PERGAMON

International Journal of Multiphase Flow 25 (1999) 755–794

International Journal of Multiphase Flow

www.elsevier.com/locate/ijmulflow

A distributed Lagrange multiplier/fictitious domain method for particulate flows

R. Glowinski[a,]*, T.-W. Pan[a], T.I. Hesla[b], D.D. Joseph[b]

[a]*Department of Mathematics, University of Houston, Houston, TX 77204, USA*
[b]*Department of Aerospace Engineering and Mechanics, University of Minnesota, Minneapolis, MN 55455, USA*

Received 5 November 1997; received in revised form 19 July 1998

Abstract

A new Lagrange-multiplier based fictitious-domain method is presented for the direct numerical simulation of viscous incompressible flow with suspended solid particles. The method uses a finite-element discretization in space and an operator-splitting technique for discretization in time. The linearly constrained quadratic minimization problems which arise from this splitting are solved using conjugate-gradient algorithms.

A key feature of the method is that the fluid–particle motion is treated implicitly via a combined weak formulation in which the mutual forces cancel—explicit calculation of the hydrodynamic forces and torques on the particles is not required. The fluid flow equations are enforced inside, as well as outside, the particle boundaries. The flow inside, and on, each particle boundary is constrained to be a rigid-body motion using a distributed Lagrange multiplier. This multiplier represents the additional body force per unit volume needed to maintain the rigid-body motion inside the particle boundary, and is analogous to the pressure in incompressible fluid flow, whose gradient is the force required to maintain the constraint of incompressibility.

The method is validated using the sedimentation of two circular particles in a two-dimensional channel as the test problem, and is then applied to the sedimentation of 504 circular particles in a closed two-dimensional box. The resulting suspension is fairly dense, and the computation could not be carried out without an effective strategy for preventing particles from penetrating each other or the solid outer walls; in the method described herein, this is achieved by activating a repelling force on close approach, such as might occur as a consequence of roughness elements on the particle. The development of physically based mathematical methods for avoiding particle–particle and particle–wall penetration is a new problem posed by the direct simulation of fluidized suspensions.

The simulation starts with the particles packed densely at the top of the sedimentation column. In the course of their fall to the bottom of the box, a fingering motion of the particles, which are heavier than

* Corresponding author. Tel.: 713 743 3473; Fax: 713 743 3505.

PII: S0301-9322(98)00048-2

the surrounding fluid, develops in a way reminiscent of the familiar dynamics associated with the Rayleigh–Taylor instability of heavy fluid above light.

We also present here the results of a three-dimensional simulation of the sedimentation of two spherical particles. The simulation reproduces the familiar dynamics of *drafting*, *kissing* and *tumbling* to side-by-side motion with the line between centers across the flow at Reynolds numbers in the hundreds.

Keywords: Particulate flow; Solid–liquid flow; Fictitious-domain method; Distributed Lagrange multiplier; Combined equation of motion; Operator splitting; Finite element

1. Preliminaries

1.1. Introduction

The current popularity of computational fluid dynamics is rooted in the perception that information implicit in the equations of fluid motion can be extracted without approximation using direct numerical simulation. The corresponding potential for solid–liquid flows—and multiphase flows generally—has yet to be fully realized. In this article, we describe a promising new method—a *fictitious-domain method*—for the direct numerical simulation of the flow of fluid-particle mixtures.

By direct numerical simulation, we understand the numerical solution of the exact initial value problem for a fluid–particle mixture. In the method described in this article, the fluid is taken to be Newtonian and the particles to be rigid bodies. To perform a direct simulation in the above sense, therefore, one must simultaneously integrate the Navier–Stokes equations (governing the motion of the fluid) and the equations of rigid-body motion (governing the motion of the particles). These equations are coupled through the no-slip condition on the particle boundaries, and through the hydrodynamic forces and torques which appear in the equations of rigid-body motion.

These hydrodynamic forces and torques must of course be those arising from the computed motion of the fluid, and so are not known in advance, but only as the integration proceeds. It is crucial that no approximation of these forces and torques be made—other than that due to the numerical discretization itself—so that the overall simulation will yield a solution of the exact coupled initial value problem—up to the numerical truncation error.

Our goal is to do direct numerical solutions with many thousands of particles in three dimensions, with large volume fractions, for the various kinds of suspensions and slurries that model the practical particulate flows arising in applications like fluidized beds, slurry transport, transport of drill cuttings for oil production, and proppant sands in reservoir stimulations. In this article, we present the results of a simulation with 504 particles in two dimensions; simulations with 10,000 particles in two dimensions will certainly appear in the near future. The direct simulation of fluidized suspensions of thousands of particles in three dimensions is a computationally intensive challenge, not yet achieved, but which will almost certainly be met in the next years. See our Web site

http : //www.aem.umn.edu/Solid-Liquid_Flows

for recent progress in this direction.

The hope is that the direct simulation of the motions of thousands of particles will, in many cases, allow the large numbers of experiments used in deriving engineering correlations to be replaced by cheaper numerical experiments in which flow, material, and process-control parameters can be altered with a computer command. There are also opportunities for the application of direct simulation to the diagnosis of industrial problems involving flowing particulates, to the establishment of benchmark standards for two-phase flow models, to lattice-Boltzmann models, and to point-particle approximations.

1.2. Approximate methods

Many excellent numerical studies of particulate flows of many particles which are not direct simulations in the above sense have appeared in recent years. These approximate methods include simulations based on potential flow, Stokes flow, and point-particle approximations; they all simplify the computation by ignoring some possibly important effects like viscosity and wakes in the case of potential flow, inertial forces which produce lateral migration and across-the-stream orientations in the case of Stokes flow, and the effects of stagnation and separation points in the case of point-particle approximations.

Without giving here a long list of such approximations, we simply point the reader to the excellent discussions of these matters, and the rather complete set of references, in the papers of Hu (1996) and Esmaeeli and Tryggvason (1998). The latter authors note that the point-particle approximations which have been used for dilute solutions, especially for turbulent flow, do advect particles by Newton's law:

> "In some cases the influence of the particles on the flow are [sic] neglected, in other cases the force on the fluid from the particles is added to the Navier–Stokes equations. Although this approach is often referred to as "direct simulations" by its practitioners, the forces on each particle are related to its motion and the fluid velocity by semi-empirical relations and this method is only applicable to dilute flows where there are no direct bubble–bubble interactions".

1.3. Direct simulations

Direct simulation of the motion of solid particles in fluids can be said to have started with the paper of Hu et al. (1992), and direct simulations of the interaction of bubbles were first carried out in two and three dimensions by Unverdi and Tryggvason (1992a, 1992b). These methods work well even at Reynolds numbers in the hundreds.

The method of Unverdi and Tryggvason is a front-tracking method in which the bubbles communicate with each other by finite differences on a structured grid, and with their immediate environment through well known interface conditions, represented on a moving mesh. The discretized interface conditions can be thought of as smoothed-out versions of the delta functions used to express the exact interface conditions. The works of the group of Tryggvason are state-of-the-art examples of the direct simulation of the motion of large numbers of bubbles in various kinds of problems, but their method has yet to be applied to solid–liquid flows.

The method developed by Hu et al. (1992) uses an implicit update of the particle translational and angular velocities, to prevent numerical instability. This is achieved by alternately computing the hydrodynamic force and torque, then updating the particle translational and angular velocities using the equations of rigid-body motion, and iterating until the translational and angular velocities converge.

Hesla (1991) derived a combined weak formulation which obviates this iterative procedure. This combined formulation is used in the improved version of Hu's scheme (1996), and—suitably generalized—in the fictitious-domain scheme developed in the present article. In Hesla's combined formulation, the fluid and particle equations of motion are combined into a single weak equation of motion, called the *combined equation of motion*, or *total momentum equation*, in which the hydrodynamic forces and torques are completely eliminated. The combined weak formulation will be derived below. It is a good starting point for constructing finite-element schemes for the direct simulation of fluid–particle flows.

1.3.1. Methods based on moving unstructured grids

The schemes described in Hu et al. (1992), Hesla (1991), and Hu (1996), all use unstructured grids. Hu's improved method (1996) uses an arbitrary Lagrangian–Eulerian (ALE) moving mesh technique—with remeshing and projection as needed—to deal with the time-dependent domain determined by the moving particle boundaries. The particle positions and angular orientations, as well as the mesh point positions, are updated explicitly. The fluid velocity, and the particle translational and angular velocities, are updated implicitly.

Using his improved method, Hu simulated the sedimentation of 400 heavy particles in an upward, two-dimensional Poiseuille flow. The same method was used by Patankar (1997) in his Ph.D. thesis. This method is now being applied to simulations of 1000 particles in two dimensions and 50 spherical particles in three dimensions. It will also be applied to the study of fluidized beds.

A different numerical method, also using unstructured grids, was developed by Johnson and Tezduyar (1996, 1997), and has been applied to cases involving multiple spheres falling in a liquid-filled tube in three dimensions with the number of particles reaching 100. Their method uses a stabilized space-time formulation to deal with the time-dependent domain, and an automatic mesh generation method which produces structured layers of elements around the particles, and unstructured elements elsewhere in the domain. The mesh is updated using an automatic mesh moving method (with the structured layers of elements "glued" to the particles), combined with remeshing and projection as needed. The method has been implemented in parallel.

The method of Johnson and Tezduyar is explicit: as in the earlier work of Hu et al. (1992), the particles are moved by the hydrodynamic forces and torques (in conjunction with gravity, of course), which must be computed. This can be compared with the implicit methods used by Hesla (1991) and Hu (1996), which are based on the combined weak formulation of Hesla in which the hydrodynamic forces and torques are completely eliminated.

Johnson and Tezduyar apply their method to the sedimentation of polydisperse spheres; their simulations reproduce the microstructural across-the-stream structures associated with *drafting*, *kissing*, and *tumbling* (Fortes et al., 1987; Joseph et al.; 1987, and Joseph, 1996). For 100 spheres, however, the number of mesh points required is extremely large, and remeshing/projection is required quite often, due to the high frequency of "near-collisions" between spheres. [1] To alleviate this problem—that is, to prevent particles from approaching each other too closely—Johnson and Tezduyar have implemented a "collision strategy," described later.

1.3.2. Fictitious domain methods

Fictitious-domain methods, also called *domain-embedding methods*, comprise a large class of solution methods for partial differential equations. The basic idea is to extend a problem on a geometrically complex (possibly time-dependent) domain to a larger, simpler domain (the "fictitious domain"). This conceptual framework provides two key advantages in constructing computational schemes:

- The extended domain is geometrically simpler, so it admits simpler, more regular meshes. This may allow specialized, fast solution methods, such as fast direct solvers for elliptic problems on rectangular domains.
- The extended domain may be time-independent—even if the original domain is time-dependent. Thus, the same, *fixed* mesh can be used for the entire computation, eliminating the need for repeated remeshing and projection. This is in sharp contrast to the situation for the unstructured-grid methods described above.

Of course, the boundary conditions on the original boundary must still be enforced, in order for the solution of the extended problem to match the solution of the original problem on the original domain.

Fictitious-domain methods were, to the best of our knowledge, introduced by Hyman (1952). They were also discussed by Saul'ev (1963) (who coined the term "fictitious domain") and by Buzbee et al. (1971). Glowinski et al. (1994a, 1994b, 1995), described fictitious-domain methods for the Dirichlet problem in which the boundary condition (on the boundary of the original domain) is enforced as a side constraint, using a (boundary-supported) Lagrange multiplier. These methods use structured, regular meshes (which are not boundary-fitted) over the extended domain. They also applied the methods to the solution of nonlinear time-dependent problems, such as the variational inequalities modeling the flow of a viscous-plastic medium in a pipe, the Ginzburg–Landau equations, and the Navier–Stokes equations.

[1] In principle, the lubrication forces implicit in the equations of fluid motion will not allow two particles to touch without film rupture, which is ignored in the equations of motion. In experiments, actual touching contacts are observed in gas-fluidized beds of heavy particles. In liquid-fluidized beds, the lubrication forces are higher, so the particles do not actually touch, in general, although they can approach *arbitrarily* closely if their relative velocity is great enough.

Examples of non Lagrange multiplier based fictitious-domain methods can be found in the *immersed boundary method* of C. Peskin and his collaborators (Peskin, 1977; Peskin and McQueen, 1980; Peskin, 1981) for the simulation of incompressible viscous flow in regions with elastic moving boundaries, and in the work of LeVeque and Li (1994, 1997).

Glowinski et al. (1997b, 1997c) describe fictitious-domain methods for incompressible viscous unsteady flow around rigid particles which have *prescribed* motions. For this problem, the original ("complex") domain is the region occupied by the fluid, and the extended domain is the region occupied by the fluid *together with* the interiors of the particles. The no-slip condition on the particle boundaries is enforced as a side constraint, using a Lagrange multiplier. A time discretization by operator splitting à la Marchuk–Yanenko is coupled with an L^2-projection technique to enforce the incompressibility condition to produce a method that is robust, stable, and easy to implement.

If the motions of the particles are caused by the hydrodynamic forces and torques (in addition to gravity) rather than being prescribed in advance, the equations of fluid motion are coupled with the equations of rigid-body motion. As for Hu's improved method (1996) referred to above, it is advantageous to derive a combined weak formulation of the problem in which the hydrodynamic forces and torques are eliminated—a generalization of Hesla's (1991) combined weak formulation to the extended domain. To make this possible, the constraint of rigid-body motion on the *boundaries* of the particles (that is, the no-slip condition) must be extended to their *interiors* as well. This extended constraint can be enforced using a *distributed* Lagrange multiplier.

A preliminary version of a finite-element scheme based on these ideas was introduced by Glowinski et al. (1997a). [2] A more complete version of this scheme—which includes a short-range repulsive force to prevent particle–particle and particle-wall interpenetration—is described in the present article. The method is quite different from that described in Glowinski et al. (1997b, 1997c); it has been applied to sedimenting particles in a channel, and accurately reproduces the hydrodynamical interactions among the particles.

Assuming that the outer boundary of the fluid–particle domain is independent of time, the extended domain is also time-independent, obviating the need for repeated remeshing/projection, as pointed out above. [3] This is a great advantage, since, for three-dimensional particulate flow, the automatic generation of unstructured, body-fitted meshes in the region outside a large number of closely spaced particles is a difficult problem. And if the outer boundary (in two dimensions) is rectangular, a structured mesh can be used, which allows fast elliptic solvers to be used.

1.4. Collision strategies

As alluded to above, when simulating dense suspensions, near-collisions occur with high frequency. In unstructured-grid methods, this necessitates an inordinately large number of

[2] The notation used in Glowinski et al. (1997a) differs from that of the present article.

[3] In order to stably compute the distributed Lagrange multiplier, a separate, coarser mesh must be generated inside each particle at each time step. However, this mesh is very simple, and can be generated very quickly.

mesh points in the narrow gap between the close particles, and causes the mesh to become distorted very quickly, requiring a high frequency of remeshing and projection. What is needed is a "collision strategy" for preventing such near-collisions, while still conserving total mass and momentum (within the numerical truncation error).

Johnson and Tezduyar (1996, 1997) implement a collision strategy based on the physics of inelastic collisions, which requires the selection of a "coefficient of restitution." This strategy is appropriate for handling the actual touching contacts which occur in gas-fluidized beds of heavy particles.

However, in liquid-fluidized beds where the particles do not actually collide, a simpler strategy may be better. The strategy adopted for the fictitious-domain method described below is to activate an artificial repelling force at close range. This circumvents the need for moving locally structured grids, and selecting a restitution coefficients, and may be a more appropriate and computationally efficient way to handle the near-collisions of particles in liquid-fluidized suspensions. Some excellent approaches for treating collisions between particles of arbitrary shape have recently been proposed by Maury (1997).

1.5. Viscoelastic fluids

The first paper on the direct simulations of the motion of particles in viscoelastic fluids was the study of the two-dimensional sedimentation of circular particles in an Oldroyd B fluid by Feng et al. (1996). The method used is an extension of the unstructured-grid method of Hu et al. (1992) to Oldroyd B fluids. The results show chains of particles aligned along, rather than across the stream—precisely the microstructures observed in actual experiments.

