A second-order accurate Immersed Boundary Method for fully resolved simulations of particle-laden flows

Wim-Paul Breugem*

Delft University of Technology, Laboratory for Aero and Hydrodynamics, Leeghwaterstraat 21, NL-2628 CA Delft, The Netherlands

Abstract

An Immersed Boundary Method (IBM) with second-order spatial accuracy is presented for fully resolved simulations of incompressible viscous flows laden with rigid particles. The method is based on the computationally efficient direct-forcing method of Uhlmann [J. Comput. Phys. 209 (2005) 448] that is embedded in a finite-volume/pressure-correction method. The IBM consists of two grids: a fixed uniform Eulerian grid for the fluid phase and a uniform Lagrangian grid attached to and moving with the particles. A regularized delta function is used to communicate between the two grids and proved to be effective in suppressing grid locking. Without significant loss of efficiency, the original method is improved by: 1) a better approximation of the no-slip/no-penetration (ns/np) condition on the surface of the particles by a multidirect forcing scheme, 2) a correction for the excess in the effective particle diameter by a slight retraction of the Lagrangian grid from the surface towards the interior of the particles with a fraction of the Eulerian grid spacing, and 3) an enhancement of the numerical stability for particle-fluid mass density ratios near unity by a direct account of the inertia of the fluid contained within the particles. The new IBM contains two new parameters: the number of iterations N_s of the multidirect forcing scheme and the retraction distance r_d. The effect of N_s and r_d on the accuracy is demonstrated for five different flows. The results show that N_s has a strong influence on the error in the ns/np condition and little influence on the effective particle diameter, while the opposite holds for r_d. Furthermore, r_d has a strong influence on the order

*Tel.: +31 15 278 8663; fax: +31 15 278 2947.
Email address: w.p.breugem@tudelft.nl (Wim-Paul Breugem)

of grid convergence. It is found that for spheres the choice of $r_d = 0.3\Delta x$ yields second-order accuracy compared to first-order accuracy of the original method that corresponds to $r_d = 0$. Finally, $N_s = 2$ appears optimal for reducing the error in the ns/np condition and maintaining the computational efficiency of the method.

Keywords: immersed boundary method, particle-laden flow, finite-volume method, grid locking, multidirect forcing scheme, retraction distance, order of grid convergence

---

1. Introduction

The Immersed Boundary Method (IBM) [21] has become a popular method for fully resolved simulations of particle-laden flows and flows through complex geometries more in general [19]. Characteristic for this method is that the computational grid for the fluid phase does not conform to the shape of the particles like in conventional methods with a body-fitted grid. Furthermore, the grid is typically structured, often Cartesian, fully continuous in space and fixed in time. The no-slip/no-penetration (ns/np) condition on the surface of a particle is not imposed explicitly, but instead additional forcing is applied to the flow in the immediate vicinity of the surface such that this condition is satisfied by good approximation. The advantage of the IBM over a method with a body-fitted grid is its computational efficiency: it does not require regridding when particles are moving and a simple structured grid enables the use of efficient computational methods for solving the Navier-Stokes equations. The price that has to be paid is a loss of accuracy because of the error in the approximation of the ns/np condition. The challenge is to develop an IBM that is both computationally efficient and sufficiently accurate.

Uhlmann [31] developed a computationally efficient IBM for particle-laden flows that is embedded in a finite-volume/pressure-correction method [34]. It makes use of two different grids: a fixed, uniform and continuous Cartesian grid for the fluid phase and a uniform grid attached to and moving with the surface of the particles. The present author will refer to the two grids as the Eulerian and the Lagrangian grid, respectively. The method solves the Navier-Stokes equations for the fluid phase and the Newton-Euler equations for the particles. The flow-induced force and torque acting on a

particle is obtained from the IBM force distribution on the Lagrangian grid. The IBM force distribution on the Lagrangian grid is computed from the requirement that on the surface of the particle the prediction velocity of the pressure-correction scheme is equal to the local particle velocity. Since the grid points of the two grids do not overlap in general, interpolation is required of the prediction velocity from the Eulerian to the Lagrangian grid. Furthermore, spreading is required of the computed IBM force from the Lagrangian back to the Eulerian grid. In Uhlmann's IBM the interpolation and spreading operations [22] are based on the regularized Dirac delta function of Roma et al. [24] with a width of three Eulerian grid cells. Consequently, the particles have a smooth (i.e. non-sharp) interface from the point of view of the fluid phase.

The smoothing of the particle interface has the important advantage of suppressing undesired high-frequency oscillations in the force and torque acting on a particle when it moves over the Eulerian grid. These oscillations originate from variations in the interpolated prediction velocity when the Lagrangian grid moves with the particle over the Eulerian grid and thus changes its orientation relative to the Eulerian grid. In other words, they are present because the interpolation operation is not translation invariant [22]. This phenomenon is dubbed as *grid locking* by the present author, since the wavelength of the oscillations is set by the dimensions of the Eulerian grid cells and their period by the time it takes for a particle to travel from one Eulerian grid cell to another. Uhlmann [31] showed for the case of a forced oscillation of a cylinder in uniform cross-flow that the amplitude of the spurious oscillations decreases when the width of the regularized Dirac delta function is increased. The regularized Dirac delta function of Roma et al. [24] with a width of three Eulerian grid cells is considered as effective for suppressing grid locking and its compact support as computationally efficient [31].

The smoothing of the particle interface has, however, also an important disadvantage. The interpolation of the prediction velocity based on the regularized Dirac delta function of Roma et al. [24] is formally second-order accurate in space, but only when applied to a *smooth* velocity field. Peskin [22] pointed out that the velocity field near a solid boundary is *non-smooth* as it contains a jump in its normal derivative over the boundary. Consequently, the interpolation of the prediction velocity becomes first-order accurate and this explains the first-order spatial accuracy of Uhlmann's IBM for the flows studied in this paper. A possible way to improve the accuracy is to resort to a sharp representation of the interface. However, this will probably am-

plify grid locking [31], which is undesired. The challenge is to keep a smooth representation of the interface in order to suppress grid locking and to find other ways for improving the accuracy of the IBM.

In this paper a new IBM is presented that is based on the IBM of Uhlmann [31]. The objective of this study was to increase the numerical accuracy of the method and to improve its numerical stability for particle-fluid mass density ratios near unity, while maintaining its computational efficiency and the suppression of grid locking. Firstly, the approximation of the ns/np condition is improved by the multidirect forcing scheme of Luo et al. [17]. Secondly, a correction is implemented for the excess in the effective particle diameter by a slight retraction of the Lagrangian grid from the surface towards the interior of a particle [11]. The excess in the effective particle diameter is a direct consequence of the smoothing of the particle interface, which causes that the effective particle diameter is larger than the actual particle diameter from the point of view of the fluid phase. Thirdly, the numerical stability of the method is enhanced for particle-fluid mass density ratios near unity by a direct account of the inertia of the fluid contained within the particles [15].

The new IBM contains two new control parameters: the number of iterations N_s of the multidirect forcing scheme and the retraction distance r_d. The effect of N_s and r_d on the numerical accuracy has been investigated in detail for two different particle-laden flows. The main novelty of the present paper is the demonstration that for an appropriate choice of r_d the original IBM becomes second-order accurate in space, while the smooth representation of the particle interface still suppresses grid locking and the computational efficiency of the method is nearly unchanged.

