame, the particle velocity is zero, and the superficial velocity $\mathbf{V}_s$ is calculated.

Both DNS and FCM simulations are performed to resolve the flow in the second reference frame. DNS simulation applies a pressure gradient of $\nabla p\prime = \nabla p \cdot \Omega/\Omega_f$ to ensure the same value of the drag force, and hence the same flow problem as in SPM and FCM simulations is solved. A non-uniform mesh of 11,232 tetrahedral elements and $P = 7$ polynomial order is used. FCM simulation uses a penalty method to calculate the external force in order to fix the particle in position as proposed in [41]. The same pressure gradient ∇p and the same mesh as in the SPM simulation is used for FCM.

Fig. 4 shows the streamwise velocity profiles along the cross-flow direction (i.e., along y-axis) using DNS, FCM and SPM, all in the second reference frame. The agreement of SPM and DNS is very good everywhere; SPM has a similar error to FCM in the far-field, but SPM resolves the near-field velocity much better. This is due to the rigid-body constraint imposition of SPM, which is not present in the FCM implementation. Also the errors of the superficial velocity for FCM and SPM simulations are very similar; both are around 0.05%.

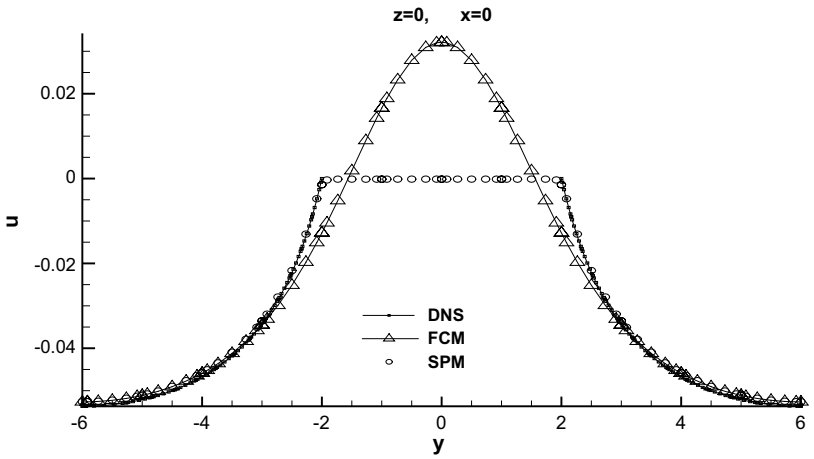


Fig. 4. Periodic array of spheres: streamwise velocity u along the center line y-axis, using DNS, FCM and SPM (with $\nu = 0.34, \xi = 0.1 = 0.05a, \Delta t = 0.015$).

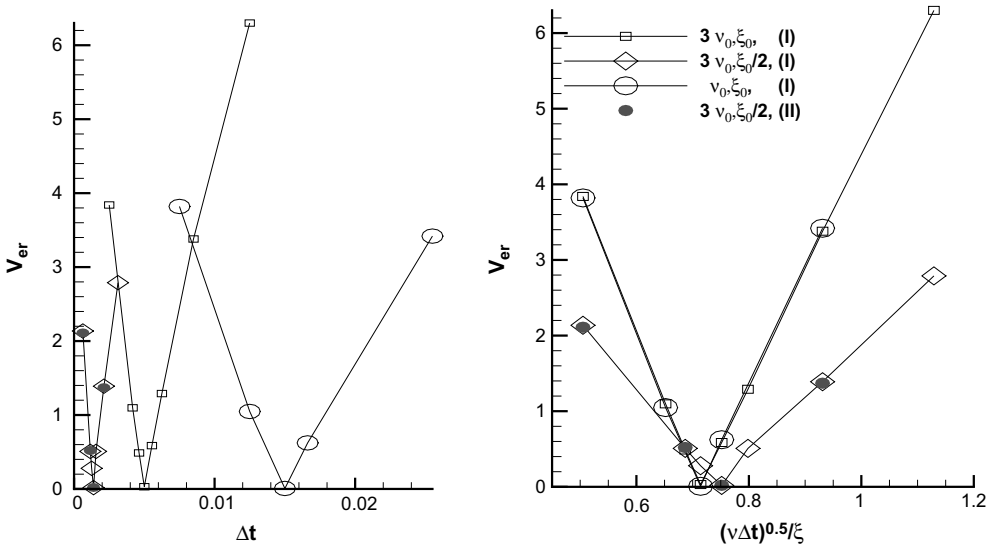


Fig. 5. SPM for a periodic array of spheres: percentage error in settling velocity V_{er} versus time step Δt (left) and versus $\sqrt{\nu\Delta t}/\xi$ (right), for two values of viscosity ν and for two values of the interface thickness ξ. (LEGEND NOTE: 'ν_0': $\nu_0 = 0.34$; 'ξ_0': $\xi_0 = 0.1 = 0.05a$; '(I)': simulation done in the first reference frame to calculate $\mathbf{V}$; '(II)': simulation done in the second reference frame with fixed spheres, $-\mathbf{V}_s$ calculated to substitute $\mathbf{V}$).

The error plots in Fig. 5 confirm the conclusions of the last subsection, namely that the SPM modeling error is a function of $\nu\Delta t$ and ξ, and the optimum time step size $(\Delta t)_o$ is determined by the balance of the Stokes layer $\delta = 2.76\sqrt{\nu\Delta t}$ and the effective smooth interface thickness l_e. We find from the above data that $l_e \sim 2.07\xi$ for sufficiently small $\xi(\xi = 0.025a)$, which is the same as the result we obtained for the Couette problem! (We note that l_e is slightly smaller $(l_e \sim 1.97\xi)$ if we employ a larger value of ξ, e.g. 0.05a.) Fig. 5 also verifies that the relative difference between the absolute values of the superficial velocity $\mathbf{V}_s$ and the settling velocity $\mathbf{V}$ is negligible.

3.3. Rotating sphere at low Reynolds number

Next we investigate the accuracy of SPM in simulating the steady flow induced by a sphere rotating with constant angular velocity in a viscous incompressible fluid which is at rest at large distances from the sphere. This problem in an open domain has been treated analytically by Bickley [42], who obtained a second-order approximation for the fluid velocity and pressure, which is quite accurate when the Reynolds number is small. We use this open domain solution as a reference to validate our numerical simulation results for a finite domain, at Reynolds number less than or equal to 8.0.

The computational domain is $[-\pi,\pi]\times[-\pi,\pi]\times[-\pi,\pi]$, with a sphere of radius $a = 0.261$ centered at the origin and spinning with an angular velocity of $\boldsymbol{\omega} = (0,0,\omega_z)$. We use this configuration of $L/2a = 2\pi/2a = 12.036$ with periodic treatment on domain boundaries in all three directions to simulate the unbounded flow domain. By the definition of the rotation Reynolds number $Re_w = \omega_z a^2/\nu$, we set $\omega_z = 10.01$ and simply change the fluid kinematic viscosity ν to obtain different values of Reynolds number, from 0.25 to 8.