More recent applications of this method can be found in the paper on the direct simulation of the motion of particles in Couette and Poisseuille flows of viscoelastic and shear-thinning fluids by Huang et al. (1997), and in the paper of Huang et al. (1998) on the effects of viscoelasticity and shear thinning on the stable orientation of ellipses falling in a viscoelastic fluid. Particulate flows of as many as 100 particles in an Oldroyd B fluid were reported in the Ph.D. thesis of Patankar (1997), and video animation of some of these can be seen on our Web site

http : //www.aem.umn.edu/Solid-Liquid_Flows

These are the only direct simulations of initial value problems for flows of particles in viscoelastic fluids which have been published to date; we believe that these represent the only numerical packages available which can move solid particles in a viscoelastic fluid.

2. Mathematical formulation

2.1. The governing equations

As stated in Section 1.1, the fluid is assumed to be Newtonian. The method will eventually be extended to the viscoelastic case. It is assumed that the lubrication forces are large enough to prevent particles from touching each other, or the walls. To make the equations associated with the angular motion of the particles simpler, we focus on the two-dimensional case. The extension to three dimensions is straightforward.

Let Ω be the entire computational domain, *including* the interior of the particles. For simplicity, Ω will be taken to be a rectangle, with sides Γ_1, Γ_2, Γ_3, and Γ_4. Let $P_i(t)$ be the interior of the ith particle, $i = 1, \ldots, N$, where N is the number of particles (see Fig. 1). We assume, for simplicity, that the fluid velocity satisfies a Dirichlet boundary condition on the outer boundary $\Gamma = \cup_{i=1}^{4}\Gamma_i$, and that ρ_d (the particle density) and $\mathbf{g}$ (the body force per unit mass) are constant, so that the body forces exert no net torque on the particles.

2.1.1. Strong form

Under the above assumptions, the motion of a fluid–particle mixture is governed by the following equations.

Fluid motion

$$\rho_\mathrm{L}\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} = \rho_\mathrm{L}\mathbf{g} + \nabla\cdot\boldsymbol{\sigma} \quad \text{in } \Omega\backslash\overline{P(t)}, \tag{1}$$

$$\nabla\cdot\mathbf{u} = 0 \quad \text{in } \Omega\backslash\overline{P(t)}, \tag{2}$$

$$\mathbf{u} = \mathbf{u}_\Gamma(t) \quad \text{on } \Gamma, \tag{3}$$

$$\mathbf{u} = \mathbf{U}_i + \omega_i\times\mathbf{r}_i \quad \text{on } \partial P_i(t), \quad i = 1,\ldots,N, \tag{4}$$

$$\mathbf{u}|_{t=0} = \mathbf{u}_0 \quad \text{in } \Omega\backslash\overline{P(0)}. \tag{5}$$

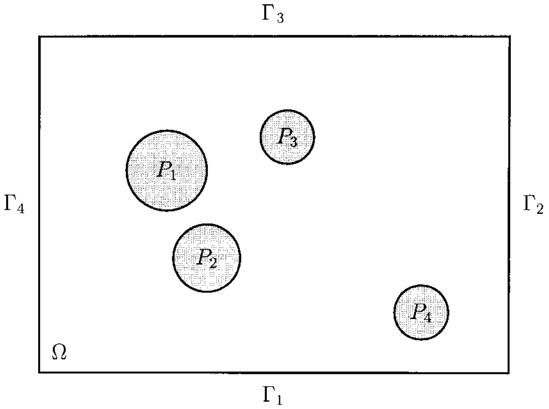

Fig. 1. Fluid–particle system.

Particle motion

$$M_i \frac{\mathrm{d}\mathbf{U}_i}{\mathrm{d}t} = M_i\mathbf{g} + \mathbf{F}_i, \tag{6}$$

$$I_i \frac{\mathrm{d}\omega_i}{\mathrm{d}t} = T_i, \tag{7}$$

$$\mathbf{U}_i|_{t=0} = \mathbf{U}_{i,0}, \tag{8}$$

$$\omega_i|_{t=0} = \omega_{i,0}. \tag{9}$$

Kinematic equations

$$\frac{\mathrm{d}\mathbf{X}_i}{\mathrm{d}t} = \mathbf{U}_i, \tag{10}$$

$$\frac{\mathrm{d}\Theta_i}{\mathrm{d}t} = \omega_i, \tag{11}$$

$$\mathbf{X}_i|_{t=0} = \mathbf{X}_{i,0}, \tag{12}$$

$$\Theta_i|_{t=0} = \Theta_{i,0}. \tag{13}$$

Here

$$\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} = \frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} \tag{14}$$

is the material derivative;

$$P(t) = \bigcup_{i=1}^{N} P_i(t)$$

is the region occupied by the particles; $\mathbf{r}_i = \mathbf{x} - \mathbf{X}_i$; ρ_L, $\mathbf{u}$, and $\boldsymbol{\sigma}$ are the fluid density (assumed constant), velocity, and stress; M_i, I_i, $\mathbf{U}_i$, ω_i, $\mathbf{X}_i$, and Θ_i, are the mass, moment of inertia, translational velocity, angular velocity, center of mass, and angular orientation of the ith particle, respectively; and

$$\mathbf{F}_i = \int_{\partial P_i(t)} \boldsymbol{\sigma}\hat{\mathbf{n}} \, \mathrm{d}s, \;\; T_i = \int_{\partial P_i(t)} \mathbf{r}_i \times \boldsymbol{\sigma}\hat{\mathbf{n}} \, \mathrm{d}s \tag{15}$$

are the hydrodynamic force and torque (about the center of mass) on the ith particle, and $\hat{\mathbf{n}}$ is the unit normal vector [to $\partial P_i(t)$] pointing *out* of the particle. Note that the initial velocity $\mathbf{u}_0$ and boundary data $\mathbf{u}_\Gamma(t)$ must satisfy the compatibility conditions [4]

$$\nabla \cdot \mathbf{u}_0 = 0 \quad \text{in } \Omega \backslash \overline{P(0)}, \quad \int_\Gamma \mathbf{u}_\Gamma(t) \cdot \hat{\mathbf{n}}\, \mathrm{d}s = 0. \tag{16}$$

Since the fluid is Newtonian, the stress tensor $\boldsymbol{\sigma}$ takes the form:

$$\boldsymbol{\sigma} = -p\mathbf{I} + 2\eta \mathbf{D}[\mathbf{u}], \tag{17}$$

where p is the pressure, η is the viscosity (assumed constant), and $\mathbf{D}[\mathbf{u}]$ is the rate-of-strain tensor.

2.1.1.1. Remarks. For two-dimensional motion, the three-dimensional vector angular velocity is

$$\boldsymbol{\omega}_i = (0,\ 0,\ \omega_i).$$

The term $\omega_i \times \mathbf{r}_i$ in (4) should thus more properly have been written

$$\boldsymbol{\omega}_i \times \mathbf{r}_i = (-\omega_i \mathbf{r}_{i,y}, \omega_i \mathbf{r}_{i,x},\ 0).$$

For simplicity, however, we shall continue to use the notation $\omega_i \times \mathbf{r}_i$. In a similar vein, the integrand $\mathbf{r}_i \times \boldsymbol{\sigma}\hat{\mathbf{n}}$ in (15) is actually the three-dimensional vector

$$\mathbf{r}_i \times \boldsymbol{\sigma}\hat{\mathbf{n}} = [0,\ 0,\ -\mathbf{r}_{i,y}(\boldsymbol{\sigma}\hat{\mathbf{n}})_x + \mathbf{r}_{i,x}(\boldsymbol{\sigma}\hat{\mathbf{n}})_y].$$

The right-hand side of (15) should be interpreted as the z component of this vector, integrated over $\partial P_i(t)$.

Eq. (7) is valid in two dimensions; it is also valid in three dimensions when the particles are spherically symmetric if we replace the scalar ω_i by $\boldsymbol{\omega}_i$, since the inertia tensor $\mathbf{I}_i$ is isotropic when the ith particle is spherically symmetric. In the general three-dimensional case, (7) must be replaced by

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{I}_i\boldsymbol{\omega}_i = \mathbf{I}_i\frac{\mathrm{d}\boldsymbol{\omega}_i}{\mathrm{d}t} + \boldsymbol{\omega}_i \times \mathbf{I}_i\boldsymbol{\omega}_i = \mathbf{T}_i,$$

where $\mathbf{T}_i$ is the (vector) torque (about the center of mass) on the ith particle.

In three dimensions, the orientation of the ith particle cannot be described by a single angle; three angles are required—for example, the Euler angles. Thus, (11) and (13) would need to be replaced by the corresponding equations for the three Euler angles.

For circular particles in two dimensions and spherical particles in three dimensions, (11) and (13) decouple from the remaining equations and may be ignored.

2.1.1.2. Collision strategy. To prevent particles from penetrating each other or the four walls Γ_1, Γ_2, Γ_3, and Γ_4, we adopt the following *collision strategy*. For simplicity, it is assumed that the particles are circular; the technique can easily be generalized to handle arbitrary shaped

[4] Eq. (16) follows from (2) by integration, making use of (4).

particles. The strategy is to replace (6) by

$$M_i \frac{\mathrm{d}\mathbf{U}_i}{\mathrm{d}t} = M_i \mathbf{g} + \mathbf{F}_i + \mathbf{F}'_{\mathbf{i}}, \tag{18}$$

where

$$\mathbf{F}'_i = \sum_{\substack{j=1 \\ j \neq i}}^{N} \mathbf{F}^{\mathrm{p}}_{i,j} + \sum_{j=1}^{4} \mathbf{F}^{\mathrm{w}}_{i,j}$$

is a short-range repulsive force exerted on the ith particle by the other particles and by the walls. For the particle–particle repulsive force, we take

$$\mathbf{F}^{\mathrm{p}}_{i,j} = \begin{cases} \mathbf{0}, & d_{i,j} > R_i + R_j + \rho, \\ \dfrac{1}{\epsilon_{\mathrm{p}}}(\mathbf{X}_i - \mathbf{X}_j)(R_i + R_j + \rho - d_{i,j})^2, & d_{i,j} \leq R_i + R_j + \rho, \end{cases} \tag{19}$$

where $d_{i,j} = |\mathbf{X}_i - \mathbf{X}_j|$ is the distance between the centers of the ith and jth particles, R_i is the radius of the ith particle, ρ is the force range, and ϵ_{p} is a small positive "stiffness" parameter. For the particle-wall repulsive force, we take

$$\mathbf{F}^{\mathrm{w}}_{i,j} = \begin{cases} \mathbf{0}, & d'_{i,j} > 2R_i + \rho, \\ \dfrac{1}{\epsilon_{\mathrm{w}}}(\mathbf{X}_i - \mathbf{X}'_{i,j})(2R_i + \rho - d'_{i,j})^2, & d'_{i,j} \leq 2R_i + \rho, \end{cases} \tag{20}$$

where $d'_{i,j} = |\mathbf{X}_i - \mathbf{X}'_{i,j}|$ is the distance between the centers of the ith particle and the imaginary particle $P'_{i,j}$ located on the other side of Γ_j (see Fig. 2), and ϵ_{w} is another (small positive) "stiffness" parameter. In the future, following Maury (1997), we will implement more sophisticated collision strategies, in order to allow particles of arbitrary shape.

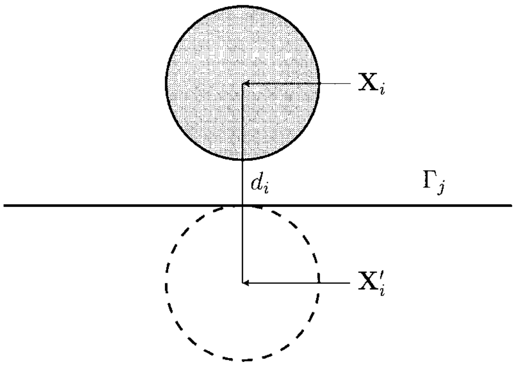

Fig. 2. Imaginary particle.

2.1.2. Weak form

In deriving the weak form of the governing equations, the hydrodynamic forces and torques on the particles can be *completely eliminated* by combining the fluid and particle equations of motion ((1), (18), and (7)) into a single weak equation of motion for the combined fluid–particle system. This equation—called the *combined equation of motion*, or *total momentum equation*—was first introduced by Hesla (1991), who used it as the basis of a finite-element scheme for the direct simulation of the motion of particles in a Newtonian fluid. [5] As will be seen, the combined equation of motion is completely general; it applies to any fluid–particle system—even if the fluid is viscoelastic (though of course for viscoelastic fluids, the stress equations must also be included in the overall formulation). For simplicity of exposition, we assume that there is only one particle. The extension to the many-particle case is straightforward.

The key to deriving the combined equation of motion is the *combined velocity space*

$$\begin{aligned}\mathbb{V}_{\mathbf{u}_\Gamma}(t) = \{(\mathbf{v},\ \mathbf{V},\ \xi) | \mathbf{v} \in H^1(\Omega\backslash\overline{P(t)})^2,\ \mathbf{V} \in \mathbf{R}^2,\ \xi \in \mathbf{R},\\ \mathbf{v} = \mathbf{V} + \xi \times \mathbf{r} \text{ on } \partial P(t),\ \text{and } \mathbf{v} = \mathbf{u}_\Gamma(t) \text{ on } \Gamma\},\end{aligned}$$

which incorporates the constraint of rigid-body motion on $\partial P(t)$ (that is, the no-slip condition). In light of (3) and (4), we see that the combined fluid–particle velocity $(\mathbf{u},\ \mathbf{U},\ \omega)$ must lie in $\mathbb{V}_{\mathbf{u}_\Gamma}(t)$. To derive the combined equation of motion, we take a combined variation $(\mathbf{v},\ \mathbf{V},\ \xi)$ from the *combined variation space*

$$\begin{aligned}\mathbb{V}_0(t) = \{(\mathbf{v},\ \mathbf{V},\ \xi) | \mathbf{v} \in H^1(\Omega\backslash\overline{P(t)})^2,\ \mathbf{V} \in \mathbf{R}^2,\ \xi \in \mathbf{R},\\ \mathbf{v} = \mathbf{V} + \xi \times \mathbf{r} \text{ on } \partial P(t),\ \text{and } \mathbf{v} = \mathbf{0} \text{ on } \Gamma\},\end{aligned}$$

and perform the following symbolic operations:

$$\int_{\Omega\backslash\overline{P(t)}} (\text{Equation (1)}) \cdot \mathbf{v}\,\mathrm{d}\mathbf{x} + (\text{Equation (18)}) \cdot \mathbf{V} + (\text{Equation (7)})\xi.$$

After integrating the stress-divergence term by parts, and using the fact that $\mathbf{v} = \mathbf{V} + \xi \times \mathbf{r}$ on $\partial P(t)$, we find that the $\partial P(t)$ integrals arising from the integration by parts exactly cancel the hydrodynamic force and torque from the particle equations of motion, yielding the *combined equation of motion*:

$$\begin{aligned}&\int_{\Omega\backslash\overline{P(t)}} \rho_L\left(\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} - \mathbf{g}\right) \cdot \mathbf{v}\,\mathrm{d}\mathbf{x} + M\left(\frac{\mathrm{d}\mathbf{U}}{\mathrm{d}t} - \mathbf{g}\right) \cdot \mathbf{V} + I\frac{\mathrm{d}\omega}{\mathrm{d}t}\xi - \mathbf{F}' \cdot \mathbf{V}\\ &= -\int_{\Omega\backslash\overline{P(t)}} \boldsymbol{\sigma} : \mathbf{D}[\mathbf{v}]\,\mathrm{d}\mathbf{x}, \quad \text{for all } (\mathbf{v},\ \mathbf{V},\ \xi) \in \mathbb{V}_0. \qquad (21)\end{aligned}$$

[5] Hesla used (6) instead of (18).