This paper is organized as follows. The governing equations for particle-laden flows are given in section 2. This is followed by a description of the IBM of Uhlmann [31] in section 3. Next, the three main improvements to this method are presented in section 4. The details of the numerical method are given in section 5. The computational results are presented in section 6. Finally, the conclusions and a discussion are given in section 7.

2. Governing equations

Particle-laden flows are described by the Navier-Stokes equations for the fluid phase and the Newton-Euler equations for the solid particles.

The Navier-Stokes equations for an incompressible Newtonian flow read:

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

$$\rho_f \left(\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot \mathbf{uu} \right) = -\nabla p_e - \nabla p + \mu_f \nabla^2 \mathbf{u}, \tag{1b}$$

where $\mathbf{u}$ is the velocity, p_e is the contribution to the total pressure from a *constant* pressure gradient that is possibly imposed to drive a flow, p is the modified pressure (i.e., the total pressure minus p_e and the contribution from the hydrostatic pressure), ρ_f is the mass density and μ_f is the dynamic viscosity of the fluid.

The velocity $\mathbf{U}_p$ of an infinitesimal particle segment at position $\mathbf{X}$ can be decomposed into a translational part and a rotational part according to:

$$\mathbf{U}_p = \mathbf{u}_c + \omega_c \times \mathbf{r}, \tag{2}$$

where $\mathbf{r} = \mathbf{X} - \mathbf{x}_c$ is the position vector relative to the particle centroid at $\mathbf{X} = \mathbf{x}_c$, $\mathbf{u}_c$ is the translational velocity of the particle centroid and ω_c is the angular velocity of the particle. In this paper only results are shown for solid spheres. The translational and angular velocities of a particle are described by the Newton-Euler equations, which for a sphere reduce to:

$$\rho_p V_p \frac{d\mathbf{u}_c}{dt} = \oint_{\partial V} \tau \cdot \mathbf{n} dA + (\rho_p - \rho_f) V_p \mathbf{g} - V_p \nabla p_e + \mathbf{F}_c, \tag{3a}$$

$$I_p \frac{d\omega_c}{dt} = \oint_{\partial V} \mathbf{r} \times (\tau \cdot \mathbf{n}) \, dA + \mathbf{T}_c. \tag{3b}$$

Here ρ_p is the mass density of the particle, V_p is the volume of the particle and equal to $(4/3)\pi R^3$ for a sphere with radius R, $\tau = -p\mathbf{I} + \mu_f \left(\nabla \mathbf{u} + \nabla \mathbf{u}^T \right)$ is the stress tensor for a Newtonian fluid with $\mathbf{I}$ the unit tensor, $\mathbf{n}$ is the outward-pointing unit normal at the surface ∂V of the particle, $\mathbf{g}$ is the gravitational acceleration and I_p is the moment of inertia of the particle and equal to $(2/5)\rho_p V_p R^2$ for a solid sphere. $\mathbf{F}_c$ and $\mathbf{T}_c$ represent, respectively, the force and torque acting on the particle as a result of collisions/physical contact with other particles or solid walls. In Eq. (3a) the terms $-\rho_f V_p \mathbf{g}$ and $-V_p \nabla p_e$ account for the forces from the linear stratification in the hydrostatic pressure and p_e, respectively.

Eqs. (1a)–(1b) and (3a)–(3b) form a system of differential equations, which are coupled through the ns/np condition on the surface of the solid

particles:

$$\mathbf{u} \;=\; \mathbf{U}_p\left(\mathbf{X}\right) \quad \forall \quad \mathbf{X} \in \partial V. \tag{4}$$

As mentioned in the introduction section, in an IBM the above condition is not imposed directly, but instead in the immediate vicinity of a solid particle a body force $\mathbf{f}$ is added to the right-hand side of Eq. (1b) such that it is satisfied by good approximation:

$$\rho_f \left(\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot \mathbf{uu} \right) \;=\; -\nabla p_e - \nabla p + \mu_f \nabla^2 \mathbf{u} + \rho_f \mathbf{f}. \tag{5}$$

Eq. (5) is solved in the entire domain, including the space occupied by the particles.

3. Original immersed boundary method

In this section the original IBM developed by Uhlmann [31] is explained in brief. The IBM makes use of two different computational grids as illustrated in Fig. 1. The first grid is a fixed and fully-staggered Cartesian grid [9] for

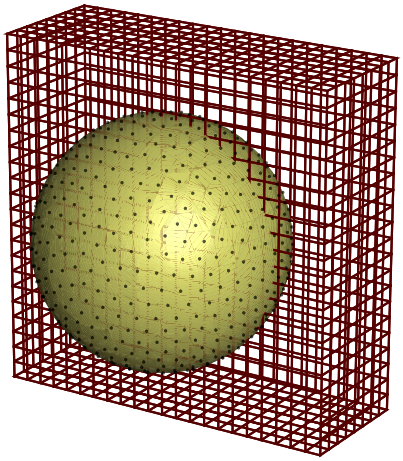

Figure 1: Eulerian grid and the distribution of the Lagrangian grid points over a sphere for $D/\Delta x = 16$, where D is the sphere diameter, and retraction distance $r_d = 0.3\Delta x$. The number of Lagrangian grid points on the retracted surface of the sphere is equal to 746.

the fluid phase and is referred to as the Eulerian grid. The second grid is a

fixed grid that is attached to and moves with the surface of the solid particles and is referred to as the Lagrangian grid.

Uhlmann's IBM is a so-called direct (or discrete) forcing method [4, 19] in which the additional forcing on the fluid in the vicinity of solid boundaries is introduced after discretization of the Navier-Stokes equations. The direct forcing method can be easily embedded in a pressure-correction scheme. In semi-discrete form and based on the Crank-Nicolson scheme [34] for the integration in time for simplicity at this moment, this scheme reads:

$$\mathbf{u}^* = \mathbf{u}^n + \frac{\Delta t}{\rho_f}\left(-\nabla p^{n-1/2} + \mathbf{rhs}^{n+1/2}\right), \tag{6a}$$

$$\mathbf{u}^{**} = \mathbf{u}^* + \Delta t \mathbf{f}^{n+1/2}, \tag{6b}$$

$$\nabla^2 \tilde{p} = \frac{\rho_f}{\Delta t}\nabla \cdot \mathbf{u}^{**}, \tag{6c}$$

$$\mathbf{u}^{n+1} = \mathbf{u}^{**} - \frac{\Delta t}{\rho_f}\nabla \tilde{p}, \tag{6d}$$

$$p^{n+1/2} = p^{n-1/2} + \tilde{p}, \tag{6e}$$

where $\mathbf{rhs} \equiv -\nabla p_e - \rho_f \nabla \cdot \mathbf{uu} + \mu_f \nabla^2 \mathbf{u}$ is defined here for convenience, $\mathbf{u}^*$ is the first prediction velocity, $\mathbf{u}^{**}$ is the second prediction velocity that includes the additional forcing from the IBM, $\tilde{p}$ is the correction pressure and Δt is the computational time step.