The required external torque T_{ext} on the sphere to achieve a Reynolds number Re_w is estimated from the asymptotic approximation obtained by Takagi [43]:

$$2T_{ext}/\rho a^5\omega_z^2 = 16\pi(1+f(Re_w))/Re_w,$$

This is a correction to Lamb's result $M = 16\pi/Re_w$ for Stokes flow [44], and has been confirmed by numerical simulations of Dennis [45] for $Re \leqslant 10$. Numerical results of the hydrodynamic torque obtained from Eq. (18b) are compared to this given external torque.

For the SPM simulations we used 4096 nonuniform hexahedral elements and a polynomial order of $P = 5$. The following indicator function, utilizing the form of (2), was applied:

$$\phi(\mathbf{x},t) = \frac{1}{2}\left[\tanh\left(\frac{f(\mathbf{x},t)}{\xi}\tau\right)+1\right], \quad \text{with} \quad f = 1-|\mathbf{x}|^2/a^2, \quad \tau = a/2. \tag{28}$$

Compared to the original spherical envelope ((1)), this envelope is slightly different and thus the effective smooth interface thickness l_e is slightly different from 2.07ξ obtained for ((1)). Let us define the effective interface thickness l_e to be twice the distance from the particle surface to the spatial point whose concentration function is $\phi_e = \left(\tanh\left(\frac{-2.07/2\xi}{\xi}\right)+1\right)/2 = 0.112$. For the original envelopes ((1)), with $\xi = 0.05a = 0.013$, $l_e = 2.07\xi = 0.0269$; with the new envelope (28), using the same ξ, we have $l_e = 2\times 0.0127 = 0.0254$. The time step size is thus chosen according to the expected optimum time step derived

from $2.76\sqrt{\nu(\Delta t)_o} = l_e$. DNS results were obtained using 6600 hexahedral elements with polynomial order $P = 7$. The FCM results from [46], which were obtained by a Fourier pseudo-spectral method with 64^3 grid points for the same domain, are also included for comparison.

We see from Fig. 6 that SPM captures the primary azimuthal velocity quite well, with a maximum pointwise error of about 3.2% near the sphere surface (with $\xi = 0.05a$). This error goes up to about 4.9% if we use a thicker smooth interface, $\xi = 0.077a$. The radial velocity is more challenging to resolve, for it comes from the secondary flow at finite Reynolds numbers and its magnitude is much smaller than the dominant azimuthal velocity. Fig. 7 shows that there is some effect due to the periodic boundary conditions as manifested by the difference between the SPM results and the second-order approximate solution [42]. The relative difference in the azimuthal velocity is around 0.00063 on the domain boundaries. Thus, we employ DNS results for a more fair comparison, and the agreement with SPM is good everywhere. We also see that SPM is resolving the radial velocity much better than FCM near the particle surface ($x/a < 1.3$), and it resolves the far-field velocity as well as FCM. By reducing the SPM interface thickness parameter from $\xi = 0.077a$ to $\xi = 0.05a$, the peak pointwise error in the radial velocity reduces from about 8.5% to 3.6% for Reynolds number 0.25.

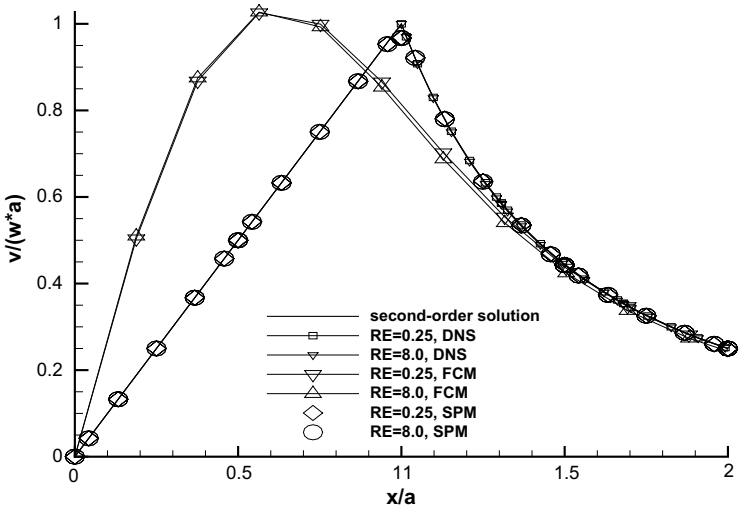


Fig. 6. Rotating sphere: non-dimensional *azimuthal* velocity v versus radial distance along the x-axis. Comparison between the second-order approximation solution and results from DNS, FCM and SPM (with $\xi = 0.05a$) for two different Reynolds numbers.

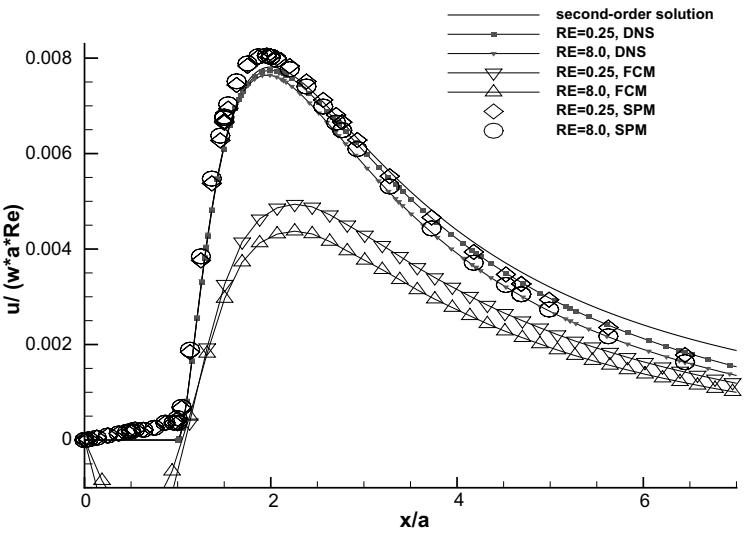


Fig. 7. Rotating sphere: Non-dimensional *radial* velocity u versus radial distance the along x-axis. Comparison between the second-order approximation solution and results from DNS, FCM and SPM (with $\xi = 0.05a$) for two different Reynolds numbers.

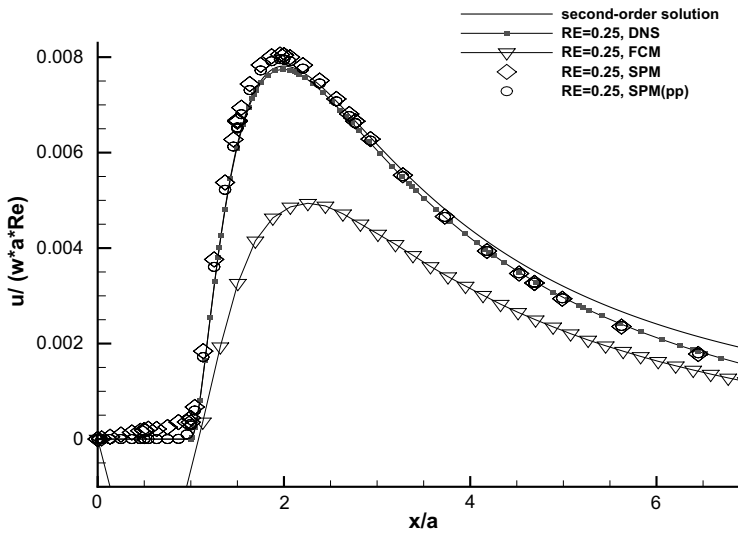


Fig. 8. Rotating sphere at Reynolds number 0.25: non-dimensional *radial* velocity u versus radial distance along the x-axis. Comparison between the second-order approximation solution and results from DNS, FCM, SPM (with $\xi = 0.05a$). (LEGEND NOTE: 'SPM(pp)': SPM with p_p modification step as in Eq. (29).)