The fluid pressure p is required to lie in the space

$$L_0^2(\Omega\backslash\overline{P(t)}) = \left\{ q \in L^2(\Omega\backslash\overline{P(t)}) \middle| \int_{\Omega\backslash\overline{P(t)}} q \, \mathrm{d}\mathbf{x} = 0 \right\}$$

in order to fix the undetermined additive constant. The weak formulation of the problem is completed by the weak form of the incompressibility constraint,

$$\int_{\Omega\backslash\overline{P(t)}} q\nabla\cdot\mathbf{u}\,\mathrm{d}\mathbf{x} = 0, \qquad \text{for all } q \in L^2(\Omega\backslash\overline{P(t)}). \tag{22}$$

Eqs. (21) and (22), together with (10) and (11), and the initial conditions (5), (8), (9), (12), and (13), form a good starting point for constructing finite-element schemes for the direct simulation of fluid–particle flows. (The fictitious-domain method derived below is based on a generalization of this formulation.) One such finite-element scheme, which uses a generalization of the θ operator-splitting scheme (Bristeau et al., 1987) for the time discretization, was developed by Hesla (1991).

Such finite-element schemes have at least two built-in advantages, owing to the fact that the hydrodynamic force and torque have been eliminated in the derivation of the combined equation of motion. The first is that the force and torque need not even be computed, thereby eliminating one step of the computation. The second—and more important—advantage is that these schemes are not subject to a numerical instability which can arise when the equations of fluid and particle motion are integrated as a coupled system with explicitly computed force and torque (Hu et al., 1992).

2.1.3. Discussion

Eq. (21) represents a virtual power principle for the combined fluid–particle system, and governs the evolution of the *total* system momentum—fluid plus particle. It is noteworthy that because it was derived without using the specific form of $\boldsymbol{\sigma}$, (21) holds for *any* fluid, even a viscoelastic one; in fact, it holds for *any continuous medium whatever*—for example, an elastic solid.

The cancellation of the $\partial P(t)$ integrals can be viewed as a consequence of Newton's third law. The hydrodynamic force and torque represent *internal* forces for the combined fluid–particle system, so cannot alter the total momentum of the system; they can only effect an *exchange* of momentum between fluid and particle.

When (17) is substituted into (21), the viscous dissipation term which arises involves $\mathbf{D}[\mathbf{u}]{:}\mathbf{D}[\mathbf{v}]$, rather than the more usual $\nabla\mathbf{u}{:}\nabla\mathbf{v}$. This is a consequence of the need to retain the complete stress tensor during the integration by parts so that the $\partial P(t)$ integrals will cancel the hydrodynamic force and torque terms from the particle equations of motion.

Finally, The Navier–Stokes equation and the equations of rigid-body motion can be recovered from (21) as special cases, by appropriate choices of $(\mathbf{v}, \mathbf{V}, \xi)\in\mathbb{V}_0(t)$.

2.2. A fictitious-domain formulation

As stated in Section 1.3.2, the basic idea of fictitious-domain methods, in the present setting, is to extend the problem from $\Omega\backslash\overline{P(t)}$ to all of Ω, while still forcing the solution to satisfy the no-slip condition on $\partial P(t)$. For simplicity of exposition, we continue to assume that there is only one particle. The extension to the many-particle case is straightforward.

Extending (22) to Ω is straightforward. The extension of (21) is carried out in two steps:

1. Obtain an analogous combined equation of motion for $P(t)$, and add it to (21) to produce a combined equation of motion for the whole of Ω;
2. Relax the constraint of rigid-body motion by removing it from the combined velocity spaces, and enforce it as a side constraint using a Lagrange multiplier.

2.2.1. Combined equation of motion for the extended domain

To obtain a combined equation of motion for $P(t)$ in a form suitable for combining with (21), it is necessary to enforce the constraint of rigid-body motion for both $\mathbf{u}$ and $\mathbf{v}$ *throughout* $P(t)$—not just on $\partial P(t)$. That is, we require

$$\mathbf{u} = \mathbf{U} + \omega \times \mathbf{r} \quad \text{in } P(t), \tag{23}$$

$$\mathbf{v} = \mathbf{V} + \xi \times \mathbf{r} \quad \text{in } P(t). \tag{24}$$

It can be checked that (23) implies

$$\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} = \frac{\mathrm{d}\mathbf{U}}{\mathrm{d}t} + \frac{\mathrm{d}\omega}{\mathrm{d}t} \times \mathbf{r} + \omega \times (\omega \times \mathbf{r}) \quad \text{in } P(t).$$

Forming the inner product of this with $\rho_\mathrm{d}\mathbf{v}$, integrating over $P(t)$, and using (24) together with the fact that

$$\int_{P(t)} \mathbf{r}\ \mathrm{d}\mathbf{x} = \mathbf{0}, \tag{25}$$

we obtain [6] (after some calculation)

$$\int_{P(t)} \rho_\mathrm{d} \frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} \cdot \mathbf{v}\,\mathrm{d}\mathbf{x} = M\frac{\mathrm{d}\mathbf{U}}{\mathrm{d}t} \cdot \mathbf{V} + I\frac{\mathrm{d}\omega}{\mathrm{d}t}\xi. \tag{26}$$

We also have, in light of (24) and (25),

$$\int_{P(t)} \rho_\mathrm{d}\mathbf{g} \cdot \mathbf{v}\,\mathrm{d}\mathbf{x} = M\mathbf{g} \cdot \mathbf{V}.$$

Subtracting this from (26), multiplying by $\rho_\mathrm{L}/\rho_\mathrm{d}$, and noting that $\mathbf{D}[\mathbf{v}] = \mathbf{0}$ in $P(t)$ because of (24), we obtain (after rearranging)

$$\int_{P(t)} \rho_\mathrm{L}\left(\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} - \mathbf{g}\right) \cdot \mathbf{v}\,\mathrm{d}\mathbf{x} - \frac{\rho_\mathrm{L}}{\rho_\mathrm{d}} M\left(\frac{\mathrm{d}\mathbf{U}}{\mathrm{d}t} - \mathbf{g}\right) \cdot \mathbf{V} - \frac{\rho_\mathrm{L}}{\rho_\mathrm{d}} I\frac{\mathrm{d}\omega}{\mathrm{d}t}\xi = -\int_{P(t)} \boldsymbol{\sigma} : \mathbf{D}[\mathbf{v}]\,\mathrm{d}\mathbf{x}. \tag{27}$$

[6] In three dimensions, the term $I\,\mathrm{d}\omega/\mathrm{d}t\ \xi$ would become $(\mathbf{I}\,\mathrm{d}\boldsymbol{\omega}/\mathrm{d}t + \boldsymbol{\omega} \times \mathbf{I}\boldsymbol{\omega}) \cdot \xi$.

To produce the combined weak equation of motion for the entire domain Ω, we extend the combined velocity and combined variation spaces to $P(t)$ using the extended rigid-motion constraint (24):

$$\begin{aligned}\widetilde{\mathbb{V}}_{\mathbf{u}_\Gamma}(t) = \{&(\mathbf{v},\ \mathbf{V},\ \xi) | \mathbf{v} \in H^1(\Omega)^2,\ \mathbf{V} \in \mathbf{R}^2,\ \xi \in \mathbf{R},\\ &\mathbf{v} = \mathbf{V} + \xi \times \mathbf{r} \text{ in } P(t),\ \text{and } \mathbf{v} = \mathbf{u}_\Gamma(t) \text{ on } \Gamma\},\end{aligned}$$

$$\begin{aligned}\widetilde{\mathbb{V}}_0(t) = \{&(\mathbf{v},\ \mathbf{V},\ \xi) | \mathbf{v} \in H^1(\Omega)^2,\ \mathbf{V} \in \mathbf{R}^2,\ \xi \in \mathbf{R},\\ &\mathbf{v} = \mathbf{V} + \xi \times \mathbf{r} \text{ in } P(t),\ \text{and } \mathbf{v} = \mathbf{0} \text{ on } \Gamma\}.\end{aligned}$$

(In light of (3) and (23), the extended combined fluid–particle velocity $(\mathbf{u},\ \mathbf{U},\ \omega)$ must lie in $\widetilde{\mathbb{V}}_{\mathbf{u}_\Gamma}(t)$.) Adding (27) to (21), we obtain the following combined weak equation of motion for the entire domain Ω:

$$\begin{aligned}&\int_\Omega \rho_\mathrm{L}\left(\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} - \mathbf{g}\right) \cdot \mathbf{v}\,\mathrm{d}\mathbf{x} + \left(1 - \frac{\rho_\mathrm{L}}{\rho_\mathrm{d}}\right)\left(M\left(\frac{\mathrm{d}\mathbf{U}}{\mathrm{d}t} - \mathbf{g}\right) \cdot \mathbf{V} + I\frac{\mathrm{d}\omega}{\mathrm{d}t}\xi\right) - \mathbf{F}' \cdot \mathbf{V}\\ &\quad = -\int_\Omega \boldsymbol{\sigma} : \mathbf{D}[\mathbf{v}]\,\mathrm{d}\mathbf{x}, \quad \text{for all } (\mathbf{v},\ \mathbf{V},\ \xi) \in \widetilde{\mathbb{V}}_0. \qquad (28)\end{aligned}$$

2.2.2. Relaxing the constraint of rigid-body motion

In (28), the solution function $\mathbf{u}$ and variation $\mathbf{v}$ are required to satisfy the (strong form of the) constraint of rigid-body motion throughout $P(t)$—that is, the constraint is enforced via the definition of the combined velocity spaces. We can relax the constraint by:

1. Removing it from the combined velocity spaces;
2. Enforcing it, in the weak sense, as a side constraint; and
3. Adding an appropriate distributed Lagrange multiplier term to the right-hand side.

This results—after using (14) and (17)—in the following weak formulation of the problem in the extended domain:

For a.e. $t > 0$, find $\mathbf{u} \in \mathbb{W}_{\mathbf{u}_\Gamma}$, $p \in L^2_0(\Omega)$, $\boldsymbol{\lambda} \in \Lambda(t)$, $\mathbf{U} \in \mathbf{R}^2$, and $\omega \in \mathbf{R}$ satisfying

$$\begin{aligned}&\int_\Omega \rho_\mathrm{L}\left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} - \mathbf{g}\right) \cdot \mathbf{v}\,\mathrm{d}\mathbf{x} - \int_\Omega p\nabla \cdot \mathbf{v}\,\mathrm{d}\mathbf{x} + \int_\Omega 2\eta\mathbf{D}[\mathbf{u}]:\mathbf{D}[\mathbf{v}]\,\mathrm{d}\mathbf{x}\\ &\quad + \left(1 - \frac{\rho_\mathrm{L}}{\rho_\mathrm{d}}\right)\left(M\left(\frac{\mathrm{d}\mathbf{U}}{\mathrm{d}t} - \mathbf{g}\right) \cdot \mathbf{V} + I\frac{\mathrm{d}\omega}{\mathrm{d}t}\xi\right) - \mathbf{F}' \cdot \mathbf{V} = \langle \boldsymbol{\lambda},\ \mathbf{v} - (\mathbf{V} + \xi \times \mathbf{r})\rangle_{P(t)}\\ &\qquad \text{for all } \mathbf{v} \in \mathbb{W}_0,\ \mathbf{V} \in \mathbf{R}^2,\ \text{and } \xi \in \mathbf{R}, \qquad (29)\end{aligned}$$

$$\int_\Omega q\nabla \cdot \mathbf{u}\,\mathrm{d}\mathbf{x} = 0 \quad \text{for all } q \in L^2(\Omega), \qquad (30)$$

$$\langle \boldsymbol{\mu},\ \mathbf{u} - (\mathbf{U} + \omega \times \mathbf{r})\rangle_{P(t)} = 0 \quad \text{for all } \boldsymbol{\mu} \in \Lambda(t), \qquad (31)$$

$$\mathbf{u}|_{t=0} = \mathbf{u}_0 \quad \text{in } \Omega, \tag{32}$$

as well as (10) and (11), and the initial conditions (8), (9), (12), and (13), where $\mathbf{u}_0$ satisfies the compatibility conditions

$$\begin{aligned} &\nabla \cdot \mathbf{u}_0 = 0 && \text{in } \Omega \backslash \overline{P(0)}, \\ &\mathbf{u}_0 = \mathbf{U}_0 + \omega_0 \times \mathbf{r}_0 && \text{in } P(0). \end{aligned}$$

Here $\mathbf{r}_0 = \mathbf{x} - \mathbf{X}(0)$,

$$\mathbb{W}_{\mathbf{u}_\Gamma} = \{\mathbf{v} \in H^1(\Omega)^2 | \mathbf{v} = \mathbf{u}_\Gamma(t) \text{ on } \Gamma\},$$

$$\mathbb{W}_0 = H_0^1(\Omega)^2,$$

$$L_0^2(\Omega) = \left\{ q \in L^2(\Omega) \middle| \int_\Omega q \,\mathrm{d}\mathbf{x} = 0 \right\},$$

and $\Lambda(t)$ is $H^1(P(t))^2$, with $\langle \cdot, \cdot \rangle_{P(t)}$ denoting an appropriate inner product. The above formulation is due to the first two authors.

2.2.3. Remarks

The velocity $\mathbf{u}$, initial velocity $\mathbf{u}_0$, and pressure p for the extended problem must of course coincide with the corresponding functions for the original problem, in $\Omega \backslash \overline{P(t)}$. This entitles us to use the same symbol for both the original and extended functions.

If $\mathbf{v} \equiv \mathbf{V} + \xi \times \mathbf{r}$ in $P(t)$, it is straightforward to check that (29) reduces to (21). Thus, the extended formulation is truly an extension of the weak formulation in $\Omega \backslash \overline{P(t)}$.

Since ρ_L and $\mathbf{g}$ are constant, the body-force term

$$\int_\Omega \rho_\mathrm{L} \mathbf{g} \cdot \mathbf{v} \,\mathrm{d}\mathbf{x}$$

in (29) can be absorbed into the pressure term. The ensuing discussion will reflect this change.

Since, in (29), $\mathbf{u}$ is divergence-free and satisfies a Dirichlet boundary condition on all of Γ, we have

$$2\int_\Omega \mathbf{D}[\mathbf{u}] : \mathbf{D}[\mathbf{v}] \,\mathrm{d}\mathbf{x} = \int_\Omega \nabla\mathbf{u} : \nabla\mathbf{v} \,\mathrm{d}\mathbf{x} \quad \text{for all } \mathbf{v} \in \mathbb{W}_0. \tag{33}$$

This is a substantial simplification from the computational point of view—another advantage of the fictitious domain method developed here.

For the inner product $\langle \cdot, \cdot \rangle_{P(t)}$ on $\Lambda(t)$, we can use the standard inner product on $H^1(P(t))^2$. Alternatively, we can use an equivalent inner product which is more closely related to the physics of the problem. One possibility is

$$\langle \boldsymbol{\mu}, \ \mathbf{v} \rangle_{P(t)} = \int_{P(t)} (\boldsymbol{\mu} \cdot \mathbf{v} + \alpha \nabla\boldsymbol{\mu} : \nabla\mathbf{v}) \,\mathrm{d}\mathbf{x}$$

where α is a scaling factor. If all the particles have the same diameter d, an obvious choice for

α, on dimensional grounds, is d^2, since d is the most important characteristic length in the problem. Another natural H^1 inner product, well suited to rigid-body motion, is

$$\langle \boldsymbol{\mu}, \mathbf{v} \rangle_{P(t)} = \int_{P(t)} (\boldsymbol{\mu} \cdot \mathbf{v} + 2\alpha \mathbf{D}[\boldsymbol{\mu}] : \mathbf{D}[\mathbf{v}])\, \mathrm{d}\mathbf{x}.$$

Neither of these inner products has yet been tested in the actual code.