Note that in Eq. (6b) the forcing from the IBM is applied to the *second prediction* velocity instead of the *actual* fluid velocity. For the accuracy of the IBM it is therefore important to include Eq. (6e) in the pressure-correction scheme. From a Taylor expansion of Eq. (6e) it is expected that $\tilde{p} = O(\Delta t)\left(\partial p/\partial t\right)^{n+1/2}$. From Eq. (6d) it then follows that for instationary flows $\mathbf{u}^{n+1} \approx \mathbf{u}^{**} + O(\Delta t^2)$, while $\mathbf{u}^{n+1} \approx \mathbf{u}^{**} + O(\Delta t)$ when Eq. (6e) would be omitted from the scheme and $p^{n-1/2}$ in Eq. (6a) replaced by $\tilde{p}$. For stationary flows $\partial p/\partial t = 0$ and hence $\mathbf{u}^{n+1} = \mathbf{u}^{**}$.

The IBM force $\mathbf{f}$ in Eq. (6b) is computed by:

1. *interpolation* of the first prediction velocity $\mathbf{u}^*$ from the Eulerian to the Lagrangian grid,
2. *computation* of the IBM force on the Lagrangian grid based on the difference between the interpolated first prediction velocity and the particle velocity,
3. *spreading* of this force from the Lagrangian to the Eulerian grid.

The interpolation and spreading operations [22] are based on a regularized Dirac delta function δ_d. Uhlmann [31] took the regularized Dirac delta function of Roma et al. [24], which extends over three grid cells in all coordinate directions. In mathematical notation the scheme for computing the IBM force can be summarized as follows:

$$\mathbf{U}_l^* = \sum_{ijk} \mathbf{u}_{ijk}^* \, \delta_d \left(\mathbf{x}_{ijk} - \mathbf{X}_l^n \right) \Delta x \Delta y \Delta z, \tag{7a}$$

$$\mathbf{F}_l^{n+1/2} = \frac{\mathbf{U}_p \left(\mathbf{X}_l^n \right) - \mathbf{U}_l^*}{\Delta t}, \tag{7b}$$

$$\mathbf{f}_{ijk}^{n+1/2} = \sum_{l} \mathbf{F}_l^{n+1/2} \, \delta_d \left(\mathbf{x}_{ijk} - \mathbf{X}_l^n \right) \Delta V_l. \tag{7c}$$

Here the upper case letters refer to quantities defined on the Lagrangian grid, while the lower case letters denote quantities defined on the Eulerian grid. $\mathbf{x}_{ijk}$ denotes the position of the Eulerian grid point with index (i, j, k). $\mathbf{X}_l$ denotes the position of the Lagrangian grid point with index l. Δx, Δy and Δz are the dimensions of the Eulerian grid cells and ΔV_l is the volume of the Lagrangian grid cells.

From Eq. (7b) the force distribution on the Lagrangian grid is computed. In Appendix A it is shown how this can be related to the total flow-induced force and torque acting on a particle. Substituting these expressions into Eqs. (3a)–(3b) yields:

$$\begin{aligned} \rho_p V_p \frac{d\mathbf{u}_c}{dt} \approx & -\rho_f \sum_{l=1}^{N_L} \mathbf{F}_l^{n+1/2} \Delta V_l + \rho_f \frac{d}{dt} \left(\int_{V_p} \mathbf{u} dV \right) \\ & + \left(\rho_p - \rho_f \right) V_p \mathbf{g} + \mathbf{F}_c^{n+1/2}, \end{aligned} \tag{8a}$$

$$\begin{aligned} I_p \frac{d\omega_c}{dt} \approx & -\rho_f \sum_{l=1}^{N_L} \mathbf{r}_l^n \times \mathbf{F}_l^{n+1/2} \Delta V_l + \rho_f \frac{d}{dt} \left(\int_{V_p} \mathbf{r} \times \mathbf{u} dV \right) \\ & + \mathbf{T}_c^{n+1/2}. \end{aligned} \tag{8b}$$

The regularized Dirac delta function of Roma et al. [24] guarantees that the total force and torque that the fluid and particles exert onto each other, are preserved in the interpolation and spreading operations defined by Eqs. (7a) and (7c). This holds only when in *each* coordinate direction the corresponding dimension of the Eulerian grid cells is spatially uniform [22]. Therefore

the Eulerian grid is taken uniform. In the present study the Eulerian grid is a Cartesian grid for which the grid cells are cubical with $\Delta x = \Delta y = \Delta z$.

Since the Eulerian and Lagrangian quantities are related to each other through Eqs. (7a) and (7c), it is desired that the resolutions match each other. Given the uniform Eulerian grid, the Lagrangian grid cells are therefore also uniformly distributed over the surface of the spheres. The volume ΔV_l of the Lagrangian grid cells is determined from the requirements that: (1) ΔV_l is as close as possible equal to Δx^3, (2) the number N_L of Lagrangian grid cells over the surface of a sphere is an integer number and (3) the radial thickness of the Lagrangian grid cells is equal to Δx. In summary:

$$N_L = \left[\frac{(R - r_d + \Delta x/2)^3 - (R - r_d - \Delta x/2)^3}{3\Delta x^3/(4\pi)} \right], \tag{9a}$$

$$\Delta V_l = \frac{(R - r_d + \Delta x/2)^3 - (R - r_d - \Delta x/2)^3}{3N_L/(4\pi)}, \tag{9b}$$

where the square brackets in Eq. (9a) denote the nearest integer value of the enclosed expression and r_d is the so-called retraction distance that will be defined later on.

For reason of computational efficiency the IBM forcing is not applied to the interior of the particles, but only to a single shell of Lagrangian grid cells located at the surface of the particles. The uniform distribution of the Lagrangian grid cells over the surface of a particle is generated by means of a separate simulation that starts from an arbitrary distribution of Lagrangian grid points which are given an electrical charge [31]. The charged points repel each other and are allowed to move over the surface of the particle. After sufficiently long time an equilibrium is reached in which the grid points are uniformly distributed over the surface of the particle.

4. Improvements to original method

4.1. Multidirect forcing scheme

The use of a regularized Dirac delta function for the interpolation and spreading operations results in a diffuse distribution of the IBM force around the interface of a particle. This is illustrated in Fig. 2. The arrows indicate all the Eulerian grid points that are involved in the forcing of the velocity at the Lagrangian grid points on the interface. The two circles show the

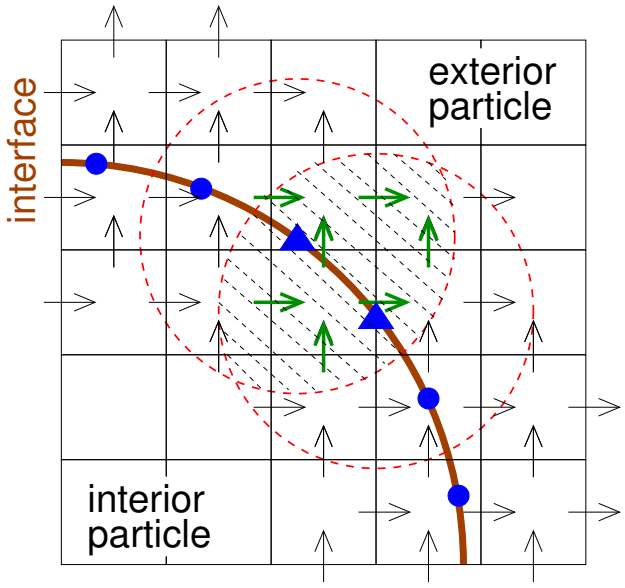

Figure 2: Illustration of the diffuse distribution of the IBM force around the interface of a particle. The dots and triangles on the thick line indicate the locations of the Lagrangian grid points on the interface of the particle. The arrows indicate the locations of the Eulerian grid points on which the IBM force is non-zero. The circles show the range of action of the regularized Dirac delta function for two neighboring Lagrangian grid points indicated by the two triangles. The arrows in the shaded area are Eulerian grid points that are involved in the forcing of the desired particle velocity at the locations of both triangles.

range of action of the delta function for the two Lagrangian grid points indicated by the triangles. The circles overlap each other, meaning that the Eulerian grid points in the overlap region are used to force the velocity at *both* Lagrangian grid points. At these Eulerian grid points the forcing required for the desired particle velocity at one of the Lagrangian grid points is perturbed by the forcing needed for the other and vice versa. As a consequence of this overlap in forcing, the distribution of the IBM force around the interface of the particle may not very well enforce the desired particle velocity at the Lagrangian grid points.