With the above SPM setup, we now check the indirect imposition of the *no-slip constraint* on the sphere boundary by validating Eq. (6) on the equator plane $z = 0$. The right hand side (RHS) is calculated using ϕ from (28), $\mathbf{u}_p = \boldsymbol{\omega} \times \mathbf{x}$ and $\mathbf{ui}_f = (\boldsymbol{\omega} \times \mathbf{x})a^3/|x|^3$ which is the exact Stokes solution for an open domain. The left hand side (LHS) is computed by taking the numerical gradient of the total velocity. We obtain the relative error between RHS and LHS (in L2-sense) to be around 1.1%. By noting that this error mainly comes from the periodic boundary treatment instead of a open domain, and this problem features in sharp transition of the vorticity across the interface, the above test serves as a sufficient verification of the imposition of no-slip boundary constraint in SPM.

Fig. 7 also shows a non-zero radial velocity inside the particle, which is of the same order of magnitude as the peak error in the radial velocity. This particle-penetrating phenomenon arises from the final updating step Eq. (22), where ∇p_p has a non-zero value even inside the particle. In order to suppress this erroneous velocity we modify (22) to:

$$\gamma_0 \frac{\mathbf{u}^{n+1} - \mathbf{u}^*}{dt} = \gamma_0 \frac{\phi(\mathbf{u}_p^{n+1} - \mathbf{u}^*)}{dt} - (1 - \phi)\nabla p_p. \tag{29}$$

We see from Fig. 8 that the result is satisfying, as this modification ensures the non-penetrating constraint is satisfied strictly inside the particle as well and this has as a result a further reduction in the error of the radial velocity outside the particle. However, this modification introduces some compressibility inside and on the particle boundary (i.e., $\phi \neq 0$), see Eqs. (21a) and (29). Indeed, the numerical results show that the divergence of the total velocity $\nabla \cdot \mathbf{u}^{n+1}$ is slightly larger than the divergence of the intermediate velocity $\nabla \cdot \mathbf{u}^*$ inside the particle, however, both are of comparable magnitude.

3.4. Stokes oscillating plate

Next we study an *unsteady* problem in order to evaluate the temporal discretization error and how it compares to the modeling error inherent in SPM. To this end, we consider the problem of an infinite plate oscillating in an unbounded fluid. The simulation domain is $[0,1] \times [0,6] \times [0,1]$, where $y \in [0,3]$ is the solid plate modeled by SPM and the fluid flow is in the interval $y \in [3,6]$. The plate moves with an oscillating velocity $\mathbf{V} = (\sin(\omega \mathbf{t}), \mathbf{0}, \mathbf{0})$. The exact solution given in [47] is

$$\mathbf{u}_{ex} = (e^{-Y/\sqrt{2}} \sin(T - Y/\sqrt{2}), 0, 0),$$

where $Y = \frac{y-3}{(\nu/\omega)^{1/2}}$ and $T = \omega t$ for our case. We treat the boundary at $y = 6$ as a Dirichlet boundary using the exact solution; periodicity is imposed along both the x- and z-directions.

We calculate the L_2 error of the streamwise velocity in the domain ($y \in [0,3]$) at each time step and then calculate the mean error after convergence (to a constant-amplitude *sine* function in time). Here, we specify the fluid kinematic viscosity as $\nu = 1$ and the oscillating frequency parameter to be $\omega = 1$. SPM simulations are performed on a mesh of 6 non-uniform hexahedral elements with $P = 7$ polynomial order. Different orders ($J_e = 1, 2$ or 3) in time integration for solving the intermediate velocity $\mathbf{u}^*$ are employed.

From Fig. 9, we see that indeed higher time-integration order J_e leads to smaller errors beyond the optimum time step size where the temporal discretization error dominates over the SPM modeling error. This behavior, in fact, confirms that the

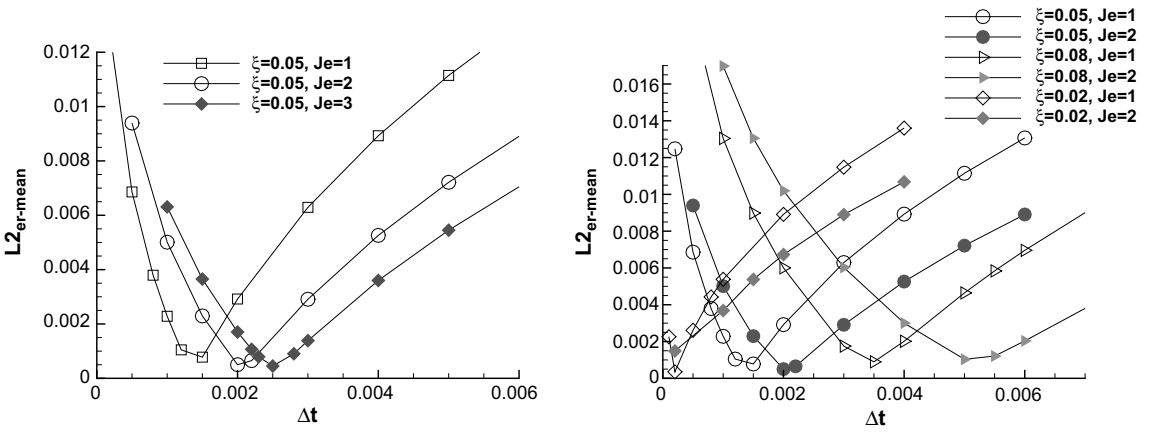


Fig. 9. Oscillating plate: Mean L_2 error ($L2_{er-mean}$) versus time step size Δt for different order J_e (left) and different interface thickness parameter ξ (right).

temporal error has two contributions: (1) the SPM modeling error, which is a function of $v\Delta t$ and ξ, and (2) the discretization error from the time-integration scheme we use for the intermediate velocity field solving Eqs. (13)–(15) and (17); the second error scales as $(v\Delta t)^{Je}$ for unsteady problems (see details in [35]). With regards to the effect of the profile thickness ξ, our results suggest that larger thickness ξ and higher order *Je* would lead to larger optimum time step $(\Delta t)_o$, which is clearly advantageous in production simulations.

4. Flow past complex-shaped particles

So far we have modeled particles with simple shapes but in this section we employ the new general envelopes (Eq. (2)) to model flow past complex-shaped particles. SPM is advantageous in that it does not require complex meshes around the particle surfaces, as the computational domain is the entire domain including particle domains. The way it represents the complex-shaped particles is rather straightforward. As long as we know the surface function of the particles or we have the coordinates of a sufficient number of points on the particle surfaces, we can obtain the concentration field ϕ representing the particles, and then solve the governing equations for the flow. In the following we examine the accuracy of SPM for ellipsoidal and biconcave particles.