2.2.4. Interpretation of the distributed Lagrange multiplier

It can be shown that—if the standard H^1 inner product is used for $\Lambda(t)$—the combined equation of motion (29) implies the following conservation equation for linear momentum inside $P(t)$:

$$\rho_\mathrm{L} \frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} = \rho_\mathrm{L}\mathbf{g} + (\boldsymbol{\lambda} - \nabla^2\boldsymbol{\lambda}) + \nabla \cdot \boldsymbol{\sigma} \quad \text{in } P(t), \tag{34}$$

together with the Neumann boundary condition

$$(\hat{\mathbf{n}} \cdot \nabla)\boldsymbol{\lambda} = \hat{\mathbf{n}} \cdot (\boldsymbol{\sigma}_\mathrm{d} - \boldsymbol{\sigma}_\mathrm{L}) \quad \text{on } \partial P(t) \tag{35}$$

where $\boldsymbol{\sigma}_\mathrm{d}$ and $\boldsymbol{\sigma}_\mathrm{L}$ are the stress inside and outside $\partial P(t)$, respectively. From this, we see that $(\boldsymbol{\lambda} - \nabla^2\boldsymbol{\lambda})$ can be interpreted as the additional body force (per unit volume) required to maintain the rigid-body motion in $P(t)$.

It is interesting to note that we can obtain a third-order partial differential equation for $\boldsymbol{\lambda}$ alone by substituting the strong form

$$\mathbf{u} = \mathbf{U} + \omega \times \mathbf{r} \quad \text{in } P(t)$$

of (31), together with (17) into (34) then taking the curl. This results (since $\mathbf{g}$ is constant) in

$$2\rho_\mathrm{L} \frac{\mathrm{d}\omega}{\mathrm{d}t} = \nabla \times \boldsymbol{\lambda} - \nabla^2(\nabla \times \boldsymbol{\lambda}). \tag{36}$$

3. Computational scheme

3.1. Finite-element discretization

To discretize the problem in space, we introduce a regular finite-element triangulation $\mathcal{T}_h$ of $\bar{\Omega}$ for $\mathbf{u}$, where h is the mesh size [see Fig. 3(a)], and a (regular) "twice-coarser" triangulation $\mathcal{T}_{2h}$ for p [see Fig. 3(b)]. [7] We then define the following finite dimensional spaces approximating $\mathbb{W}_{\mathbf{u}_\Gamma}$, $\mathbb{W}_0$, $L^2(\Omega)$, $L_0^2(\Omega)$:

[7] In actual practice $\mathcal{T}_{2h}$ is constructed first; then $\mathcal{T}_h$ is constructed by joining the midpoints of the edges of $\mathcal{T}_{2h}$, dividing each triangle of $\mathcal{T}_{2h}$ into 4 similar subtriangles.

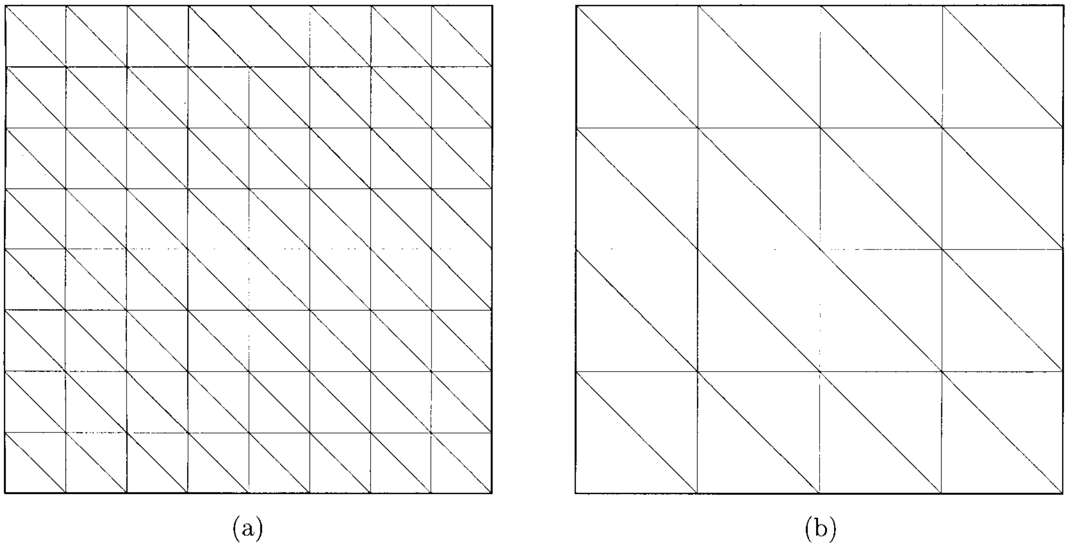

Fig. 3. (a) Velocity mesh; (b) pressure mesh.

$$\mathbb{W}_{\mathbf{u}_{\Gamma,h}} = \{\mathbf{v}_h \in C^0(\bar{\Omega})^2 \,|\, \mathbf{v}_h|_T \in P_1 \times P_1 \text{ for all } T \in \mathcal{T}_h,\ \mathbf{v}_h = \mathbf{u}_{\Gamma,h} \text{ on } \Gamma\},$$

$$\mathbb{W}_{0,h} = \{\mathbf{v}_h \in C^0(\bar{\Omega})^2 \,|\, \mathbf{v}_h|_T \in P_1 \times P_1 \text{ for all } T \in \mathcal{T}_h,\ \mathbf{v}_h = \mathbf{0} \text{ on } \Gamma\},$$

$$L_h^2 = \{q_h \in C^0(\bar{\Omega}) \,|\, q_h|_T \in P_1 \text{ for all } T \in \mathcal{T}_{2h}\},$$

$$L_{0,h}^2 = \left\{ q_h \in L_h^2 \,\middle|\, \int_\Omega q_h \,\mathrm{d}\mathbf{x} = 0 \right\}.$$

Here $\mathbf{u}_{\Gamma,h}$ is an approximation of $\mathbf{u}_\Gamma$ satisfying

$$\int_\Gamma \mathbf{u}_{\Gamma,h} \cdot \hat{\mathbf{n}} \,\mathrm{d}s = 0,$$

and P_1 is the space of polynomials in two variables of degree 1 or less.

For functions in $\Lambda(t)$, we must use a coarser triangulation to ensure stability (see Section 3.1.1. for details). Let $\mathcal{T}_{P_h}(t)$ be a finite-element triangulation of $\overline{P_h(t)}$, a polygonal domain inscribed in $\overline{P(t)}$ (see Fig. 4). Then, a finite-dimensional space approximating $\Lambda(t)$ is

$$\Lambda_h(t) = \{\boldsymbol{\mu}_h \in C^0(\overline{P_h(t)})^2 \,|\, \boldsymbol{\mu}_h|_T \in P_1 \times P_1 \text{ for all } T \in \mathcal{T}_{P_h(t)}\}.$$

This space was used in Glowinski et al. (1998). Another possibility is to choose M points in $\overline{P(t)}$, $\mathbf{x}_1, \ldots, \mathbf{x}_M$, which cover $\overline{P(t)}$ uniformly, and define

$$\Lambda_h(t) = \left\{ \boldsymbol{\mu}_h \,\middle|\, \boldsymbol{\mu}_h = \sum_{i=1}^{M} \boldsymbol{\mu}_{h,i}\delta(\mathbf{x} - \mathbf{x}_i),\ \boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_M \in \mathbf{R}^2 \right\}, \tag{37}$$

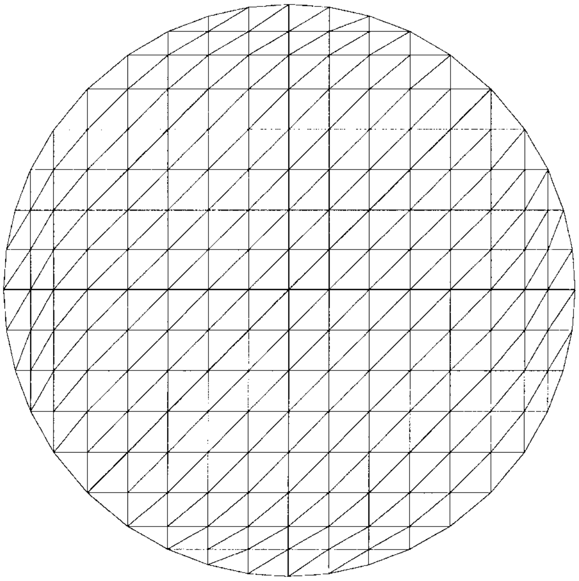

Fig. 4. Triangulation of $\overline{P_h(t)}$, if $P(t)$ is circular.

where $\delta(\cdot)$ is the Dirac delta function at $\mathbf{x} = 0$; then define a discrete L^2 "inner product" $\langle \cdot, \cdot \rangle_{P_h(t)}$ by

$$\langle \boldsymbol{\mu}_h,\ \mathbf{v}_h \rangle_{P_h(t)} = \sum_{i=1}^{M} \boldsymbol{\mu}_{h,i} \cdot \mathbf{v}_h(\mathbf{x}_i), \quad \text{for all } \boldsymbol{\mu}_h \in \Lambda_l(t), \mathbf{v}_h \in \mathbb{W}_{\mathbf{u}_{\Gamma,h}} \text{ or } \mathbb{W}_{0,h}. \tag{38}$$

Using this "inner product" is equivalent to enforcing the constraint of rigid-body motion inside $P(t)$ with a *collocation method.* A similar technique is used by Bertrand et al. (1997) to enforce a Dirichlet boundary condition. Other approaches are possible, and will be investigated in the future.

Using these finite-dimensional spaces, we obtain the following finite-element approximation of the problem (29)–(32):

Find $\mathbf{u}_h \in \mathbb{W}_{\mathbf{u}_{\Gamma,h}}$, $p_h \in L^2_{0,h}$, $\boldsymbol{\lambda}_h \in \Lambda_h(t)$, $\mathbf{U} \in \mathbf{R}^2$, and $\omega \in \mathbf{R}$ satisfying

$$\begin{aligned}
&\int_\Omega \rho_\mathrm{L}\left(\frac{\mathrm{d}\mathbf{u}_h}{\mathrm{d}t} + (\mathbf{u}_h \cdot \nabla)\mathbf{u}_h\right) \cdot \mathbf{v}_h \,\mathrm{d}\mathbf{x} - \int_\Omega p_h \nabla \cdot \mathbf{v}_h \,\mathrm{d}\mathbf{x} + \int_\Omega 2\eta \mathbf{D}[\mathbf{u}_h] : \mathbf{D}[\mathbf{v}_h] \,\mathrm{d}\mathbf{x} \\
&\quad + \left(1 - \frac{\rho_\mathrm{L}}{\rho_\mathrm{d}}\right)\left(M\left(\frac{\mathrm{d}\mathbf{U}}{\mathrm{d}t} - \mathbf{g}\right) \cdot \mathbf{V} + I\frac{\mathrm{d}\omega}{\mathrm{d}t}\xi\right) - \mathbf{F}' \cdot \mathbf{V} = \langle \boldsymbol{\lambda}_h,\ \mathbf{v}_h - (\mathbf{V} + \xi \times \mathbf{r}) \rangle_{P_h(t)} \\
&\qquad \text{for all } \mathbf{v}_h \in \mathbb{W}_{0,h},\ \mathbf{V} \in \mathbf{R}^2,\ \text{and } \xi \in \mathbf{R},
\end{aligned} \tag{39}$$

$$\int_\Omega q_h \nabla \cdot \mathbf{u}_h \,\mathrm{d}\mathbf{x} = 0 \quad \text{for all } q_h \in L^2_h, \tag{40}$$

$$\langle \boldsymbol{\mu}_h,\ \mathbf{u}_h - (\mathbf{U} + \omega \times \mathbf{r})\rangle_{P_h(t)} = 0 \quad \text{for all } \boldsymbol{\mu}_h \in \Lambda_h(t), \tag{41}$$

$$\mathbf{u}_h|_{t=0} = \mathbf{u}_{0,h} \quad \text{in } \Omega, \tag{42}$$

as well as (10) and (11), and the initial conditions (8), (9), (12), and (13), where $\mathbf{u}_{0,h}$ is an approximation of $\mathbf{u}_0$ satisfying the compatibility conditions

$$\int_\Omega q_h \nabla \cdot \mathbf{u}_{0,h}\,\mathrm{d}\mathbf{x} = 0 \qquad \text{for all } q_h \in L_h^2,$$

$$\langle \boldsymbol{\mu}_h,\ \mathbf{u}_{0,h} - (\mathbf{U}_0 + \omega_0 \times \mathbf{r}_0)\rangle_{P_h(0)} = 0 \quad \text{for all } \boldsymbol{\mu}_h \in \Lambda_h(0).$$

3.1.1. Remarks

The viscous dissipation term

$$\int_\Omega 2\eta \mathbf{D}[\mathbf{u}_h] : \mathbf{D}[\mathbf{v}_h]\,\mathrm{d}\mathbf{x} \tag{43}$$

in (39) inextricably couples the x and y equations of motion. (This is inevitable since the fluid and particle equations of motion are coupled in their strong form to begin with, through (4) and through the hydrodynamic force and torque.) This doubles the size of the matrix and thus quadruples the work required for each matrix–vector multiply. However, this is offset by the fact that only one system needs to be solved, instead of two. Furthermore, we can use preconditions based on the *point-block ordering* of the velocity unknowns. That is, if $\mathbf{u} = (u, v)$, we order the velocity unknowns u_1, v_1, u_2, v_2, ... —always dealing with the x and y components of $\mathbf{u}$ at a single node *together*, as a single entity. This means, for example, that instead of using a diagonal preconditioner, we would use a *block diagonal* preconditioner, based on the 2×2 point blocks. Since the coupling between the x and y components of velocity at a single node is generally tighter than that between velocity components at separate nodes, this strategy should significantly reduce the number of iterations required for convergence of iterative solution techniques.

When $\mathbf{u}$ satisfies a Dirichlet boundary condition on the *entire* outer boundary Γ, the viscous dissipation term (43) can be simplified considerably: It can be written in the form

$$\int_\Omega \eta \nabla \mathbf{u}_h : \nabla \mathbf{v}_h\,\mathrm{d}\mathbf{x},$$

in view of (33).

The mesh sizes h_Ω and $h_{P(t)}$ for $\mathcal{T}_h$ and $\mathcal{T}_{P_h}(t)$, respectively, must be related by a condition of the form

$$h_\Omega \sim \kappa h_{P(t)}, \tag{44}$$

for some κ in the interval (0.5, 1). This follows from general results on the approximation of *generalized saddle-point problems* (that is, problems involving Lagrange multipliers), (29)–(32) being a typical example of such a problem. For a thorough discussion of the approximation of steady saddle-point problems see, for example, Babuska (1973) and Brezzi (1974). Without going into details, we simply state that the conditioning of the algebraic systems to be solved at each time step is enhanced by requiring

$$h_\Omega \ll h_{P(t)},$$

while accuracy is enhanced by requiring

$$h_{P(t)} \ll h_\Omega.$$

Relation (44) represents a compromise between these two alternatives.

3.2. Time discretization by operator splitting

Following Chorin (1967, 1968, 1973), most "modern" Navier–Stokes solvers are based on operator-splitting algorithms, in order to enforce the incompressibility condition via a Stokes solver or an L^2-projection method (Glowinski and Pironneau, 1992; Turek, 1996). This approach can be applied to the initial value problem (39)–(42) to decouple its three principal numerical difficulties:

1. The incompressibility condition, and the related unknown pressure p_h;
2. The advection and diffusion terms;
3. The constraint of rigid-body motion in $P_h(t)$, and the related distributed Lagrange multiplier $\boldsymbol{\lambda}_h$.

Each of these corresponds to a specific operator. (The first and last are essentially *projection operators*.)

Problem (39)–(42) is a particular case of the abstract initial value problem

$$\frac{\mathrm{d}\phi}{\mathrm{d}t} + A_1(\phi) + A_2(\phi) + A_3(\phi) = f,$$

$$\phi(0) = \phi_0,$$

where the operators A_1, A_2, and A_3, can be multiple-valued. Among the many operator-splitting methods which could be employed to solve this problem, we advocate the following one (analyzed in, for example, Marchuk, 1990). It is only first-order accurate, but this is compensated for by its good stability and robustness properties. In the following, Δt is a time step.