Luo et al. [17] and Kriebitzsch et al. [16] proposed a *multidirect forcing* scheme as a remedy for this problem. The idea is to iteratively determine the IBM forces on the involved Eulerian grid points such that they *collectively* enforce $\mathbf{U}^{**} \approx \mathbf{U}_p$ at the Lagrangian points with a certain desired accuracy.

In between Eqs. (6b) and (6c) the following iterative scheme is included:

do s=1,N$_s$

$$\mathbf{U}_l^{**,s-1} = \sum_{ijk} \mathbf{u}_{ijk}^{**,s-1}\, \delta_d\left(\mathbf{x}_{ijk} - \mathbf{X}_l^n\right) \Delta x \Delta y \Delta z, \tag{10a}$$

$$\mathbf{F}_l^{n+1/2,s} = \mathbf{F}_l^{n+1/2,s-1} + \frac{\mathbf{U}_p\left(\mathbf{X}_l^n\right) - \mathbf{U}_l^{**,s-1}}{\Delta t}, \tag{10b}$$

$$\mathbf{f}_{ijk}^{n+1/2,s} = \sum_{l} \mathbf{F}_l^{n+1/2,s}\, \delta_d\left(\mathbf{x}_{ijk} - \mathbf{X}_l^n\right) \Delta V_l, \tag{10c}$$

$$\mathbf{u}^{**,s} = \mathbf{u}^{*} + \Delta t \mathbf{f}^{n+1/2,s}, \tag{10d}$$

enddo

where N_s is the total number of force iterations and $\mathbf{u}^{**,0}$ is the second prediction velocity as computed from Eq. (6b). The original method of Uhlmann [31] corresponds to the case of $N_s = 0$. The value of N_s can be chosen at will, but for retaining the computational efficiency of the method it should preferably be kept low.

4.2. Inward retraction of Lagrangian grid

Another consequence of the use of a regularized delta function is that the sharp interface of the particle is replaced by a thin porous shell, which for the particular delta function of Roma et al. [24] has a width of three grid cells. This porous shell affects the drag force experienced by the fluid phase in different ways. On the one hand, it increases the (outer) radius of a spherical particle from R to $R + 3\Delta x/2$, which tends to increase the drag force. On the other hand, it is well known that wall porosity tends to decrease the drag force, at least when the flow close to the wall is laminar [2]. The simulation results shown later in this paper indicate that the former effect is stronger than the latter. Consequently, when the Lagrangian grid points are located exactly on the surface of the particle, the effective particle diameter is larger than the actual particle diameter. To correct for this the Lagrangian grid is slightly retracted from the surface towards the interior of the particle [11, 36]. This is illustrated in Fig. 3.

Where the IBM forcing is applied, the fluid is solidified and forced to follow the rigid-body motion of the particle. Ideally, the drag force experienced by the flow around the solidified fluid is the same as the drag force $\mathbf{F}_d$ experienced by the flow around a true particle. In Appendix A it is argued

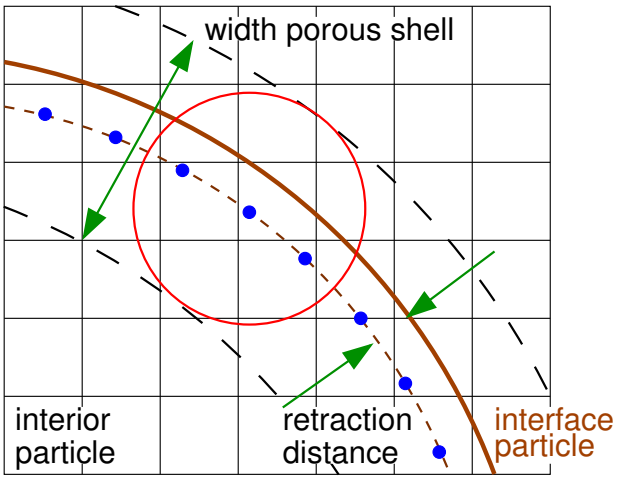

Figure 3: Illustration of the porous shell covering a solid particle. The dots indicate the position of Lagrangian grid points, which are retracted from the actual interface (the solid line) with a fraction of the Eulerian grid spacing (about $0.3\Delta x$ in this case). The circle depicts the range of action of the regularized Dirac delta function.

that the retraction distance can be chosen such that by good approximation $\mathbf{F}_d$ is equal to:

$$\mathbf{F}_d \approx -\rho_f \sum_{ijk} \mathbf{f}_{ijk} \Delta x \Delta y \Delta z + \rho_f \frac{d}{dt} \left(\int_{V_p} \mathbf{u} dV \right) + V_p \nabla p_e, \qquad (11)$$

where the sum on the right-hand side is over all IBM forces *including* that part of the distribution of $\mathbf{f}$ that is located outside the particle.

It can be expected that the optimal retraction distance scales with the width of the regularized delta function. Since in the present study the width of the delta function is equal to $3\Delta x$, it should therefore scale with Δx. In the limit of $\Delta x \to 0$, the retraction then goes to zero and Eq. (11) becomes exact. The optimal retraction distance can be determined from simulations as will be shown later in this paper. The original method of Uhlmann [31] corresponds to the case of zero retraction.

4.3. Direct account of fluid inertia within particle

For reason of computational efficiency Uhlmann [31] assumed rigid-body motion of the fluid *on the surface* of the particles in order to simplify the volume integral in Eq. (8a). This assumption is not entirely valid, but is

expected to be a good approximation when in the modified IBM the number of iterations of the multidirect forcing scheme is chosen sufficiently large. In that case the second term on the right-hand side of Eq. (8a) can be replaced by $\rho_f V_p d\mathbf{u}_c/dt$ [31]. Similarly, by assuming rigid-body motion of the fluid *throughout the interior* of the particles, the second term on the right-hand side of Eq. (8b) can be replaced by $(\rho_f/\rho_p) I_p d\omega_c/dt$ [31]. The error in this assumption is expected to be larger since in the IBM forcing is applied only at the interface of the particle. However, apart from a loss of accuracy associated with the errors in these assumptions, they cause singularities in the Newton-Euler equations for a mass density ratio $\rho_p/\rho_f \rightarrow 1$. It is for this reason that Uhlmann found that for spheres his method is only stable for mass density ratios larger than about 1.2 [31].