4.1. Stokes flow past an ellipsoid

We first study ellipsoidal particles, specifically spheroids. Similar to what we employed for the rotating sphere, here too we use an indicator function utilizing Eq. (2) with

$$f = 1 - (x^2/a^2 + y^2/b^2 + z^2/c^2), \quad \text{and} \quad \tau = \sqrt[3]{abc}/2,$$

where a, b, c are the semi-axes of the ellipsoid centered at the origin.

With this concentration function we apply SPM to the problem of Stokes flow past a spheroid (aligned with the flow) between two infinite parallel walls. The simulation domain is $[80, 80] \times [-3.3333, 10] \times [-20, 20]$ with $y = -3.3333$ and $y = 10$ to be the walls. Periodicity is imposed in both x- and z-directions. A spheroid with semi-axes of $a = 2, b = 1$ and $c = 1$ is centered at the origin with its long axis aligned with x-axis. A uniform pressure gradient of $\nabla p = (0.06075, 0, 0)$ is applied to drive the flow. Thus, the flow far from the ellipsoid can be modeled as a fully developed Poiseuille flow driven by this pressure gradient, and correspondingly the centerline velocity is $U_{cl} = 1.35$. Based on these values, we obtain from Table 7–5.1 in [48] the "exact" drag force $(F_x)_{ex} = 29.1765$ for a fixed ellipsoid immersed in an infinite Poseuille flow. We use this force value as the "exact" drag force to compare against our SPM results on a finite computational domain.

In the SPM simulation we use 4560 rectilinear elements with polynomial order $P = 3$, with finer resolution around the spheroid (see Fig. 10). We choose the interface thickness to be $\xi = 0.1b$.

For comparison, we have also performed both DNS and FCM simulations. In FCM we employed the same mesh as in the SPM simulation (i.e., 4560 elements with $P = 3$), and we computed the hydrodynamic force and torque using a penalty method as in [41]. DNS simulation employs 7168 hexahedral elements with polynomial order $P = 4$. The meshes, displaying the elements only, are shown in Fig. 10.

We have summarized the results of our simulations in Table 1. It shows that for this value of ξ, the optimum time step size is around 0.005, which, in fact, is the value expected from the balance of the Stokes layer thickness δ and the effective interface thickness l_e, i.e., $\delta = 2.76\sqrt{\nu(\Delta t)_o} = l_e$ (obtained from the Couette problem). For $\xi = 0.1b$, we obtain the effective interface thickness in x, y and z direction as $(l_e)_x = 2 \times 0.1529, (l_e)_y = (l_e)_z = 2 \times 0.07645$. Let us define the total effective

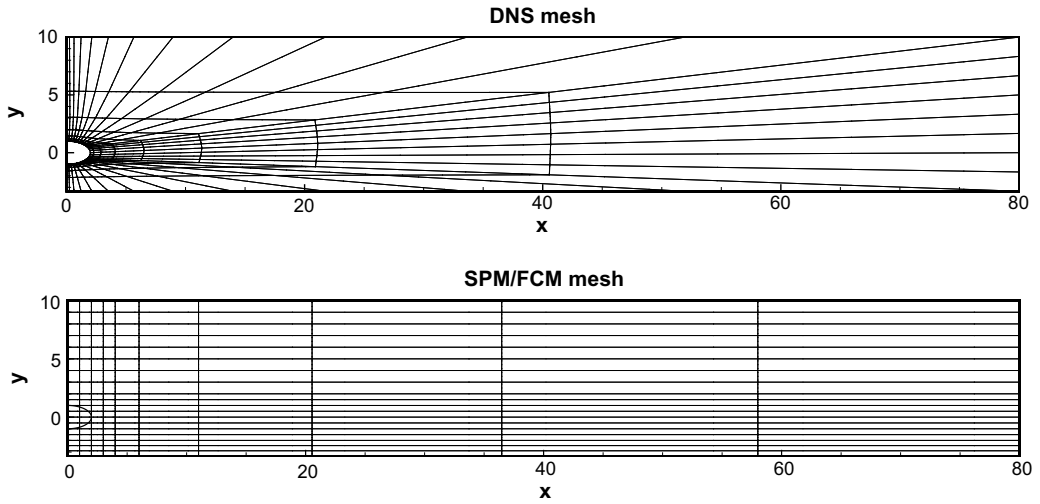


Fig. 10. Flow past an ellipsoid: slices ($z = 0$ plane) of meshes used in the simulations of DNS (top), SPM and FCM (bottom). Due to the symmetry in the streamwise x-direction, only half of the domain is shown.

Table 1
Flow past an ellipsoid: hydrodynamic force in streamwise direction F_x and torque in z-direction T_z. Two different boundary conditions are applied at $x = \pm 80.0$: 'periodic' means periodic at both the inlet $x = -80.0$ and the outlet $x = 80.0$; 'inflow' means inflow type at the inlet and Neumann type at the outlet.

	F_x	Difference from exact (%)	T_z	Difference from DNS (%)
Exact [48]	29.1765			
DNS (periodic)	28.89703	0.958	4.29880	
FCM (periodic)	29.43436	0.884	4.38510	2.01
SPM (periodic, Δt=0.003)	29.90128	2.484	4.61497	7.35
SPM (periodic, Δt=0.005)	28.98194	0.667	4.28673	0.28
SPM (periodic, Δt=0.01)	27.55520	5.557	3.81726	11.20
DNS (inflow)	29.61001	1.486	4.32863	
FCM (inflow)	29.62734	1.545	4.42328	2.187
SPM (inflow, Δt=0.005)	29.170	0.022	4.31411	0.335

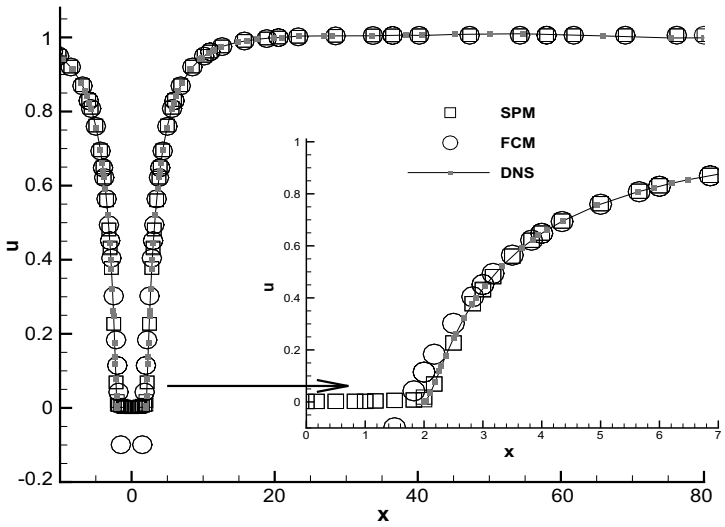


Fig. 11. Flow past an ellipsoid: Comparison between SPM, FCM and DNS of streamwise velocity (u) versus streamwise distance (x) along the long axis of the ellipsoid (x-axis). In the bottom right a close-up around the spheroid surface is shown.