Fractional-step scheme à la Marchuk–Yanenko

Set $\phi^0 = \phi_0$.

For $n = 0, 1, \ldots$, compute an approximation ϕ^{n+1} to $\phi((n+1)\Delta t)$ by solving the three successive problems

$$\frac{\phi^{n+1/3} - \phi^n}{\Delta t} + A_1(\phi^{n+1/3}) = f_1^{n+1},$$

$$\frac{\phi^{n+2/3} - \phi^{n+1/3}}{\Delta t} + A_2(\phi^{n+2/3}) = f_2^{n+1},$$

$$\frac{\phi^{n+1} - \phi^{n+2/3}}{\Delta t} + A_3(\phi^{n+1}) = f_3^{n+1},$$

where $f_1^{n+1} + f_2^{n+1} + f_3^{n+1} = f((n+1)\Delta t)$.

In applying this scheme to (39)–(42), we assume for simplicity that the particles are circular so that (11) and (13) may be ignored as stated in Section 2.1.1.1. The resulting algorithm can easily be generalized to arbitrary shaped particles. With $0 \leq \alpha, \beta \leq 1$ and $\alpha + \beta = 1$ we obtain, after dropping some of the subscripts h:

Set $\mathbf{u}^0 = \mathbf{u}_{0,h}$, $\mathbf{U}^0 = \mathbf{U}_0$, $\omega^0 = \omega_0$, $\mathbf{X}^0 = \mathbf{X}_0$.
For $n = 0, 1, \ldots$, assuming $\mathbf{u}^n$, $\mathbf{U}^n$, ω^n, and $\mathbf{X}^n$ are known, proceed as follows:
Find $\mathbf{u}^{n+1/3} \in \mathbb{W}_{\mathbf{u}_{\Gamma,h}}^{n+1}$ and $p^{n+1/3} \in L^2{}_{0,h}$ satisfying

$$\rho_L \int_\Omega \frac{\mathbf{u}^{n+1/3} - \mathbf{u}^n}{\Delta t} \cdot \mathbf{v}\, d\mathbf{x} - \int_\Omega p^{n+1/3} \nabla \cdot \mathbf{v}\, d\mathbf{x} = 0 \quad \text{for all } \mathbf{v} \in \mathbb{W}_{0,h}, \tag{45}$$

$$\int_\Omega q \nabla \cdot \mathbf{u}^{n+1/3}\, d\mathbf{x} = 0 \quad \text{for all } q \in L_h^2. \tag{46}$$

Find $\mathbf{u}^{n+2/3} \in \mathbb{W}_{\mathbf{u}_{\Gamma,h}}^{n+1}$ satisfying

$$\rho_L \int_\Omega \frac{\mathbf{u}^{n+2/3} - \mathbf{u}^{n+1/3}}{\Delta t} \cdot \mathbf{v}\, d\mathbf{x} + \rho_L \int_\Omega (\mathbf{u}^{n+1/3} \cdot \nabla)\mathbf{u}^{n+2/3} \cdot \mathbf{v}\, d\mathbf{x}$$
$$+ 2\alpha\eta \int_\Omega \mathbf{D}[\mathbf{u}^{n+2/3}] : \mathbf{D}[\mathbf{v}]\, d\mathbf{x} = 0 \quad \text{for all } \mathbf{v} \in \mathbb{W}_{0,h}. \tag{47}$$

Compute $\mathbf{U}^{n+2/3}$ and $\mathbf{X}^{n+2/3}$ using the *prediction procedure*
Set $\mathbf{U}^{n,0} = \mathbf{U}^n$, $\mathbf{X}^{n,0} = \mathbf{X}^n$.
do $k = 1, K$

$$\mathbf{U}^{*\mathrm{n,k}} = \mathbf{U}^{n,k-1} + \left(\mathbf{g} + \left(1 - \frac{\rho_L}{\rho_d}\right)^{-1} M^{-1}\mathbf{F}'(\mathbf{X}^{n,k-1})\right)\frac{\Delta t}{K}, \tag{48}$$

$$\mathbf{X}^{*\mathrm{n,k}} = \mathbf{X}^{n,k-1} + \left(\frac{\mathbf{U}^{n,k-1} + \mathbf{U}^{*\mathrm{n,k}}}{2}\right)\frac{\Delta t}{K}, \tag{49}$$

$$\mathbf{U}^{n,k} = \mathbf{U}^{n,k-1} + \left(\mathbf{g} + \left(1 - \frac{\rho_L}{\rho_d}\right)^{-1} M^{-1} \frac{\mathbf{F}'(\mathbf{X}^{n,k-1}) + \mathbf{F}'(\mathbf{X}^{*\mathrm{n,k}})}{2}\right)\frac{\Delta t}{K}, \tag{50}$$

$$\mathbf{X}^{n,k} = \mathbf{X}^{n,k-1} + \left(\frac{\mathbf{U}^{n,k-1} + \mathbf{U}^{n,k}}{2}\right)\frac{\Delta t}{K}, \tag{51}$$

end do
Set $\mathbf{U}^{n+2.3} = \mathbf{U}^{n,K}$, $\mathbf{X}^{n+2/3} = \mathbf{X}^{n,K}$.

Find $\mathbf{u}^{n+1} \in \mathbb{W}^{n+1}_{\mathbf{u}_{\Gamma,h}}$, $\mathbf{U}^{n+1} \in \mathbf{R}^2$, $\omega^{n+1} \in \mathbf{R}$, and $\boldsymbol{\lambda}^{n+1} \in \Lambda_h^{n+2/3}$ satisfying

$$\begin{aligned}
&\rho_L \int_\Omega \frac{\mathbf{u}^{n+1} - \mathbf{u}^{n+2/3}}{\Delta t} \cdot \mathbf{v}\, d\mathbf{x} + 2\beta\eta \int_\Omega \mathbf{D}[\mathbf{u}^{n+1}] : \mathbf{D}[\mathbf{v}]\, d\mathbf{x} \\
&\quad + \left(1 - \frac{\rho_L}{\rho_d}\right)\left(M \frac{\mathbf{U}^{n+1} - \mathbf{U}^{n+2/3}}{\Delta t} \cdot \mathbf{V} + I \frac{\omega^{n+1} - \omega^n}{\Delta t} \xi\right) \\
&\quad = \langle \boldsymbol{\lambda}^{n+1},\ \mathbf{v} - (\mathbf{V} + \xi \times \mathbf{r}^{n+2/3})\rangle_{P_h^{n+2/3}} \\
&\qquad\qquad \text{for all } \mathbf{v} \in \mathbb{W}_{0,h},\ \mathbf{V} \in \mathbf{R}^2,\ \text{and } \xi \in \mathbf{R},
\end{aligned} \tag{52}$$

$$\langle \boldsymbol{\mu},\ \mathbf{u}^{n+1} - (\mathbf{U}^{n+1} + \omega^{n+1} \times \mathbf{r}^{n+2/3})\rangle_{P_h^{n+2/3}} = 0 \quad \text{for all } \boldsymbol{\mu} \in \Lambda_h^{n+2/3}. \tag{53}$$

Compute $\mathbf{X}^{n+1}$ using the *correction procedure*
Set $\mathbf{X}^{n+1,0} = \mathbf{X}^n$.
do $k = 1, K$

$$\mathbf{X}^{*\mathrm{n+1,k}} = \mathbf{X}^{n+1,k-1} + \left(\frac{\mathbf{U}^n + \mathbf{U}^{n+1}}{2}\right)\frac{\Delta t}{K}, \tag{54}$$

$$\mathbf{X}^{n+1,k} = \mathbf{X}^{*\mathrm{n+1,k}} + \left(1 - \frac{\rho_L}{\rho_d}\right)^{-1} M^{-1}\left(\frac{\mathbf{F}'(\mathbf{X}^{n+1,k-1}) + \mathbf{F}'(\mathbf{X}^{*\mathrm{n+1,k}})}{2}\right)\frac{(\Delta t)^2}{2K^2}, \tag{55}$$

end do
Set $\mathbf{X}^{n+1} = \mathbf{X}^{n+1,K}$.

3.2.1. Remarks

In (45)–(55), $\mathbb{W}^{n+1}_{\mathbf{u}_{\Gamma,h}} = \mathbb{W}_{\mathbf{u}_{\Gamma((n+1)\Delta t),h}}$, $\Lambda_h^{n+s} = \Lambda_h((n+s)\Delta t)$, $P_h^{n+s} = P_h((n+s)\Delta t)$, and $\mathbf{r}^{n+s} = \mathbf{x} - \mathbf{X}^{n+s}$. In (48)–(51), we predict the particle center-of-mass position, and use it in (52), (53). Then, in (54), (55), we correct the particle center-of-mass position. With operator splitting, we can use a smaller time step in these prediction and correction procedures than we use in the rest of the algorithm. In the numerical simulations presented in this article, we use $K = 10$ in (48)–(51), and (54), (55).

Since the advecting field $\mathbf{u}^{n+1/3}$ in (47) is divergence-free (actually, *quasi* divergence-free because of the space discretization), the linearization of the advection term has very little effect on the overall stability of the splitting scheme. Actually, when one uses the *method of characteristics* to treat the advection (as in, for example, Pironneau, 1989) the advecting field is a divergence free velocity field computed at a previous step by either a Stokes solver for velocity or a Poisson solver for pressure.

As shown, for example, in Marchuk (1990) and LeVeque and Oliger (1983) (see also Dean and Glowinski, 1997) the Marchuk–Yanenko fractional step scheme employed here can be made second-order accurate without losing its good stability and robustness properties. This higher accuracy can be obtained by using, for example, a symmetrization procedure and then, for each fractional step, a stiff A-stable, second-order accurate, Runge–Kutta scheme (instead of the backward Euler scheme used here). See, for example, Dean and Glowinski (1997) for more details. These second-order accurate fractional step schemes are clearly more complicated to implement than the one used in this article. Also, it is very likely that the main source of numerical errors is the space discretization since, in order to capture the complicated dynamics of the flows discussed here, we must use small time steps. Nevertheless, we intend to make a systematic comparison between the first-order and second-order accurate fractional step schemes in the near future.

3.3. Solution of subproblems (45), (46), (47), and (52), (53)

By inspection of (45), (46), it is clear that $\mathbf{u}^{n+1/3}$ is the $L^2(\Omega)$-projection of $\mathbf{u}^n$ on the subspace

$$\left\{\mathbf{v}_h \in \mathbb{W}^{n+1}_{\mathbf{u}_{\Gamma,h}} \middle| \int_\Omega q_h \nabla \cdot \mathbf{v}_h \, \mathrm{d}\mathbf{x} = 0 \text{ for all } q_h \in L^2_h \right\},$$

with $p^{n+1/3}$ the corresponding Lagrange multiplier in $L^2_{0,h}$. The pair $\{\mathbf{u}^{n+1/3}, p^{n+1/3}\}$ is *unique*. To compute $\{\mathbf{u}^{n+1/3}, p^{n+1/3}\}$, we can use a Uzawa/conjugate-gradient algorithm operating in $L^2_{0,h}$ equipped with the inner product

$$\{q, q'\} \mapsto \int_\Omega \nabla q \cdot \nabla q' \, \mathrm{d}\mathbf{x}.$$

This gives an algorithm preconditioned by the discrete equivalent of $-\nabla^2$ with homogeneous Neumann boundary condition (Glowinski et al., 1997b,). This algorithm is *very* easy to implement and seems to have excellent convergence properties.

If $\alpha > 0$, (47) is a classical advection-diffusion problem. It can be easily solved, for example, by a least-squares/conjugate-gradient algorithm like those discussed in Glowinski (1984).

The solution of (52), (53) can be computed by algorithms similar to those for elliptic problems (Glowinski et al., 1994a), with the additional difficulty that there are now three additional equations—the ones used to compute the translational and angular velocity of the particle. Problem (52), (53) has the following form.

Find $\mathbf{u} \in \mathbb{W}_{\mathbf{u}_{\Gamma,h}}$, $\mathbf{U} \in \mathbf{R}^2$, $\omega \in \mathbf{R}$, $\boldsymbol{\lambda} \in \Lambda_h$ satisfying

$$\alpha \int_\Omega \mathbf{u} \cdot \mathbf{v} \, \mathrm{d}\mathbf{x} + 2\eta \int_\Omega \mathbf{D}[\mathbf{u}] : \mathbf{D}[\mathbf{v}] \, \mathrm{d}\mathbf{x} + \left(1 - \frac{\rho_\mathrm{L}}{\rho_\mathrm{d}}\right)\left(M \frac{\mathbf{U} - \mathbf{U}_0}{\Delta t} \cdot \mathbf{V} + I \frac{\omega - \omega_0}{\Delta t} \xi\right)$$

$$\int_\Omega \mathbf{f} \cdot \mathbf{v} \, \mathrm{d}\mathbf{x} + \langle \boldsymbol{\lambda}, \ \mathbf{v} - (\mathbf{V} + \xi \times \mathbf{r}) \rangle_{P_h} \quad \text{for all } \mathbf{v} \in \mathbb{W}_{0,h}, \ \mathbf{V} \in \mathbf{R}^2, \ \text{and } \xi \in \mathbf{R}, \tag{56}$$

$$\langle \boldsymbol{\mu}, \ \mathbf{u} - (\mathbf{U} + \omega \times \mathbf{r}) \rangle_{P_h} = 0 \quad \text{for all } \boldsymbol{\mu} \in \Lambda_h, \tag{57}$$

where the center of mass $\mathbf{X}$ of particle P_h is assumed known, and $\mathbb{W}_{\mathbf{u}_{\Gamma,h}} = \mathbb{W}^{n+1}_{\mathbf{u}_{\Gamma,h}}$. A conjugate-gradient algorithm for solving (56), (57) is the following:

Step 0: Initialization

Assume $\boldsymbol{\lambda}^0 \in \Lambda_h$ is given.
Find $\mathbf{u}^0 \in \mathbb{W}_{\mathbf{u}_{\Gamma,h}}$, $\mathbf{U}^0 \in \mathbf{R}^2$, and $\omega^0 \in \mathbf{R}$ satisfying

$$\alpha \int_\Omega \mathbf{u}^0 \cdot \mathbf{v} \, \mathrm{d}\mathbf{x} + 2\eta \int_\Omega \mathbf{D}[\mathbf{u}^0] : \mathbf{D}[\mathbf{v}] \, \mathrm{d}\mathbf{x} = \int_\Omega \mathbf{f} \cdot \mathbf{v} \, \mathrm{d}\mathbf{x} + \langle \boldsymbol{\lambda}^0, \ \mathbf{v} \rangle_{P_h} \quad \text{for all } \mathbf{v} \in \mathbb{W}_{0,h}, \tag{58}$$

$$\left(1 - \frac{\rho_\mathrm{L}}{\rho_\mathrm{d}}\right) M \frac{\mathbf{U}^0 - \mathbf{U}_0}{\Delta t} \cdot \mathbf{V} + \langle \boldsymbol{\lambda}^0, \ \mathbf{V} \rangle_{P_h} = 0 \quad \text{for all } \mathbf{V} \in \mathbf{R}^2, \tag{59}$$

$$\left(1 - \frac{\rho_\mathrm{L}}{\rho_\mathrm{d}}\right) I \frac{\omega^0 - \omega_0}{\Delta t} \xi + \langle \boldsymbol{\lambda}^0, \ \xi \times \mathbf{r} \rangle_{P_h} = 0 \quad \text{for all } \xi \in \mathbf{R}. \tag{60}$$

Find $\mathbf{g}^0 \in \Lambda_h$ satisfying

$$\langle \boldsymbol{\mu}, \ \mathbf{g}^0 \rangle_{P_h} = \langle \boldsymbol{\mu}, \ \mathbf{u}^0 - (\mathbf{U}^0 + \omega^0 \times \mathbf{r}) \rangle_{P_h} \quad \text{for all } \boldsymbol{\mu} \in \Lambda_h. \tag{61}$$