Kempe et al. [15] recognized the singularity problem caused by the assumption of rigid-body motion. They proposed to directly evaluate the volume integrals by means of a second-order accurate midpoint rule. For instance, the momentum integral in Eq. (8a) is computed as:

$$\int_{V_p} \mathbf{u} dV \;\; = \;\; \sum_{ijk} \mathbf{u}_{ijk} \alpha_{ijk} \Delta x \Delta y \Delta z. \tag{12}$$

Here α_{ijk} is the solid volume fraction of the particle in an Eulerian grid cell with index (i, j, k). Kempe et al. [15] determine α_{ijk} from a level-set function ϕ given by the signed distance to the particle surface ∂V with $\phi < 0$ inside and $\phi > 0$ outside the particle. The solid volume fraction is calculated from:

$$\alpha_{ijk} \;\; = \;\; \frac{\sum_{n=1}^{8} -\phi_n H\left(-\phi_n\right)}{\sum_{n=1}^{8} |\phi_n|}, \tag{13}$$

where the sum is over all 8 corner nodes of the grid cell volume and H is the Heaviside step function. Kempe et al. [15] validated the second-order accuracy of the midpoint rule.

5. Numerical method

The governing equations for the fluid phase are integrated in time with the explicit low-storage three-step Runge-Kutta method of Wray [29, 34] for all terms except the pressure gradient in the Navier-Stokes equations. For the latter the Crank-Nicolson scheme is used. The advancement of the solution

from time step n to $n+1$ is given by the following pressure-correction scheme:

do q=1,3

$$\mathbf{u}^{*} = \mathbf{u}^{q-1} + \frac{\Delta t}{\rho_f}\left(-(\alpha_q+\beta_q)\nabla p^{q-3/2} + \alpha_q \mathbf{rhs}^{q-1} + \beta_q \mathbf{rhs}^{q-2}\right), \tag{14a}$$

$$\mathbf{U}_l^{*} = \sum_{ijk}\ \mathbf{u}_{ijk}^{*}\ \delta_d\left(\mathbf{x}_{ijk} - \mathbf{X}_l^{q-1}\right)\Delta x\Delta y\Delta z, \tag{14b}$$

$$\mathbf{F}_l^{q-1/2,0} = \frac{\mathbf{U}_p\left(\mathbf{X}_l^{q-1}\right) - \mathbf{U}_l^{*}}{\Delta t}, \tag{14c}$$

$$\mathbf{f}_{ijk}^{q-1/2,0} = \sum_{l}\ \mathbf{F}_l^{q-1/2,0}\ \delta_d\left(\mathbf{x}_{ijk} - \mathbf{X}_l^{q-1}\right)\Delta V_l, \tag{14d}$$

$$\mathbf{u}^{**,0} = \mathbf{u}^{*} + \Delta t \mathbf{f}^{q-1/2,0}, \tag{14e}$$

do s=1,N_s

$$\mathbf{U}_l^{**,s-1} = \sum_{ijk}\ \mathbf{u}_{ijk}^{**,s-1}\ \delta_d\left(\mathbf{x}_{ijk} - \mathbf{X}_l^{q-1}\right)\Delta x\Delta y\Delta z, \tag{14f}$$

$$\mathbf{F}_l^{q-1/2,s} = \mathbf{F}_l^{q-1/2,s-1} + \frac{\mathbf{U}_p\left(\mathbf{X}_l^{q-1}\right) - \mathbf{U}_l^{**,s-1}}{\Delta t}, \tag{14g}$$

$$\mathbf{f}_{ijk}^{q-1/2,s} = \sum_{l}\ \mathbf{F}_l^{q-1/2,s}\ \delta_d\left(\mathbf{x}_{ijk} - \mathbf{X}_l^{q-1}\right)\Delta V_l, \tag{14h}$$

$$\mathbf{u}^{**,s} = \mathbf{u}^{*} + \Delta t \mathbf{f}^{q-1/2,s}, \tag{14i}$$

enddo

$$\nabla^2 \tilde{p} = \frac{\rho_f}{(\alpha_q+\beta_q)\Delta t}\nabla\cdot\mathbf{u}^{**,N_s}, \tag{14j}$$

$$\mathbf{u}^{q} = \mathbf{u}^{**,N_s} - \frac{(\alpha_q+\beta_q)\Delta t}{\rho_f}\nabla\tilde{p}, \tag{14k}$$

$$p^{q-1/2} = p^{q-3/2} + \tilde{p}, \tag{14l}$$

enddo

where the Runge-Kutta step q corresponds to time step n for $q = 0$ and $n+1$ for $q = 3$ and $\mathbf{u}^{**,s}$ is the second prediction velocity at iteration step s of the multidirect forcing scheme. The Runge-Kutta coefficients can be found in Wesseling [34]: $\alpha_1 = 32/60$, $\beta_1 = 0$, $\alpha_2 = 25/60$, $\beta_2 = -17/60$, $\alpha_3 = 45/60$, $\beta_3 = -25/60$.

The equations are discretized in space on a uniform, staggered Cartesian

grid with the finite-volume method in which spatial derivatives are estimated with the central-differencing scheme. The spatially continuous grid without any holes in the interior of the flow domain, enables the use of an efficient, FFT-based, direct solver to compute the correction pressure from Eq. (14j). Stability restrictions for the computational time step of Wray's Runge-Kutta scheme have been derived by Wesseling [34, p. 200]. For a uniform Eulerian grid with $\Delta x = \Delta y = \Delta z$ and the central-differencing scheme, a sufficient criterion for von Neumann stability is given by:

$$\Delta t \;\; \leq \;\; \min\left(\frac{1.65}{12}\frac{\Delta x^2}{\nu_f}, \frac{\sqrt{3}\Delta x}{\sum_{i=1}^{3}|u_i^q|}\right), \tag{15}$$

where $\nu_f \equiv \mu_f/\rho_f$ is the kinematic viscosity of the fluid phase. Unless specified otherwise, in the simulations discussed in this article the computational time step was set equal to half times the maximum allowed time step determined from Eq. (15).

Based on a numerical convergence study of an array of decaying 2D Taylor-Green vortices [31] in the *absence* of particles, it was found by the present author that the above pressure-correction scheme is second-order accurate in space and approximately third-order accurate in time for the velocity, while it is second-order accurate in both space and time for the pressure.