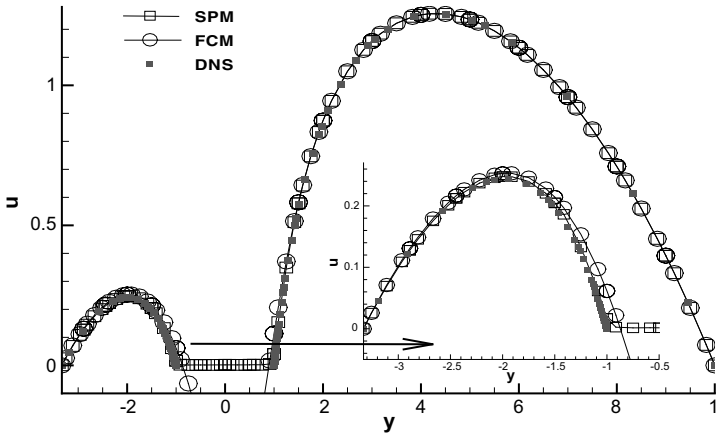


Fig. 12. Flow past an ellipsoid: Comparison between SPM, FCM and DNS of streamwise velocity (u) versus inter-wall distance (y) along the short axis of the ellipsoid (y-axis). In the bottom right a close-up around the spheroid surface is shown.

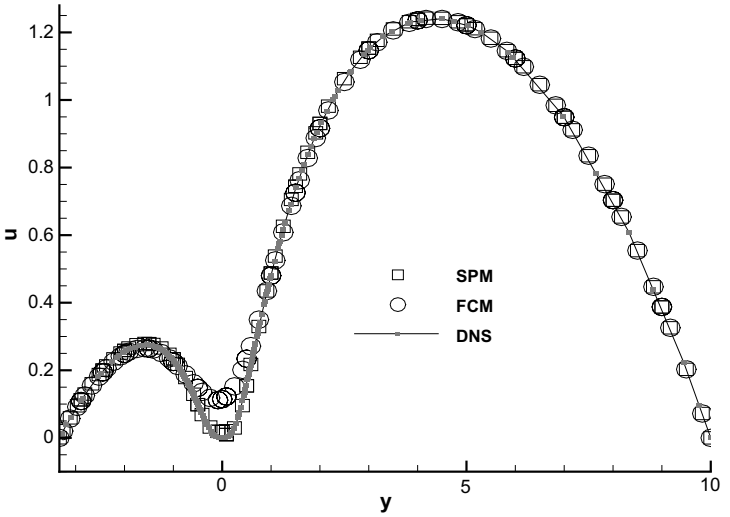


Fig. 13. Flow past an ellipsoid: Comparison between SPM, FCM and DNS of streamwise velocity (u) versus inter-wall distance (y) along the tangential line ($z = 0, x = -2$).

interface thickness as the geometric mean of all three directions to obtain $l_e = \sqrt[3]{(l_e)_x(l_e)_y(l_e)_z} = 2 \times 0.09632$, and hence $(\Delta t)_o = \left(\frac{l_e}{2.76}\right)^2/\nu = 0.00524$, which agrees with the SPM simulation results. Table 1 also shows that the boundary treatment at the inlet and outlet has certain effects in the numerical results.

Next, we compare the SPM results against DNS and FCM results for the same conditions (with periodic boundary treatment) in the pointwise sense, e.g., we compare velocity profiles at different locations, in Figs. 11–13. The results indicate that if we choose the time step wisely and the interface thickness reasonably, SPM resolves the near velocity fields better than FCM while in far-field it is comparable to FCM.

4.2. Flow past a biconcave particle

Biconcave shaped particles resemble red blood cells, so we are interested to assess the accuracy of SPM for flow past a single particle tilted with respect to the flow direction. The concentration function for biconcave is derived from (2) using the surface function f given in [49]:

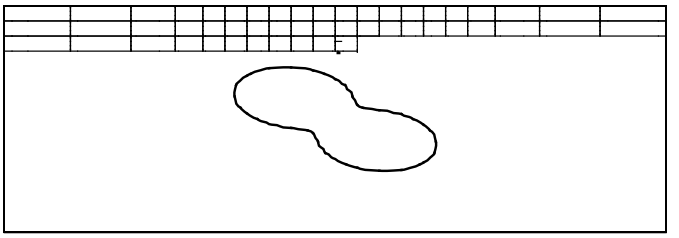
$$f(\mathbf{x}, t) = |\mathbf{x}| - (a - b)\sin^q\theta + b, \quad \text{and} \quad \tau = 1$$

where a, b are the semi-axes of the biconcave, θ is the orientation angle as $\theta = \arccos(x_l/|\mathbf{x}|)$, and x_l is the projection of $\mathbf{x}$ onto the symmetry axis of the biconcave. We simulate the steady Stokes flow past a tilted biconcave particle between two walls.

the z-direction by an angle of $-\pi/4$. Two walls are located at $y = \pm 4$, and periodicity is imposed along the x- and z-directions. A pressure gradient $\nabla p = (0.05, 0, 0)$ is prescribed to drive the flow. As the biconcave is tilted, both drag force F_x and lift force F_y are expected.

The SPM simulation uses 7200 nonuniform hexahedral elements with fourth-order polynomials, while DNS uses 50941 tetrahedral elements and a polynomial order of $P = 8$; the meshes (elements only) are shown in Fig. 14. Compared to the DNS mesh, the rectilinear mesh for SPM is much simpler and more manageable.

Typical results for velocity profiles at different locations are shown in Figs. 15–17. We could not obtain FCM results for this complex shape so we compare against the DNS results with spectral element method. The agreement is very good and comparable to the simple shaped particle simulations we considered in previous sections.



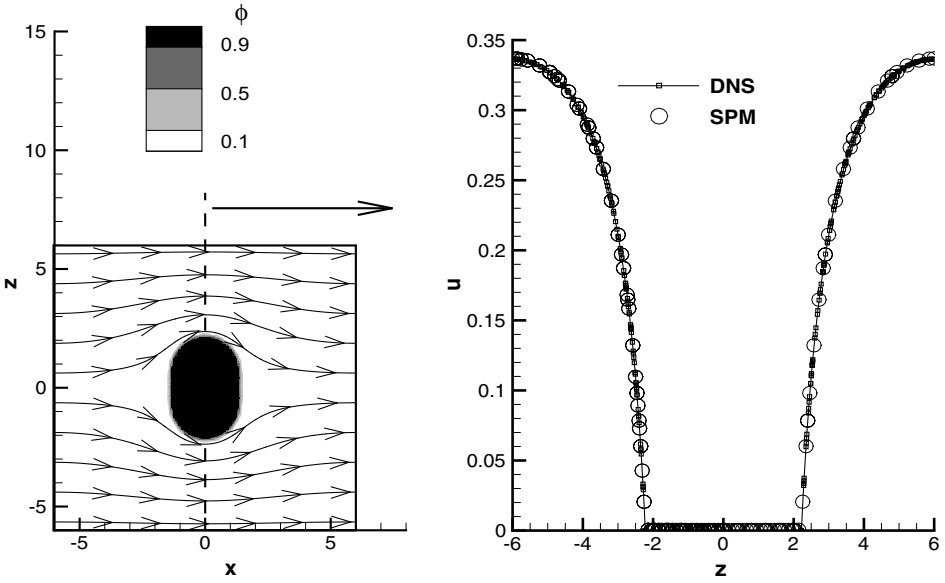


Fig. 16. Flow past a biconcave particle: Left: 2D streamlines and concentration function contours at the $y = 0$ plane by SPM. Right: comparison between SPM and DNS: streamwise velocity (u) versus depth distance (z) along the center line: $y = 0, x = 0$.