Set $\mathbf{w}^0 = \mathbf{g}^0$.
For $m = 0, 1, \ldots$, assuming $\mathbf{u}^m$, $\mathbf{U}^m$, ω^m, $\boldsymbol{\lambda}^m$, $\mathbf{g}^m$, and $\mathbf{w}^m$ are known, compute $\mathbf{u}^{m+1}$, $\mathbf{U}^{m+1}$, ω^{m+1}, $\boldsymbol{\lambda}^{m+1}$, $\mathbf{g}^{m+1}$, and $\mathbf{w}^{m+1}$ as follows:

Step 1: Descent

Find $\bar{\mathbf{u}}^m \in \mathbb{W}_{0,h}$, $\bar{\mathbf{U}}^m \in \mathbf{R}^2$, and $\bar{\omega}^m \in \mathbf{R}$ satisfying

$$\alpha \int_\Omega \bar{\mathbf{u}}^m \cdot \mathbf{v} \, \mathrm{d}\mathbf{x} + 2\eta \int_\Omega \mathbf{D}[\bar{\mathbf{u}}^m] : \mathbf{D}[\mathbf{v}] \, \mathrm{d}\mathbf{x} = \langle \mathbf{w}^m, \ \mathbf{v} \rangle_{P_h} \quad \text{for all } \mathbf{v} \in \mathbb{W}_{0,h}, \tag{62}$$

$$\left(1 - \frac{\rho_\mathrm{L}}{\rho_\mathrm{d}}\right) \frac{M}{\Delta t} \bar{\mathbf{U}}^m \cdot \mathbf{V} + \langle \mathbf{w}^m, \ \mathbf{V} \rangle_{P_h} = 0 \quad \text{for all } \mathbf{V} \in \mathbf{R}^2, \tag{63}$$

$$\left(1-\frac{\rho_{\mathrm{L}}}{\rho_{\mathrm{d}}}\right)\frac{I}{\Delta t}\bar{\omega}^m\xi+\langle\mathbf{w}^m,\ \xi\times\mathbf{r}\rangle_{P_h}=0\quad\text{for all }\xi\in\mathbf{R}. \tag{64}$$

Find $\bar{\mathbf{g}}^m\in\Lambda_h$ satisfying

$$\langle\boldsymbol{\mu},\ \bar{\mathbf{g}}^m\rangle_{P_h}=\langle\boldsymbol{\mu},\ \bar{\mathbf{u}}^m-(\bar{\mathbf{U}}^m+\bar{\omega}^m\times\mathbf{r})\rangle_{P_h}\quad\text{for all }\boldsymbol{\mu}\in\Lambda_h. \tag{65}$$

Set

$$\rho_m=\frac{\langle\mathbf{g}^m,\ \mathbf{g}^m\rangle_{P_h}}{\langle\mathbf{w}^m,\ \bar{\mathbf{g}}^m\rangle_{P_h}}, \tag{66}$$

$$\boldsymbol{\lambda}^{m+1}=\boldsymbol{\lambda}^m-\rho_m\mathbf{w}^m, \tag{67}$$

$$\mathbf{u}^{m+1}=\mathbf{u}^m-\rho_m\bar{\mathbf{u}}^m, \tag{68}$$

$$\mathbf{U}^{m+1}=\mathbf{U}^m-\rho_m\bar{\mathbf{U}}^m, \tag{69}$$

$$\omega^{m+1}=\omega^m-\rho_m\bar{\omega}^m, \tag{70}$$

$$\mathbf{g}^{m+1}=\mathbf{g}^m-\rho_m\bar{\mathbf{g}}^m. \tag{71}$$

Step 2: Convergence test/new descent direction

If

$$\frac{\langle\mathbf{g}^{m+1},\ \mathbf{g}^{m+1}\rangle_{P_h}}{\langle\mathbf{g}^0,\ \mathbf{g}^0\rangle_{P_h}}\leq\epsilon$$

take $\mathbf{u}=\mathbf{u}^{m+1}$, $\mathbf{U}=\mathbf{U}^{m+1}$, $\omega=\omega^{m+1}$, and $\boldsymbol{\lambda}=\boldsymbol{\lambda}^{m+1}$.

Otherwise, set

$$\gamma_m=\frac{\langle\mathbf{g}^{m+1},\ \mathbf{g}^{m+1}\rangle_{P_h}}{\langle\mathbf{g}^m,\ \mathbf{g}^m\rangle_{P_h}}, \tag{72}$$

$$\mathbf{w}^{m+1}=\mathbf{g}^{m+1}+\gamma_m\mathbf{w}^m, \tag{73}$$

$$m=m+1, \tag{74}$$

and go to (62).

3.3.1. Remarks

The above algorithm, as it stands, cannot be used if $\rho_d = \rho_L$—that is, if the particles are neutrally buoyant. However, it can easily be made to work by adding extra conditions to the space of the distributed Lagrange multipliers.

4. Numerical experiments

We now present the results of some numerical experiments. For all computations, the collocation method (37), (38) is used to enforce the constraint of rigid-body motion, and α and β are taken to be 1 and 0, respectively, in the Marchuk–Yanenko scheme (45)–(55).

4.1. Code validation: sedimentation of two circular particles in a two-dimensional channel

To validate the method developed in the preceding sections, we study the sedimentation of two circular particles in a two-dimensional channel, comparing the results with two different mesh sizes, and two different time steps.

The computational domain is a finite portion of a channel, which is moving along with the particles. Its x and y dimensions are 2 and 5, respectively. The diameter of the particles is $d = 0.25$. The fluid and particle densities are $\rho_L = 1.00$ and $\rho_d = 1.5$, respectively, and the fluid viscosity is $\eta = 0.01$. The range over which the repulsive force $\mathbf{F}'$ is active is $= 1.5\,h_{\mathbf{u}}$, and the two stiffness parameters in (19) and (20) are taken to be $\epsilon_p = 10^{-5}$ and $\epsilon_w = \epsilon_p/2$, respectively.[8] Both particles are initially on the channel centerline, with a distance of 0.5 between their centers. The initial fluid and particle velocities are

$$\begin{aligned} &\mathbf{u}_0 = \mathbf{0}, \\ &\mathbf{U}_{i,0} = \mathbf{0} \qquad i = 1,\ 2, \\ &\omega_{i,0} = 0 \qquad i = 1,\ 2. \end{aligned}$$

The simulation is run for two different time steps, $\Delta t = 0.0005$ and $\Delta t = 0.00025$. For the code validation studies, we use two different velocity mesh sizes, $h_{\mathbf{u}} = 1/192$ and $h_{\mathbf{u}} = 1/256$. (These values were chosen on the basis of boundary layer thickness and particle diameter.) In all cases, we take the pressure mesh size h_p to be $2\,h_{\mathbf{u}}$. Fig. 5 shows the characteristic *drafting*, *kissing*, and *tumbling* behavior of two sedimenting particles, computed with velocity mesh size $h_{\mathbf{u}} = 1/256$ and time step $\Delta t = 0.0005$.

In Figs. 6–8, the particle centers, translational velocities, and angular velocities are plotted against time, for time step $\Delta t = 0.0005$, and velocity mesh sizes $h_{\mathbf{u}} = 1/192$ and $h_{\mathbf{u}} = 1/256$. Note, in Fig. 6(b), that the time histories of the y-components of the particle centers, computed using the two different mesh sizes, are essentially indistinguishable. The particles are closest to

[8] The choice of ϵ_p and ϵ_w is not critical except in the final stage of a simulation of the sedimentation of a large number of particles in a closed box, when the particles accumulate at the bottom of box.

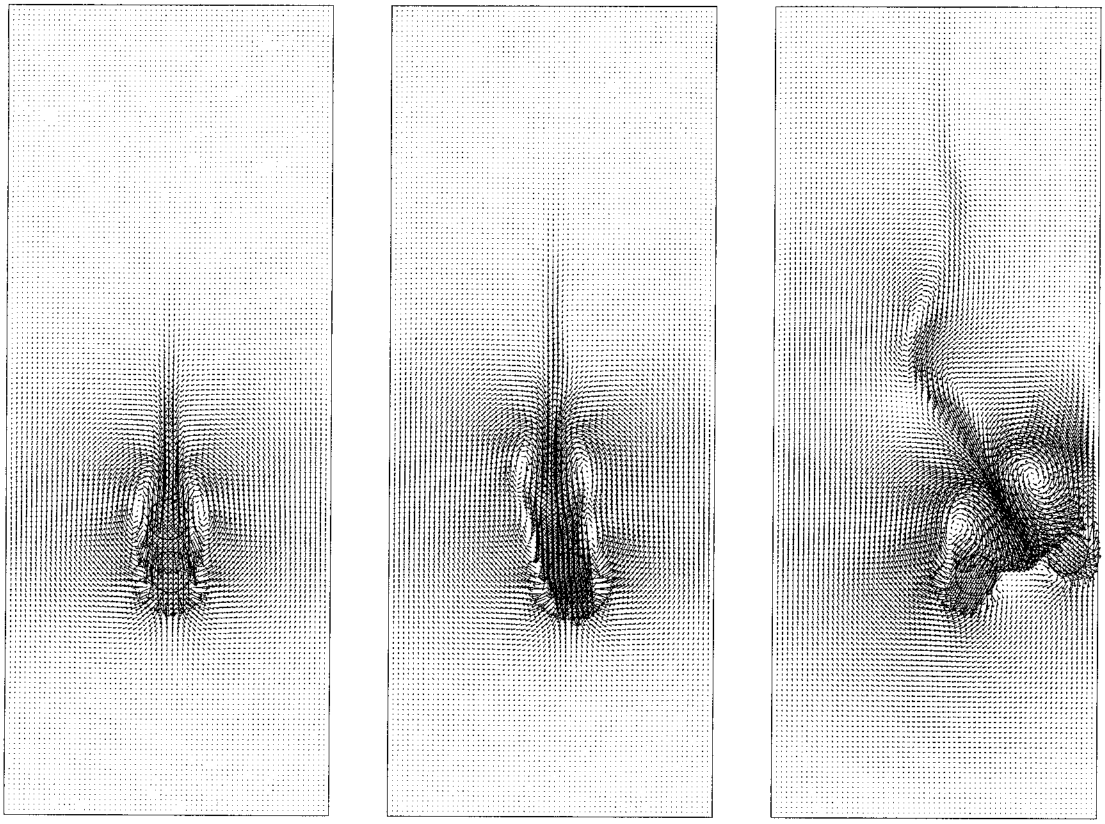

Fig. 5. Sedimentation of two circular particles: t = 0.15, 0.2, and 0.3 (left to right).

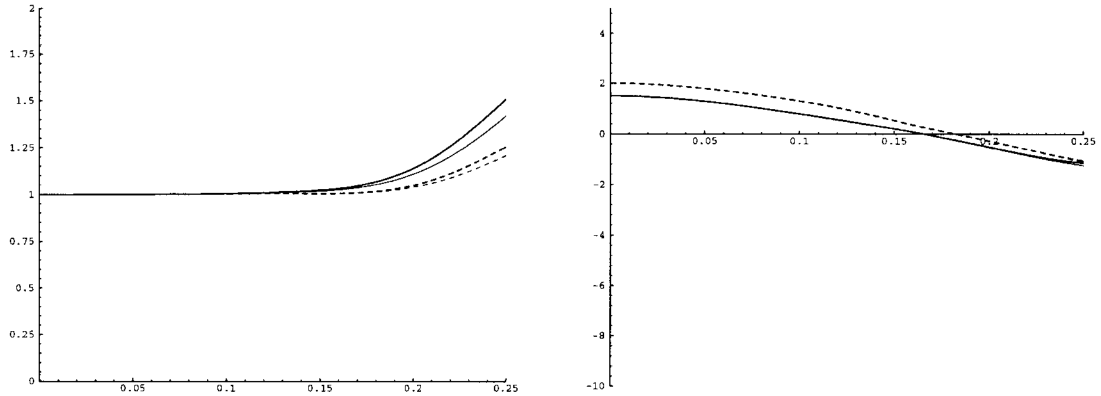

Fig. 6. Sedimentation of two circular particles: time histories of the x-coordinates (left) and y-coordinates (right) of the particle centers, obtained using velocity mesh sizes $h_{\mathbf{u}}$ = 1/192 (thick solid and dotted lines), and $h_{\mathbf{u}}$ = 1/256 (thin solid and dotted lines).

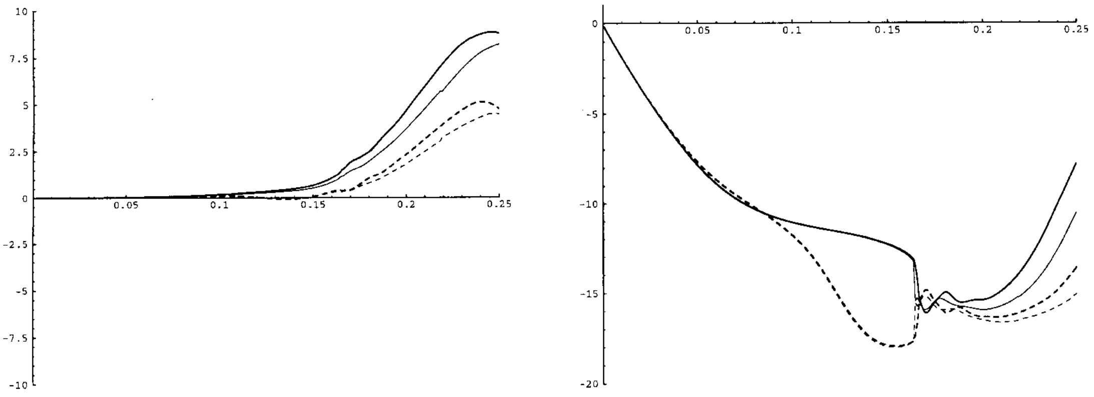

Fig. 7. Sedimentation of two circular particles: time histories of the x-components (left) and y-components (right) of the particle translational velocities, obtained using velocity mesh sizes $h_\mathbf{u} = 1/192$ (thick solid and dotted lines), and $h_\mathbf{u} = 1/256$ (thin solid and dotted lines).

each other ("kissing") at $t = 0.1665$. Up to this point, we have very good agreement between the two simulations. After "kissing," the particles "tumble"—a manifestation of the instability of a falling long body aligned with the stream (Fortes et al., 1987; Joseph et al., 1987; Joseph, 1996)—the particle motions computed with the two different velocity mesh sizes are still very close, considering the difficulty of the problem. Note that "tumbling" occurs at essentially the same time for both mesh sizes (see Figs. 7 and 8). The maximal particle Reynolds number is about 450.

To check that the computed solutions are converged with respect to the mesh size, another run was performed with velocity mesh size $h_\mathbf{u} = 1/384$ (and time step $\Delta t = 0.00025$). The results were essentially indistinguishable from those for $h_\mathbf{u} = 1/256$ (and time step $\Delta t = 0.00025$).

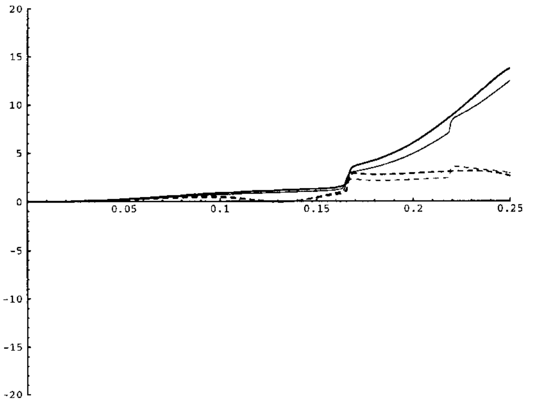

Fig. 8. Sedimentation of two circular particles: time histories of the particle angular velocities, obtained using velocity mesh sizes $h_\mathbf{u} = 1/192$ (thick solid and dotted lines), and $h_\mathbf{u} = 1/256$ (thin solid and dotted lines).