The governing equations for the solid particles are advanced in time with the same Runge-Kutta method and follow directly after Eq. (14l) in the do

loop:

$$\begin{aligned}\mathbf{u}_c^q &= \mathbf{u}_c^{q-1} - \frac{\Delta t}{V_p}\frac{\rho_f}{\rho_p}\sum_l \mathbf{F}_l^{q-1/2,N_s}\Delta V_l \\ &+ \frac{1}{V_p}\frac{\rho_f}{\rho_p}\left(\left\{\int_{V_p}\mathbf{u}dV\right\}^q - \left\{\int_{V_p}\mathbf{u}dV\right\}^{q-1}\right) \\ &+ (\alpha_q+\beta_q)\,\Delta t\left(1-\frac{\rho_f}{\rho_p}\right)\mathbf{g} + \left(\frac{\alpha_q+\beta_q}{2}\right)\Delta t\frac{(\mathbf{F}_c^q+\mathbf{F}_c^{q-1})}{(\rho_p V_p)},\end{aligned} \tag{16a}$$

$$\mathbf{x}_c^q = \mathbf{x}_c^{q-1} + \left(\frac{\alpha_q+\beta_q}{2}\right)\Delta t\left(\mathbf{u}_c^q+\mathbf{u}_c^{q-1}\right), \tag{16b}$$

$$\begin{aligned}\omega_c^q &= \omega_c^{q-1} - \Delta t\frac{\rho_f}{I_p}\sum_l \mathbf{r}_l^{q-1}\times\mathbf{F}_l^{q-1/2,N_s}\Delta V_l \\ &+ \frac{\rho_f}{I_p}\left(\left\{\int_{V_p}\mathbf{r}\times\mathbf{u}dV\right\}^q - \left\{\int_{V_p}\mathbf{r}\times\mathbf{u}dV\right\}^{q-1}\right) \\ &+ \left(\frac{\alpha_q+\beta_q}{2}\right)\Delta t\frac{(\mathbf{T}_c^q+\mathbf{T}_c^{q-1})}{I_p},\end{aligned} \tag{16c}$$

$$\mathbf{U}_p\left(\mathbf{X}_l^q\right) = \mathbf{u}_c^q + \omega_c^q\times\left(\mathbf{X}_l^q-\mathbf{x}_c^q\right). \tag{16d}$$

It is remarked that the Euler angles do not need to be calculated for solid particles with rotational symmetry such as spheres, since in this case there is no need to rotate the Lagrangian grid along with the rotation of the particle; the relative position vector $\mathbf{r}_l^q = \mathbf{X}_l^q - \mathbf{x}_c^q$ is independent of time.

Note that the Navier-Stokes equations are advanced from step $q-1$ to step q given the particle velocity and position at step $q-1$, see Eq. (14c). This is known as a *fully explicit* coupling of the Navier-Stokes and Newton-Euler equations and has an effect on the temporal accuracy as well as the numerical stability of the scheme. Firstly, it can be expected that the forcing of $\mathbf{U}^{**}$ to $\mathbf{U}_p^{q-1}$ instead of $\mathbf{U}_p^q$ results in an error of $O(\Delta t)$. Secondly, the explicit coupling poses a lower limit for the particle-fluid mass density ratio due to an added-mass effect as pointed out by Hu et al. [12]. For the case of the inertial migration of a buoyant sphere in tube Poiseuille flow, which is discussed in section 6.3.3, it was found that the present method becomes unstable when $\rho_p/\rho_f \lesssim 0.3$. Except for stability problems for low particle-fluid mass density ratio, it is the experience of the author that particles do

not affect the numerical stability [23]. In the presence of particles the same time step criterion (15) is used as in the absence of particles.

The computational algorithm detailed in this section, is coded in Fortran with the MPI extension for parallel execution on multi-processor machines with distributed memory. For the parallelization of the Navier-Stokes equations a standard domain decomposition method is used. For the parallelization of the particle-related subroutines use was made of a master-slave method, which was implemented in a similar fashion as described by Uhlmann [30].

6. Results

In this section the accuracy of the present IBM is demonstrated for both *fixed* and *freely moving* spheres. Flows containing just one sphere as well as flows with two interacting spheres are considered. The influence of the number of force iterations (N_s) and of the retraction distance (r_d) on the numerical accuracy is discussed. The improved numerical stability of the present method is demonstrated for the migration of both neutrally-buoyant and buoyant spheres in tube Poiseuille flow.

6.1. Simple cubic lattice of fixed spheres

For laminar flow through a simple cubic lattice of fixed spheres, it suffices to simulate a single sphere positioned in the center of a fully periodic cubical flow domain. The dimension of the cubical domain is taken equal to twice the particle diameter, which corresponds to a fluid volume fraction of $\epsilon = 1-(\pi/48) \approx 0.935$. Since the sphere is held fixed in space, the Newton-Euler equations do not need to be solved. The desired velocity at the position of the Lagrangian grid points is simply $\mathbf{U}_p = \mathbf{0}$. The flow is driven by a constant pressure gradient in the x-direction, $dp_e/dx = -0.2336\rho_f\nu_f^2/D^3$, where D is the sphere diameter. Periodic boundary conditions are applied to the (prediction) velocity and (correction) pressure at the boundaries of the flow domain. The Reynolds number, $Re \equiv U_b D/\nu_f$, based on the so-called superficial bulk velocity U_b, is in all simulations smaller than 0.1, so the flow is in the Stokes regime. The superficial bulk velocity is defined as the streamwise velocity averaged over the entire volume of the flow domain, including the space occupied by the sphere.

The simulations are started from an initially uniform flow field. With time the flow adapts itself to the flow geometry and imposed pressure gradient

and after the initial transient it reaches a steady state. Note that for steady flows the correction pressure $\tilde{p}$ becomes zero and hence the second prediction velocity $\mathbf{u}^{**}$ is then equal to the velocity $\mathbf{u}$, see Eq. (14k). This eliminates the error resulting from forcing the *second prediction* velocity to the particle velocity instead of forcing the *actual* fluid velocity, see Eqs. (14e) and (14i).

To illustrate the flow pattern, Fig. 4 depicts a cross-section of the flow at a resolution of $D/\Delta x = 16$, $N_s = 2$ force iterations and a retraction distance of $r_d = 0.25\Delta x$. Within the sphere two counter-rotating recirculation cells

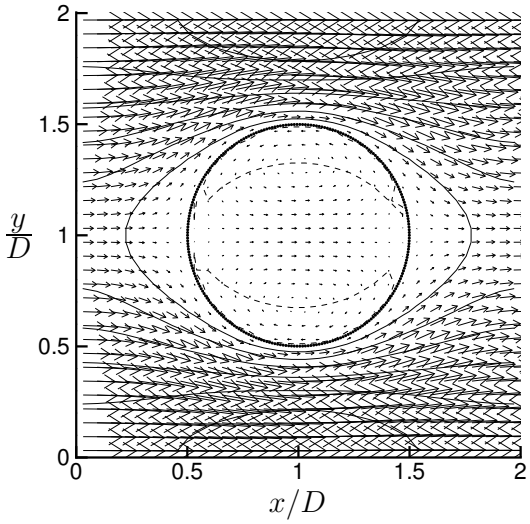

Figure 4: Cross-section of Stokes flow through a simple cubic lattice of fixed spheres at a solid volume fraction of $\pi/48$. The thin lines are contours of the streamwise velocity with a contour interval of $0.2\,U_b$ and are dashed when smaller or equal to zero. The circle shows the position of the sphere. The resolution of the Eulerian grid is $D/\Delta x = 16$, the number of force iterations is $N_s = 2$ and the retraction distance is $r_d = 0.25\Delta x$. The number of Lagrangian grid cells is $N_L = 756$.

are present. They originate from the fact that the driving pressure gradient, dp_e/dx, is applied *everywhere* inside the domain, including the space occupied by the sphere. However, the details of the flow inside the sphere are not important as long as the velocity *on* the surface of the sphere satisfies the desired ns/np condition to good approximation.