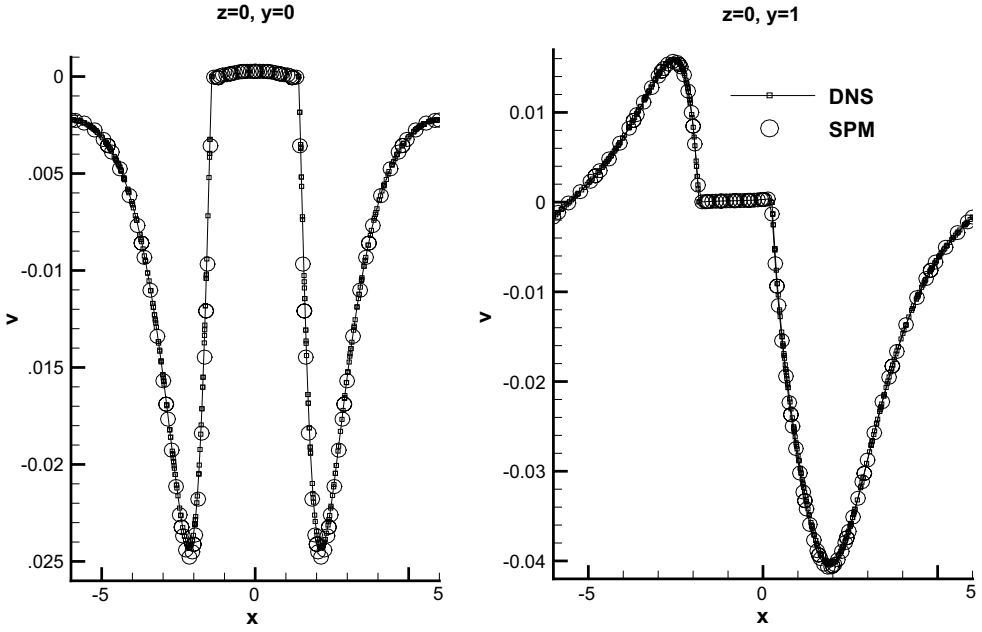


Fig. 17. Flow past a biconcave particle: Inter-wall velocity (v) versus streamwise distance (x). Left: along the center line $z = 0, y = 0$. Right: along the line $z = 0, y = 1$.

5. Two interacting spheres

We employ SPM to study the pairwise interaction between two spheres rising in a vertical channel at low Reynolds number. We check if the SPM simulation can reproduce the drafting–kissing–tumbling (DKT) scenario observed in the experiments [50,51]. The numerical results are compared with both the experimental and FCM results presented in [52].

The experimental setup was a vertical channel of $[0, 150] \times [-5, 5] \times [-50, 50]$ (all in *mm*s). The channel was filled with a mixture of glycerol and water with density $\rho_f = 1.094$ g/cm^3. Two spheres with radius $a = 1$ mm and density

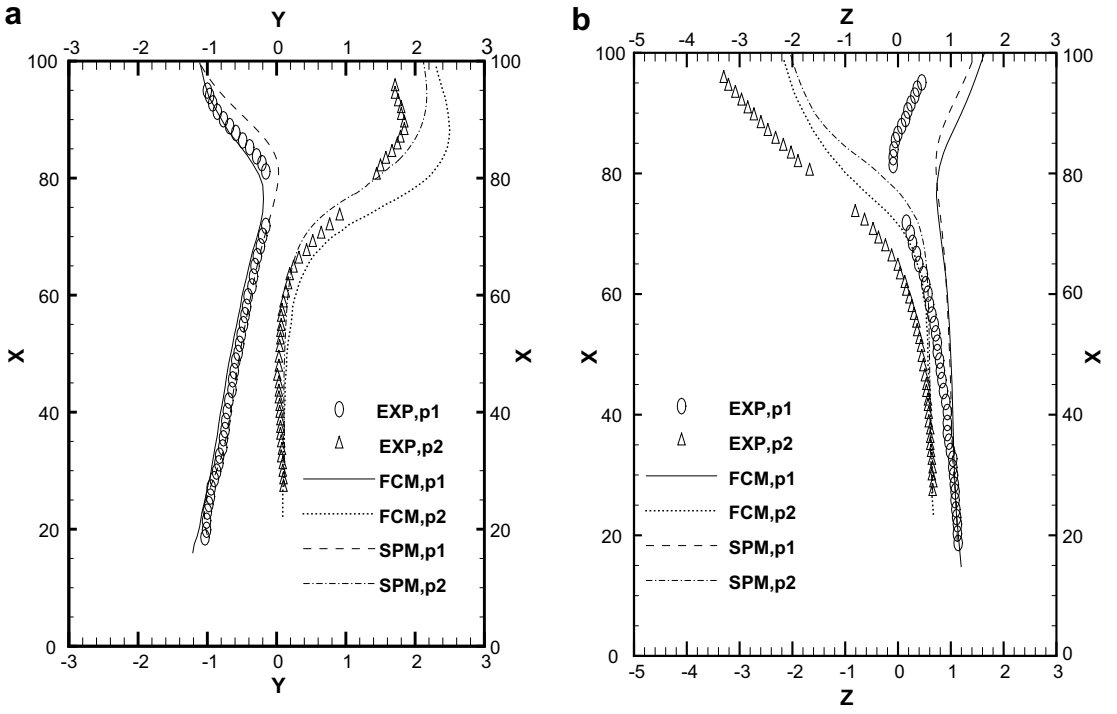


Fig. 18. Two interacting spheres: trajectories of the two particles (center positions) projected into the *Z*-plane (a) and *Y*-plane (b), with 'p1' referring to the first particle and 'p2' to the second.

$\rho_p = 1.081$ g/cm^3 were introduced into the channel through small holes in the bottom, as in the third set of experiment in Table 1 of [52]. Hence, the particles were rising under the composition of the gravity and buoyant force $\mathbf{F} = (\rho_p - \rho_f)(4\pi a^3/3)\mathbf{g}$. Based on the Stokes settling velocity, the Reynolds number was about 13.5.

The simulation domain for SPM is $[0, 40] \times [-5, 5] \times [-10, 10]$ (all in *mms*), with periodic boundaries in *x*- and *z*-direction, and two walls at $y = \pm 5$. The two spheres are initially placed at $(18.65, -1.03, 1.13)$ and $(27.2, 0.1, 0.655)$, and they are driven by the force **F**. To discretize the domain, 6400 hexahedral elements with fifth-order polynomials are used. We choose $\xi = 0.1a$ for the concentration function of Eq. (2), and the optimum time step $(\Delta t)_o$ from $\delta = l_e$ is used in the time integration.