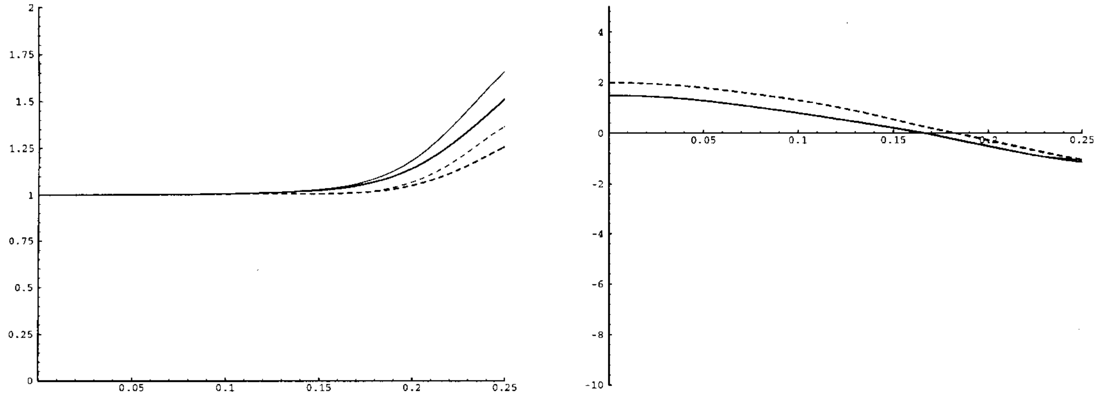

Fig. 9. Sedimentation of two circular particles: time histories of the x-coordinates (left) and y-coordinates (right) of the particle centers, obtained using time steps $\Delta t = 0.0005$ (thick solid and dotted lines), and $\Delta t = 0.00025$ (thin solid and dotted lines).

In Fig. 9–11, the same three quantities are plotted, this time for velocity mesh size $h_{\mathbf{u}} = 1/192$, and time steps, $\Delta t = 0.0005$ and $\Delta t = 0.00025$. Note, in Fig. 9(b), that the time histories of the y-components of the particle centers, computed using the two different time steps, are essentially indistinguishable. The particles are closest to each other ("kissing") at $t = 0.17125$. Again, we find excellent agreement between the two simulations prior to this time, and good agreement afterward. Note that "tumbling" occurs at essentially the same time for both time steps (see Figs. 10 and 11). The maximal particle Reynolds number is about 465 for $\Delta t = 0.00025$.

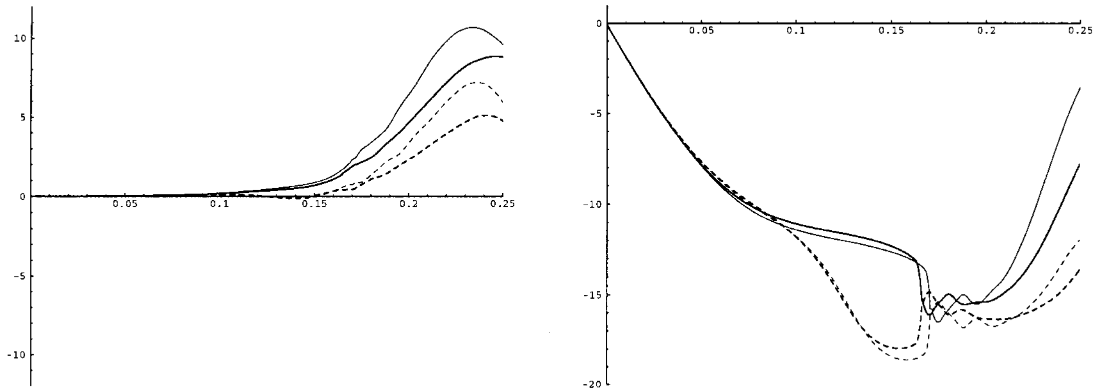

Fig. 10. Sedimentation of two circular particles: time histories of the x-components (left) and y-components (right) of the particle translational velocities, obtained using time steps $\Delta t = 0.0005$ (thick solid and dotted lines), and $\Delta t = 0.00025$ (thin solid and dotted lines).

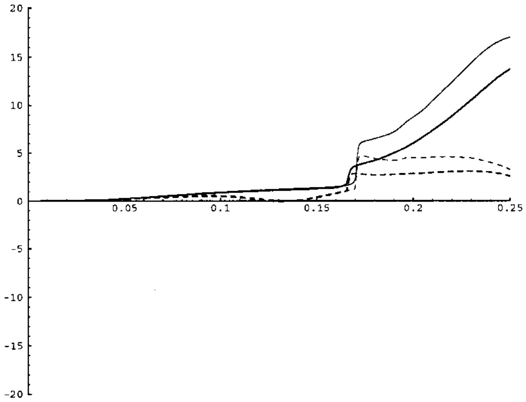

Fig. 11. Sedimentation of two circular particles: time histories of the particle angular velocities, obtained using time steps $\Delta t = 0.0005$ (thick solid and dotted lines), and $\Delta t = 0.00025$ (thin solid and dotted lines).

These results compare qualitatively well with those of Hu et al. (1992), which were obtained with different physical parameters and a different numerical methodology. They also compare well with the results of laboratory experiments with quasi two-dimensional flow.

4.2. Sedimentation of 504 circular particles in a closed two-dimensional box

The sedimentation of 504 circular particles in a closed two-dimensional box is shown in Fig. 12–20. The x and y dimensions of the box are both 2, and the diameter of the particles is

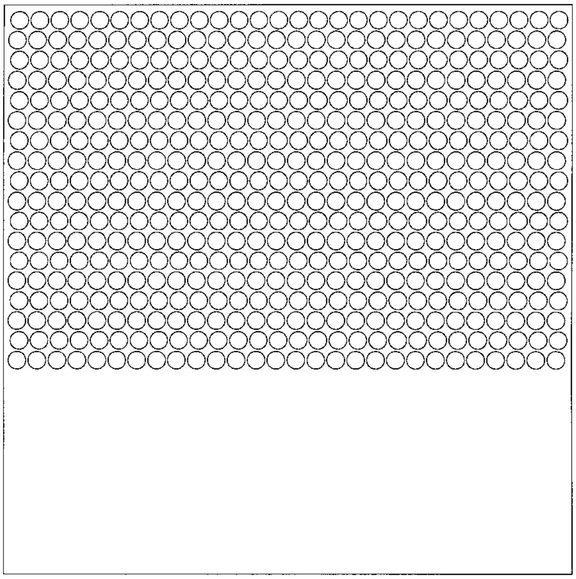

Fig. 12. Sedimentation of 504 circular particles: $t = 0$.

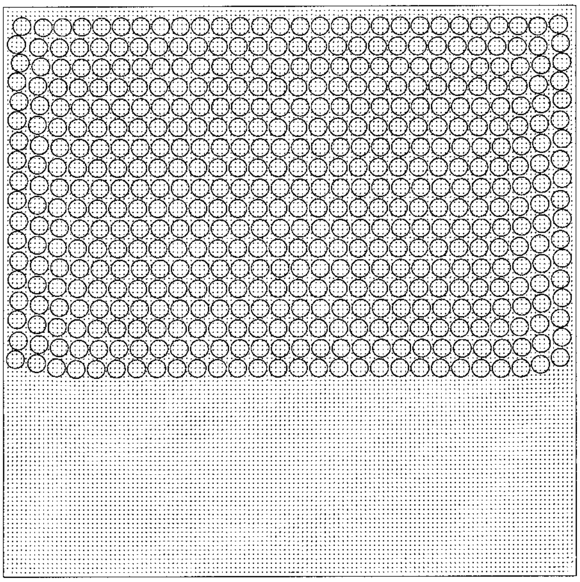

Fig. 13. Sedimentation of 504 circular particles: $t = 1$.

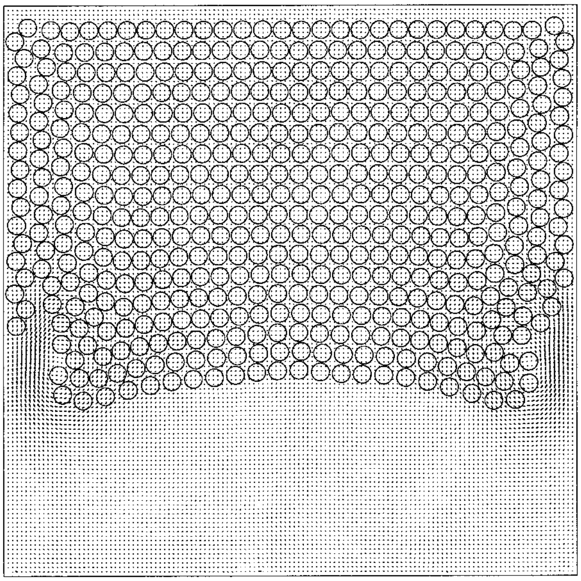

Fig. 14. Sedimentation of 504 circular particles: $t = 2$.

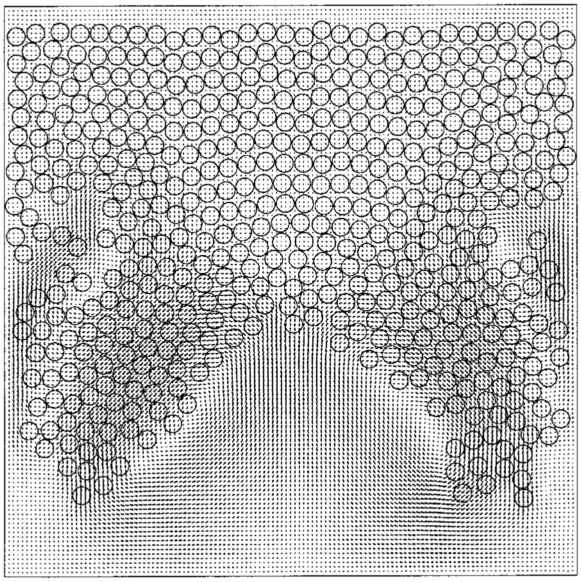

Fig. 15. Sedimentation of 504 circular particles: $t = 3$.

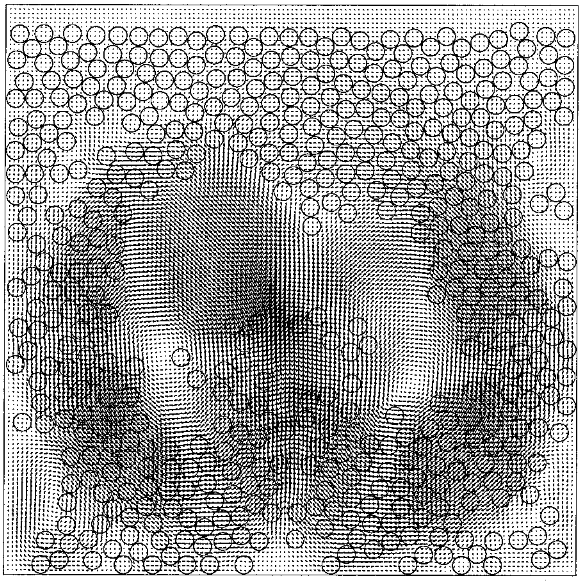

Fig. 16. Sedimentation of 504 circular particles: $t = 4$.

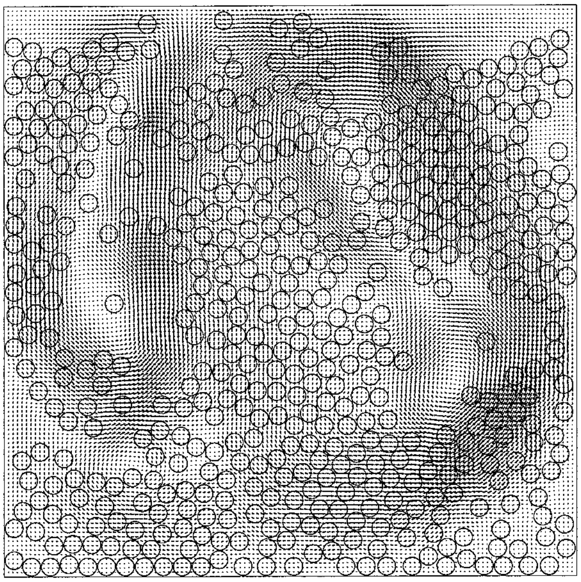

Fig. 17. Sedimentation of 504 circular particles: $t = 5$.

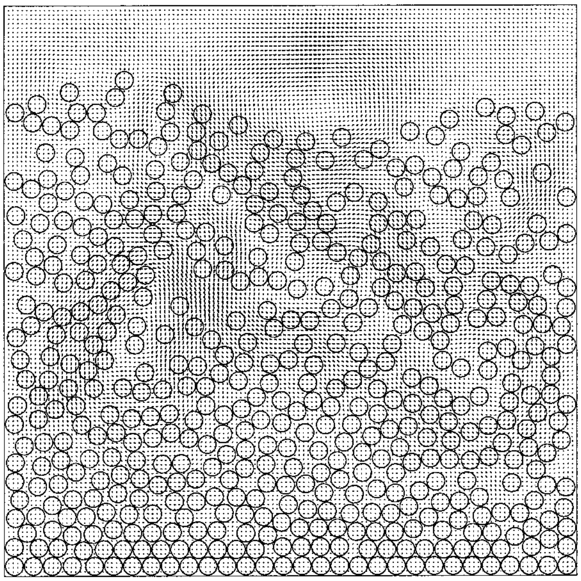

Fig. 18. Sedimentation of 504 circular particles: $t = 8$.

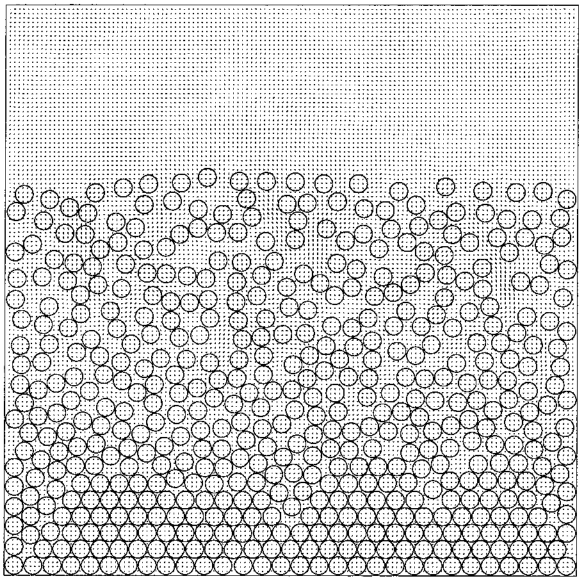

Fig. 19. Sedimentation of 504 circular particles: $t = 12$.

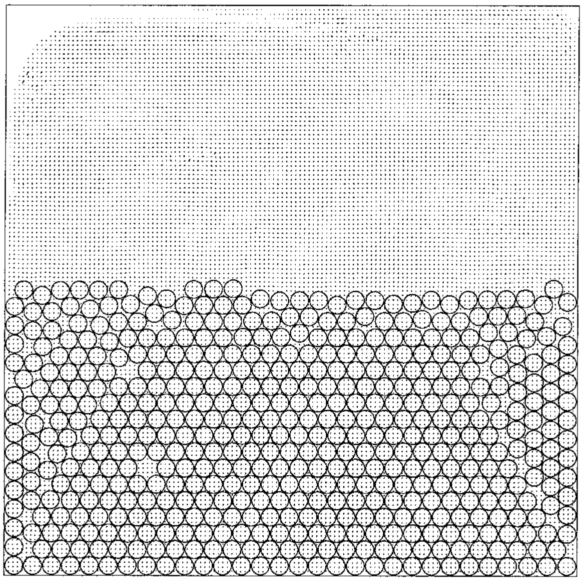

Fig. 20. Sedimentation of 504 circular particles: $t = 24$.