6.1.1. Effect of multidirect forcing

To investigate the effect of multidirect forcing on the numerical accuracy, the retraction distance was fixed at $r_d = 0.25\Delta x$. Multiple simulations have

been executed at various grid resolutions ($D/\Delta x$ = 16, 24 and 36) and for different numbers of force iterations (N_s = 0..10). The velocity field obtained from the simulations was interpolated to the Lagrangian grid at the surface of the sphere according to Eq. (7a) with $\mathbf{u}^*_{ijk}$ replaced by $\mathbf{u}_{ijk}$. Ideally, the velocity $\mathbf{U}_l$ at the Lagrangian grid would be zero. Fig. 5.a shows the maximum value of $\|\mathbf{U}_l\|$ normalized by the so-called intrinsic bulk velocity, which is the superficial bulk velocity divided by the fluid volume fraction (U_b/ϵ). The error is the weighted sum of the maximum error at each of

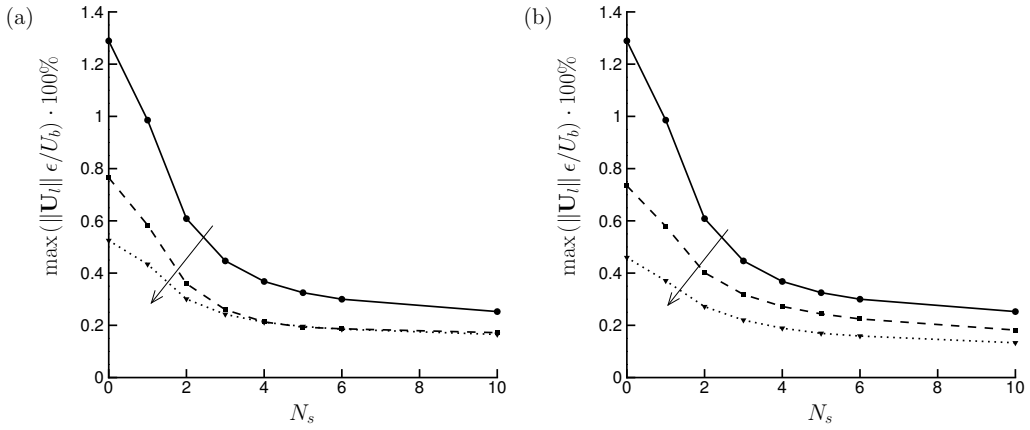

Figure 5: Maximum error in the norm of the Lagrangian velocity $\|\mathbf{U}_l\|$ as function of the total number of force iterations N_s with the retraction distance fixed at $r_d = 0.25\Delta x$. The error is given in percentage of the intrinsic bulk velocity. (a) Effect of grid resolution: —, $D/\Delta x$ = 16; - - -, $D/\Delta x$ = 24; $\cdots$, $D/\Delta x$ = 36. Arrow points in direction of increasing grid resolution. (b) Effect of computational time step with grid resolution fixed at $D/\Delta x = 16$: —, $\Delta t_{16} = 0.06875\Delta x^2/\nu_f$; - - -, $(16/24)^2\Delta t_{16}$; $\cdots$, $(16/36)^2\Delta t_{16}$. Arrow points in direction of decreasing time step.

the three Runge-Kutta steps with the weight of step q given by $(\alpha_q + \beta_q)$. Initially, the error decreases rapidly with increasing N_s, but at higher values of N_s the decrease in the error becomes slow. Increasing the value of N_s from 2 to 3 yields a smaller decrease in the error than increasing N_s from 1 to 2. This suggests that $N_s = 2$ is optimal for increasing the accuracy of the IBM with little increase in the computational costs. At a resolution of $D/\Delta x = 16$ the error is reduced by more than a factor 2 after 2 force iterations and equal to about 0.6% of the intrinsic bulk velocity.

For fixed value of N_s the error in the Lagrangian velocity decreases with increasing grid resolution. This behavior might be expected for several rea-

sons. Firstly, the details of the flow are better resolved at higher grid resolution. Secondly, the width of the porous shell surrounding the sphere is equal to $3\Delta x$ and therefore the interface becomes sharper with increasing grid resolution. Thirdly, according to Eq. (15) the time step scales with Δx^2 for Stokes flow. This implies that at a higher grid resolution the time step is smaller, which means that the number of force iterations per unit time effectively increases. Thus, the accuracy of the IBM depends on the computational time step, even when the flow is steady. In Fig. 5.b the grid resolution is fixed at $D/\Delta x = 16$, while the computational time step is varied between the values corresponding to the three grid resolutions shown in Fig. 5.a. The similarity between Figs 5.a and 5.b indicates that the decrease in the error with increasing grid resolution originates for a large part from the decrease in the computational time step. Motivated by this observation, in Fig. 6 the error in the Lagrangian velocity is plotted as function of the computational time step for various values of N_s. The dashed lines represent the data of Fig. 5.a, i.e. the time step was varied by varying the spatial grid resolution. The solid lines represent the data of Fig. 5.b with the spatial grid resolution fixed at $D/\Delta x = 16$. The data for varying and fixed spatial grid resolution show a similar trend, at least for the larger time steps. This figure suggests that the error in the Lagrangian velocity scales with the time step to a power of roughly 0.5. The number of force iterations appears to have little influence on the order of convergence, with the order being closer to 0.6 for $N_s = 0$ and 0.4 for $N_s = 10$. Note that the number of force iterations has, however, a strong influence on the vertical position of the error curve.

Fig. 7 shows the effect of multidirect forcing on the so-called Darcy number. The Darcy number is defined as the permeability of the lattice of spheres normalized by the square of the sphere diameter. It is computed from the simulations as follows:

$$Da \;=\; \frac{\mu_f U_b}{(-dp_e/dx)\, D^2}. \tag{17}$$

The higher the Darcy number, the stronger the ability of a porous medium to transmit fluid through its pores. Since μ_f, D and dp_e/dx are fixed in the simulations, the error in the Darcy number is proportional to the error in the superficial bulk velocity. It is thus a measure of the global error in the velocity field. The Darcy number varies mostly in the first 2 force iterations. Further increasing N_s shows only little influence on the Darcy number. This reaffirms that 2 force iterations seem optimal.

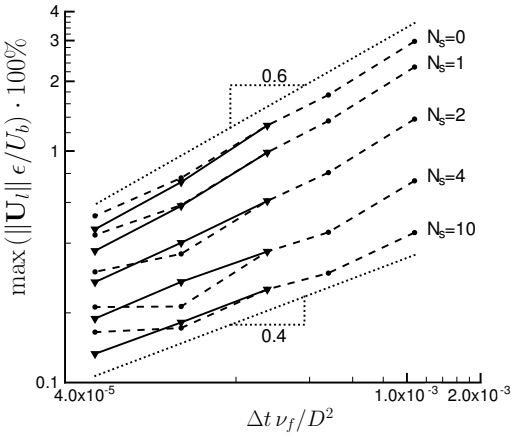

Figure 6: Maximum error in the norm of the Lagrangian velocity $\|\mathbf{U}_l\|$ as function of the computational time step for various values of N_s and with $r_d = 0.25\Delta x$ fixed. The error is given in percentage of the intrinsic bulk velocity. Dashed lines with dots: data from Fig. 5.a with extra data points for $D/\Delta x = 8$ and 12. Solid lines with gradient symbols: data from Fig. 5.b with the grid resolution fixed at $D/\Delta x = 16$.