The particle trajectories from the SPM simulation are shown in Fig. 18, with comparison against the experimental and FCM results in [52]. Fig. 18 indicates that SPM reproduces the DKT process and yields satisfactory agreement with the experimental and FCM data, despite the fact that we could not obtain exactly the same initial conditions as in the experiment or FCM simulation and also the simulation domain is somewhat different.

6. Summary and discussion

In this paper, we have reviewed and analyzed a new computational method for particulate flows, the Smoothed Profile method (SPM). We extended the original fully-explicit time-integration scheme to a semi-implicit high-order splitting scheme and implemented it with a spectral/*hp* element discretization in space. We showed that the modeling error of SPM has a *non-monotonic* dependence on time step size, with the optimum time step $(\Delta t)_o$ obtained from a balance between the effective interface thickness and the Stokes layer thickness. We also extended the application of the method to flows past complex-shaped particles. Comparisons with the force coupling method (FCM) and direct numerical simulations (DNS) for several problems show that SPM is resolving accurately the near-field flows as well as the far-field flows.

Besides its accuracy, SPM is also a fast method for modeling particulate flows. Our numerical results show that compared to DNS, SPM is generally much less expensive, since SPM avoids complicated discretizations around the particles, and hence it can reduce computational complexity significantly, especially for complex-shaped particles. Moreover, SPM is advantageous for simulating flows involving moving particles, whereas ALE-based methods require frequent re-meshing. When compared to another fast method FCM, SPM is about 50% more expensive if the same computational meshes are used, as SPM needs to solve the extra equations (Eqs. (21a) and (22)). Also, SPM requires slightly higher spatial resolution than FCM, since it adopts a sharper interface representation (tanh function) instead of the Gaussian envelope used by FCM. Typically, taking the grid spacing Δx around the particle surface to scale as the interface thickness parameter ξ, i.e. $\Delta x \sim \xi$, would resolve the interface sufficiently. This extra work of SPM is a trade-off for better near-field resolution (thus advantageous for dense concentration particulate flow) and also for greater flexibility in modeling complex-shaped particles than FCM.

Although in the previous papers SPM has only been applied to zero and low Reynolds number flow problems ($Re < 20$), extensions to moderate and high Reynolds number flows are feasible. For higher Reynolds number flow, both the Stokes layer thickness ($\delta \sim 2.76\sqrt{\nu\Delta t}$) and the boundary layer thickness ($\delta_{bl} \sim x/\sqrt{Re}$) become thinner. For best modeling accuracy, we need to choose a time step close to the optimum size $(\Delta t)_o$ which balances the effective interface thickness ($l_e = 2.07\xi$) and the Stokes layer thickness (δ), i.e., $l_e = \delta$ as stated in subSection 3.1. In addition, we need to make sure that the smooth interface adequately resolves the boundary layer (i.e., $l_e < \delta_{bl}$), which is *usually* automatically satisfied in our simulations if the optimum time step $(\Delta t)_o$ is used and it is below the CFL limit. Our SPM simulation results for moderate Reynolds number flow around a 2D circular cylinder reveal good agreement with the experimental results of Hammache et al. [53]; e.g., the relative error in the Strouhal number is about 2.2% and 0.9%, respectively, for $Re = 100$ and 150. Also, the SPM results of velocity profiles averaged over one Strouhal period (not shown here for brevity) at different locations are in excellent agreement with the DNS results for $Re = 100 \sim 200$.

SPM can be readily extended to many interacting particles as suggested by our simulations of two particles. However, a proper contact force model is required to prevent the particles from overlapping. Furthermore, SPM is a good candidate for multiscale modeling of colloidal suspensions, and the initial results in [54] based on a similar modeling method are encouraging.

Acknowledgments

This work was supported by National Science Foundation. Special thanks are given to Prof. Don Liu who provided the original FCM-spectral element code, and to Kyongmin Yeo who contributed the FCM results for the rotating sphere section. We would also like to thank Dr. Sune Lomholt for his original data of two interacting particles.