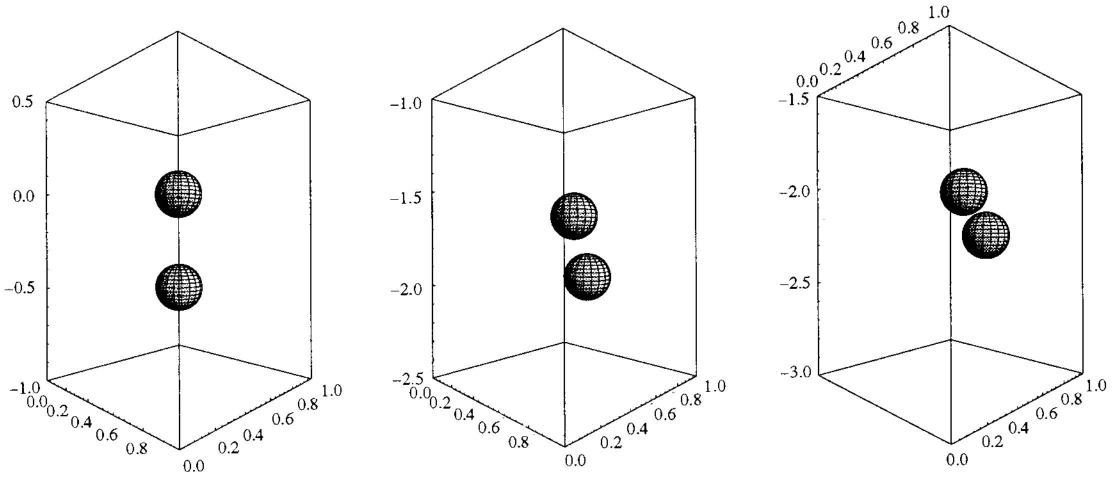

Fig. 21. Sedimentation of two spherical particles: t = 0.00, 0.35, and 0.40 (left to right).

$d = 0.0625$, giving a solids fraction of 38.66%. The fluid and particle densities are $\rho_L = 1.00$ and $\rho_d = 1.01$, respectively, and the fluid viscosity is $\eta = 0.01$. The range over which the repulsive force $\mathbf{F}'$ is active is $\rho = h_{\mathbf{u}}$, and the stiffness parameters in (19) and (20) are taken as $\epsilon_p = 8.15 \times 10^{-5}$ and $\epsilon_w = \epsilon_p/2$, respectively.

The initial fluid and particle velocities are

$$\begin{aligned} &\mathbf{u}_0 = \mathbf{0}, \\ &\mathbf{U}_{i,0} = \mathbf{0} \qquad i = 1, \ldots, 504, \\ &\omega_{i,0} = 0 \qquad i = 1, \ldots, 504. \end{aligned}$$

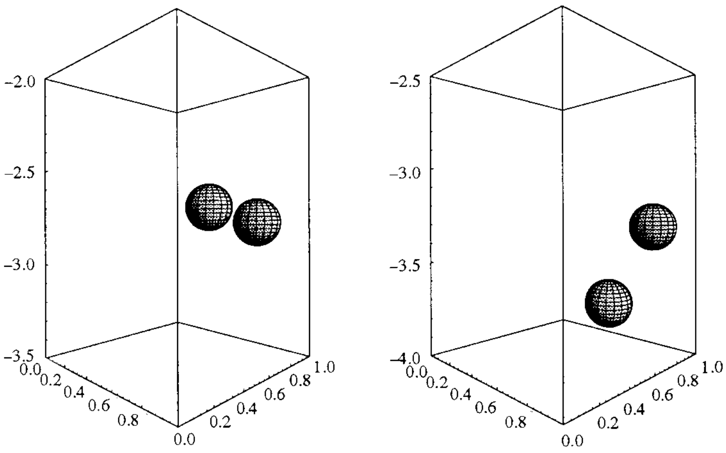

Fig. 22. Sedimentation of two spherical particles: t = 0.50 and 0.70 (left to right).

and the boundary condition is

$$\mathbf{u}_\Gamma(t) = \mathbf{0} \quad t \geq 0.$$

The time step, velocity mesh size, and pressure mesh size are $\Delta t = 0.001$, $h_\mathbf{u} = 1/256$ (263169 nodes), and $h_\mathrm{p} = 1/128$ (66049 nodes), respectively. (A fine mesh is required because of the large number of particles.)

The numbers of iterations for the divergence-free projection (45), (46), the linearized advection–diffusion problem (47), and the rigid-body-motion projection (52), (53) are 12–14, 5, and about 7, respectively. These numbers are nearly independent of the mesh size and of the number of particles.

A Rayleigh–Taylor instability develops in the advancing front between $t = 1$ and $t = 3$. By $t = 2$, an eddy has formed close to each wall, which "pulls" particles downward. Between $t = 3$ and $t = 5$, a pair of stronger eddies forms in the lower center of the box, which push particles almost to the top of the box. At the end, all particles have settled to the bottom of the box. The maximal particle Reynolds number in the entire simulation is $\rho_\mathrm{L}\mathbf{U}_i d/\eta = 14.56$.

4.3. Sedimentation of two spherical particles in a square three-dimensional channel

Feng et al. (1995) computed the force and torque on an ellipsoidal particle falling slowly in a square, three-dimensional channel filled with a viscoelastic liquid. The problem solved was for a small perturbation of Stokes flow with inertia and viscoelastic normal stresses. The problem was solved using a fictitious domain method—but with an explicit formulation in which the forces and torques were needed and calculated.

Here we carry out a comparable direct numerical simulation using the combined weak formulation described in this article (with forces and torques eliminated): two spherical particles sedimenting in a square three-dimensional channel filled with a Newtonian fluid. (See Figs. 21 and 22.)

The computational domain is a finite portion of a square three-dimensional channel, which follows (approximately) the center of mass of the particles. Its x, y, and z dimensions are 1, 1, and 2.5, respectively. The diameter of the particles is $d = 0.25$. The fluid and particle densities are $\rho_\mathrm{L} = 1.00$ and $\rho_\mathrm{d} = 1.14$, respectively, and the fluid viscosity is $\eta = 0.01$. The initial fluid and particle velocities are

$$\begin{aligned} &\mathbf{u}_0 = \mathbf{0}, \\ &\mathbf{U}_{i,0} = \mathbf{0} \qquad i = 1,\ 2, \\ &\boldsymbol{\omega}_{i,0} = \mathbf{0} \qquad i = 1,\ 2. \end{aligned}$$

The time step, velocity mesh size, and pressure mesh size are $\Delta t = 0.001$, $h_\mathbf{u} = 1/40$, and $h_\mathrm{p} = 1/20$, respectively.

The simulation reproduces the familiar dynamics of *drafting*, *kissing*, and *tumbling*, as seen in Figs. 21 and 22. (The computation was performed in a moving frame of reference, so the particles appear to move upward during some of the time steps.) The maximal particle Reynolds numbers for the two particles are 168.89 and 198.86. The distance of closest approach (not shown) is 0.363 $h_\mathbf{u}$.

5. Conclusions

We have presented a new Lagrange-multiplier based fictitious domain method for the direct numerical simulation of the flow of fluid–particle mixtures. The method has the following advantages:

1. The hydrodynamic forces and torques do not need to be computed explicitly;
2. The computational domain is time-independent, so a fixed mesh can be used, eliminating the need for repeated remeshing and projection;
3. The computational domain is geometrically simple, so regular grids and fast solvers can be used.

Acknowledgements

We acknowledge the helpful comments and suggestions of E. J. Dean, V. Girault, J. He, Y. Kuznetsov, B. Maury, and G. Rodin and also the support of NEC concerning the use of an SX-3 supercomputer.

We acknowledge the support of the NSF under HPCC Grand Challenge grant ECS-9527123, NSF (grants DMS 8822522, DMS 9112847, DMS 9217374), University of Minnesota Supercomputing Institute, DRET (grant 89424), DARPA (contracts AFOSR F49620-89-C-0125 and AFOSR-90-0334), the Texas Board of Higher Education (grants 003652156ARP and 003652146ATP) and the University of Houston (PEER grant 1-27682).

T. Hesla's work was supported in part by an ARO-AASERT grant.

References

Babuska, I., 1973. The finite element method with lagrange multipliers. Numer. Math. 20, 179–192.

Bertrand, F., Tanguy, P.A., Thibault, F., 1997. A three-dimensional fictitious domain method for incompressible fluid flow problems. Int. J. Num. Meth. Fluids 25, 719–736.

Brezzi, F., 1974. On the existence, uniqueness and approximation of saddle-point problems arising from Lagrange multipliers. RAIRO Anal. Numer. R2, 129–151.

Bristeau, M.O., Glowinski, R., Périaux, J., 1987. Numerical methods for the Navier–Stokes equations. Comp. Phys. Rep. 6, 73–187.

Buzbee, B.L., Dorr, F.W., George, J.A., Golub, G.H., 1971. The direst solution of the discrete Poisson equation on irregular regions. SIAM J. Num. Anal. 8, 722–736.

Chorin, A.J., 1967. A numerical method for solving incompressible viscous flow problems. J. Comp. Phys. 2, 12–26.

Chorin, A.J., 1968. On the convergence and approximation of discrete approximation to the Navier–Stokes equations. Math. Comp. 23, 341–353.

Chorin, A.J., 1973. Numerical study of slightly viscous flow. J. Fluid Mech. 57, 785–796.

Dean, E., Glowinski, R., 1997. A wave equation approach to the numerical solution of the Navier–Stokes equations for incompressible viscous flow. C. R. Acad. Sci. Paris T 325 Série 1, 783–791.

Esmaeeli, A., Tryggvason, G. 1998. Direct numerical simulations of bubbly flows. Part 1: low Reynolds number arrays. J. Fluid Mech. in press.

Feng, J., Joseph, D.D., Glowinski, R., Pan, T-W., 1995. A three-dimensional computation of the force and moment on an ellipsoid settling slowly through a viscoelastic fluid. J. Fluid Mech. 283, 1–16.

Feng, J., Huang, P.Y., Joseph, D.D., 1996. Dynamic simulation of the sedimentation of solid particles in an Oldroyd B fluid. J. Non-Newt. Fluid mech. 63, 63–88.

Fortes, A., Joseph, D.D., Lundgren, T.S., 1987. Nonlinear mechanism of fluidization of beds of spherical particles. J. Fluid Mech. 177, 467–483.

Glowinski, R., 1984. Numerical Methods for Nonlinear Variational Problems. Springer, New York.

Glowinski, R., Pironneau, O., 1992. Finite element methods for Navier–Stokes equations. Ann. Rev. Fluid Mech. 24, 167–204.

Glowinski, R., Pan, T-W., Périaux, J., 1994a. A fictitious domain method for Dirichlet problems and applications. Comp. Meth. Appl. Mech. Engng. 111, 283–303.

Glowinski, R., Pan, T-W., Périaux, J., 1994b. A fictitious domain method for external incompressible viscous flow modeled by Navier–Stokes equations. Comp. Meth. Appl. Mech. Engng. 112, 133–148.

Glowinski, R., Pan, T-W., Périaux, J., 1995. A Lagrange multiplier/fictitious domain method for the Dirichlet problem. Generalization to some flow problems. Japan J. Ind. Appl. Math. 12, 87–108.

Glowinski, R., Hesla, T., Joseph, D.D., Pan, T-W., Périaux, J., 1997a. Distributed Lagrange multiplier methods for particulate flows. In: M-O. Bristeau, G. Etgen, W. Fitzgibbon, J.L. Lions, J. Périaux, M.F. Wheeler (Ed.). Computational Science for the Twenty-First Century. Wiley, Chichester, pp. 270–279.

Glowinski, R., Pan, T-W., Périaux, J., 1997b. Fictitious domain methods for incompressible viscous flow around moving rigid bodies. In: J.R. Whiteman (Ed.). The Mathematics of Finite Elements and Applications, Highlight 1996. Wiley, Chichester, pp. 155–174.

Glowinski, R., Pan, T-W., Périaux, J., 1997c. A Lagrange multiplier/fictitious domain method for the numerical simulation of incompressible viscous flow around moving rigid bodies (I): the case where the rigid body motions are known a priori. C. R. Acad. Sci. Paris 324, 361–369.

Glowinski, R., Pan, T-W., Périaux, J. 1998. Distributed Lagrange multiplier methods for incompressible viscous flow around moving rigid bodies. Comp. Meth. Appl. Mech. Engng. 151, 181–194.

Hesla, T.I., 1991. The dynamical simulation of two-dimensional fluid/particle systems. Unpublished notes.

Hu, H.H., 1996. Direct simulation of flows of solid–liquid mixtures. Int. J. Multiphase Flow 22, 335–352.

Hu, H.H., Joseph, D.D., Crochet, M.J., 1992. Direct simulation of fluid particle motions. Theor. Comp. Fluid Dyn. 3, 285–306.

Huang, P.Y., Feng, J., Hu, H.H., Joseph, D.D., 1997. Direct simulation of the motion of solid particles in Couette and Poiseuille flows of viscoelastic fluids. J. Fluid Mech. 343, 73–94.

Hyman, M.A., 1952. Non-iterative numerical solution of boundary-value problems. Appl. Sci. Res., Sec. B 2, 325–351.

Johnson, A., Tezduyar, T., 1996. 3d simulation of fluid–particle interactions with the number of particles reaching 100, Research report 96-037. Army High Performance Computing Research Center, University of Minnesota.

Johnson, A., Tezduyar, T., 1997. Fluid–particle simulations reaching 100 particles, Research report 97-010. Army High Performance Computing Research Center, University of Minnesota.

Joseph, D.D., 1996. Flow induced microstructure in Newtonian and viscoelastic fluids. Proceedings of the Fifth World Congress of Chemical Engineering, Particle Technology Track, vol. 6. American Institute of Chemical Engineers, Keynote presentation (paper no. 95a, Second Particle Technology Forum). San Diego, California.

Joseph, D.D., Fortes, A., Lundgren, T.S., Singh, P., 1987. Nonlinear mechanics of fluidization of beds of spheres, cylinders and disks in water. In: G. Papanicolau (Ed.). Advances in Multiphase Flow and Related Problems. SIAM, pp. 101–122.

LeVeque, R.J., Li, Z., 1994. The immersed interface method for elliptic equations with discontinuous coefficients and singular sources. SIAM J. Num. Anal. 31, 1019–1044.

LeVeque, R.J., Li, Z., 1997. Immersed interface methods for Stokes flow with elastic boundaries or surface tension. SIAM J. Sci. Stat. Comp. 18, 709–735.

LeVeque, R.J., Oliger, J., 1983. Numerical methods based on additive splitting for hyperbolic partial differential equations. Math. Comp. 40, 469–497.

Marchuk, G.I., 1990. Splitting and alternate direction methods. In: P.G. Ciarlet, J.L. Lions (Ed.). Handbook of Numerical Analysis, vol. 1. North-Holland, Amsterdam, pp. 197–462.

Maury, B. 1997. A many-body lubrication model. C. R. Acad. Sci. Paris. T. 325, Série 1, 1053–1058.

Patankar, N.A., 1997. Numerical simulation of particulate two-phase flow. Ph.D. thesis. University of Pennsylvania, Pennsylvania.

Peskin, C.S., 1977. Numerical analysis of blood flow in the heart. J. Comp. Phys. 25, 220–252.

Peskin, C.S., 1981. Lectures on mathematical aspects of physiology. In: F.C. Hoppensteadt (Ed.). Lectures in Applied Mathematics, vol. 19. American Mathematical Society, Providence, Rhode Island, pp. 69–107.

Peskin, C.S., McQueen, D.M., 1980. Modeling prosthetic heart valves for numerical analysis of blood flow in the heart. J. Comp. Phys. 37, 113–132.

Pironneau, O., 1989. Finite Element Methods for Fluids. Wiley, Chichester.

Saul'ev, V.K., 1963. On solving boundary-value problems on high-performance computers by fictitious-domain methods. Sib. Math. J. 4, 912–925.

Turek, S., 1996. A comparative study of time-stepping techniques for the incompressible Navier–Stokes equations: from fully implicit non-linear schemes to semi-implicit projection methods. Int. J. Num. Meth. Fluids 22, 987–1011.

Unverdi, S.O., Tryggvason, G., 1992a. Computations of multi-fluid flows. Physica D 60, 70–83.

Unverdi, S.O., Tryggvason, G., 1992b. A front-tracking method for viscous, incompressible, multi-fluid flows. J. Comp. Phys. 100, 25–37.