References

[1] Y. Nakayama, R. Yamamoto, Simulation method to resolve hydrodynamic interactions in colloidal dispersions, Phys. Rev. E 71 (2005) 036707.
[2] H.H. Hu, Direct simulation of flows of solid–liquid mixtures, Int. J. Multiphase Flow 22 (2) (1996) 335–352.
[3] H.H. Hu, N.A. Patankar, M.Y. Zhu, Direct numerical simulations of fluid–solid systems using the arbitrary Lagrangian–Eulerian technique, J. Comput. Phys. 169 (2) (2001) 427–462.
[4] R. Glowinski, T.W. Pan, T.I. Hesla, D.D. Joseph, A distributed Lagrange multiplier/fictitious domain method for particulate flow, Int. J. Multiphase Flow 25 (1999) 755–794.
[5] R. Glowinski, T.W. Pan, T.I. Hesla, D.D. Joseph, J. Periaux, A fictitious domain approach to the direct numerical simulation of incompressible viscous flow past moving rigid bodies: application to particulate flow, J. Comput. Phys. 169 (2001) 363–426.
[6] N.A. Patankar, P. Singh, D.D. Joseph, R. Glowinski, T.W. Pan, A new formulation of the distributed Lagrange multiplier/factious domain method for particulate flows, Int. J. Multiphase Flow 26 (2000) 1509–1524.
[7] M.R. Maxey, B.K. Patel, Localized force representations for particles sedimenting in Stokes flow, Int. J. Multiphase Flow 27 (9) (2001) 1603–1626.
[8] S. Lomholt, M.R. Maxey, Force-coupling method for particulate two-phase flow: Stokes flow, J. Comput. Phys. 184 (2003) 381–405.
[9] D. Liu, M.R. Maxey, G.E. Karniadakis, A fast method for particulate microflows, J. Microelectromech. Syst. 11 (2002) 691–702.
[10] D. Liu, M.R. Maxey, G.E. Karniadakis, FCM-spectral element method for simulating colloidal micro-devices, Comput. Fluid Solid Mech. 2003 (2003) 1413–1416.
[11] C.S. Peskin, Flow patterns around heart valves: a numerical method, J. Comput. Phys. 10 (1972) 252.
[12] C.S. Peskin, Numerical analysis of blood flow in the heart, J. Comput. Phys. 25 (1977) 220.
[13] A. Fogelson, C. Peskin, A fast numerical method for solving the three-dimensional stokes equation in the presence of suspended particles, J. Comput. Phys. 79 (1988) 50–69.
[14] R.P. Beyer, R.J. LeVeque, Analysis of a one-dimensional model for the immersed boundary method, SIAM J. Numer. Anal. 29 (2) (1992) 332–364.
[15] M.C. Lai, C.S. Peskin, An immersed boundary method with formal second-order accuracy and reduced numerical viscosity, J. Comput. Phys. 160 (2000) 705–719.
[16] D. Goldstein, R. Handler, L. Sirovich, Modeling a no-slip boundary with an external force field, J. Comput. Phys. 105 (1993) 354–366.
[17] E. Fadlun, R. Verzicco, P. Orlandi, J. Mohd-Yusof, Combined immersed boundary finite-difference methods for three-dimensional complex flow simulations, J. Comput. Phys. 161 (2000) 35–60.
[18] M. Uhlmann, An immersed boundary method with direct forcing for the simulation of particulate flows, J. Comput. Phys. 209 (2) (2005) 448–476.
[19] R.J. LeVeque, Z. Li, The immersed interface method for elliptic equations with discontinuous coefficients and singular sources, SIAM J. Numer. Anal. 31 (1994) 1019–1044.
[20] R.J. LeVeque, Z. Li, Immersed interface methods for Stokes flow with elastic boundaries or surface tension, SIAM J. Sci. Comput. 18 (3) (1997) 709–735.
[21] Z. Li, M.C. Lai, The immersed interface method for the Navier–Stokes equations with singular forces, J. Comput. Phys. 171 (2) (2001) 822–842.
[22] L. Lee, R.J. LeVeque, An immersed interface method for incompressible Navier–Stokes equations, SIAM J. Sci. Comput. 25 (2003) 832–856.
[23] K. Kim, Y. Nakayama, R. Yamamoto, A smoothed profile method for simulating charged colloidal dispersions, Comput. Phys. Commun. 169 (1–3) (2005) 104–106.
[24] R. Yamamoto, Y. Nakayama, K. Kim, A method to resolve hydrodynamic interactions in colloidal dispersions, Comput. Phys. Commun. 169 (1–3) (2005) 301–304.
[25] Y. Nakayama, K. Kim, R. Yamamoto, Hydrodynamic effects in colloidal dispersions studied by a new efficient direct simulation, in: Flow Dynamics, American Institute of Physics Conference Series, vol. 832, 2006, pp. 245–250.
[26] Y. Nakayama, K. Kim, R. Yamamoto, Simulating (electro)hydrodynamic effects in colloidal dispersions: smoothed profile method (cond-mat/0601322).
[27] K. Kim, Y. Nakayama, R. Yamamoto, Direct numerical simulations of electrophoresis of charged colloids (cond-mat/0601534).
[28] R. Yamamoto, K. Kim, Y. Nakayama, Strict simulations of non-equilibrium dynamics of colloids, Colloid Surf. A: Physicochem. Eng. Asp. 311 (1–3) (2007) 42–47.
[29] H.N.O.S. Takagia, Z. Zhangb, A. Prosperetti, PHYSALIS: a new method for particle simulation Part II: Two-dimensional Navier–Stokes flow around cylinders, J. Comput. Phys. 189 (2) (2003) 493–511.
[30] H. Huang, S. Takagi, PHYSALIS: a new method for particle flow simulation, Part III: Convergence analysis of two-dimensional flows, J. Comput. Phys. 189 (2) (2003) 493–511.
[31] Y. Nakayama, K. Kim, R. Yamamoto, Simulating (electro)hydrodynamic effects in colloidal dispersions: smoothed profile method, Eur. Phys. J. E 26 (2008) 361–368.
[32] K. Kim, Y. Nakayama, R. Yamamoto, Direct numerical simulations of electrophoresis of charged colloids, Phys. Rev. Lett. 96 (2006) 208302.

[33] D. Funaro, D. Gottlieb, A new method of imposing boundary conditions in pseudospectral approximations of hyperbolic equations, Math. Comput. 51 (184) (1988) 599–613.
[34] J.S. Hesthaven, D. Gottlieb, A stable penalty method for the compressible Navier–Stokes equations: I. Open boundary conditions, SIAM J. Sci. Comput. 17 (3) (1996) 579.
[35] G.E. Karniadakis, M. Israeli, S.A. Orszag, High-order splitting methods for the incompressible Navier–Stokes equations, J. Comput. Phys. 97 (1991) 414.
[36] G.E. Karniadakis, S.J. Sherwin, Spectral/hp Element Methods for CFD, Oxford University Press, New York, 1999.
[37] D. Xiu, G.E. Karniadakis, A semi-Lagrangian high-order method for Navier–Stokes equations, J. Comput. Phys. 172 (2001) 658–684.
[38] P.K. Kundu, I.M. Cohen, Fluid Mechanics, second ed., Academic Press, 2002 (Chapter 9, pp. 282–288).
[39] H. Hasimoto, On the periodic fundamental solutions of the Stokes equation and their application to viscous flow past a cubic array of spheres, J. Fluid Mech. 5 (1959) 317–328.
[40] R.M. Kirby, T.C. Warburton, S.J. Sherwin, A. Beskok, G.E. Karniadakis, The Nektar code: dynamic simulations without re-meshing, in: Proceedings of the 2nd International Conference on Computational Technologies for Fluid/Thermal/Chemical Systems with Industrial Applications, August 1–5, 1999.
[41] D. Liu, M.R. Maxey, G.E. Karniadakis, Modeling and optimization of colloidal micro-pumps, J. Micromech. Microeng. 14 (9) (2004) 567–575.
[42] W.G. Bickley, On the secondary flow due to a sphere rotating in a viscous fluid, Talbot Bull. Lond. Math. Soc. 5 (1973) 349–354.
[43] H. Takagi, Viscous flow induced by slow rotation of a sphere, J. Phys. Soc. Jpn. 42 (1) (1977) 319–325.
[44] H. Lamb, Hydrodynamics, Cambridge University Press, 1932. pp. 558–559.
[45] S.C.R. Dennis, S.N. Singh, D.B. Ingham, The steady flow due to a rotating sphere at low and moderate Reynolds number, J. Fluid Mech. 101 (1980) 257–279.
[46] E. Climent, K. Yeo, M.R. Maxey, G.E. Karniadakis, Dynamic self-assembly of spinning particles, J. Fluid Eng. 129 (2007) 379–387.
[47] R.L. Panton, Incompressible Fluid Flow, Wiley, New York, 1984.
[48] J. Happel, H. Brenner, Low Reynolds Number Hydrodynamics, Prentice-Hall, 1965 (Chapter 7, pp. 331–337).
[49] J.Q. Lu, P. Yang, X.H. Hu, Simulations of light scattering from a biconcave red blood cell using the finite-difference time-domain method, J. Biomed. Opt. 10 (2) (2005) 024022.
[50] K.O.L.F. Jayaweera, B.J. Mason, G.W. Slack, The behaviour of clusters of spheres falling in a viscous fluid Part 1. Experiment, J. Fluid Mech 20 (1964) 121–128.
[51] A. Fortes, D. Joseph, T. Lundgren, Nonlinear mechanics of fluidization of beds of spherical particles, J. Fluid Mech. 177 (1987) 467–483.
[52] S. Lomholt, B. Stenum, M. Maxey, Experimental verification of the force-coupling method for particulate flows, Int. J. Multiphase Flow 28 (2002) 225–246.
[53] M. Hammache, M. Gharib, An experimental study of the parallel and oblique vortex shedding from circular cylinders, J. Fluid Mech. Dig. Arch. 232 (1991) 567–590.
[54] M. Fujita, Y. Yamaguchi, Multiscale simulation method for self-organization of nanoparticles in dense suspension, J. Comput. Phys. 223 (1) (2007) 108–120.