For fixed value of N_s the Darcy number increases with increasing grid resolution. Fig. 7.b shows that with decreasing time step the Darcy number decreases, but the effect is relatively small. This is in contrast with the relatively large influence of the time step on the maximum error in the Lagrangian velocity. Fig. 7 suggests that the error in the Darcy number is predominantly controlled by the spatial resolution and the width of the porous shell around the particle. At higher grid resolution the Darcy number increases, which indicates a decrease in the effective particle diameter with increasing grid resolution.

Fig. 8.a depicts the percentual error in the Darcy number as function of the grid resolution and various values of N_s. For every value of N_s the error is computed relative to the value Da_r obtained from Richardson extrapolation [6]. As can be observed from Fig. 8.b, the extrapolated value of the Darcy number varies less than 0.1 % for the investigated range of N_s. The order of grid convergence is also fairly insensitive to the value of N_s; it varies from 1.75 for $N_s = 0$ till 1.65 for $N_s = 10$. Surprisingly, the percentual error increases with increasing N_s, though the increase is relatively small; at fixed grid resolution the relative error is of the same order for all values of N_s.

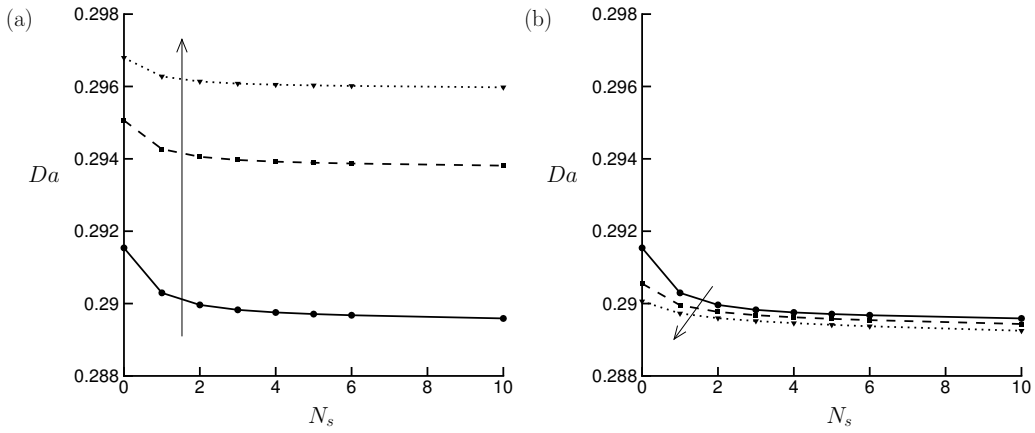

Figure 7: Darcy number Da as function of the total number of force iterations N_s with the retraction distance fixed at $r_d = 0.25\Delta x$. (a) Effect of grid resolution: —, $D/\Delta x = 16$; - - -, $D/\Delta x = 24$; $\cdots$, $D/\Delta x = 36$. Arrow points in direction of increasing grid resolution. (b) Effect of computational time step with grid resolution fixed at $D/\Delta x = 16$: —, $\Delta t_{16} = 0.06875\Delta x^2/\nu_f$; - - -, $(16/24)^2\Delta t_{16}$; $\cdots$, $(16/36)^2\Delta t_{16}$. Arrow points in direction of decreasing time step.

6.1.2. Effect of retraction

To investigate the effect of retraction on the numerical accuracy, the number of force iterations was fixed at $N_s = 2$. Multiple simulations have been executed at various grid resolutions ($D/\Delta x = 16$, 24 and 36) and for different values of the retraction distance ($r_d = 0\,..\,0.5\Delta x$). The velocity field obtained from the simulations was interpolated to the Lagrangian grid at the surface of the sphere. Fig. 9 shows the maximum error in the norm of the Lagrangian velocity as function of the retraction distance. The error decreases with increasing grid resolution. For fixed resolution, however, the error appears fairly insensitive to changes in the retraction distance. This behavior might also be expected in view of the nature of the error in the Lagrangian velocity as discussed in section 4.1. The small variation in the error with a change in the retraction distance is possibly caused by a small variation in the convergence speed of the multidirect forcing scheme with a change in the positions of the Lagrangian grid points relative to the surrounding Eulerian grid points.

Fig. 10 shows the Darcy number as function of the retraction distance. The dash-dotted line is the Darcy number Da_r obtained from Richardson

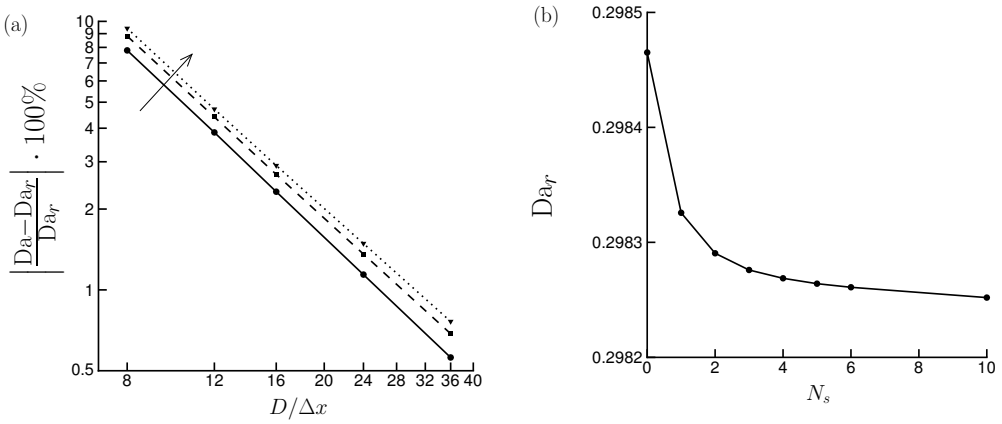

Figure 8: (a) Percentual error in Darcy number Da as function of the grid resolution at a retraction distance of $r_d = 0.25\Delta x$. The error is relative to the value of the Darcy number Da_r obtained from Richardson extrapolation using the data points of $D/\Delta x = 16$, 24 and 36. The lines show the effect of the number of force iterations: —, $N_s = 0$; - - -, $N_s = 1$; $\cdots$, $N_s = 10$. Arrow points in direction of increasing value of N_s. (b) Da_r as function of N_s.

extrapolation and is considered as an estimate of the exact solution. For a retraction distance up to about 0.3, Da_r varies only slightly with r_d with $Da_r = 0.299$ at $r_d = 0.3\Delta x$. This value is in close agreement with the analytical solution of Hasimoto [10] of $Da = 0.297$ accurate up to $O\left([1-\epsilon]^2\right)$. The close agreement supports the postulation in Appendix A that an optimal retraction distance exists for which the right-hand side of Eq. (A.3) is a good approximation of the flow-induced drag force experienced by a true sphere of radius R.

Fig. 10.a shows that for $r_d \lesssim 0.3\Delta x$ the Darcy number is underpredicted, while it is overpredicted for $r_d \gtrsim 0.35\Delta x$. The higher the grid resolution, the smaller the error. In the range of $0.35 \lesssim r_d \lesssim 0.39$ the Darcy number does not vary monotonically with the grid resolution and does only weakly depend on grid resolution.

In Fig. 10.b the retraction distance is scaled with $D/2$ instead of Δx. At all three grid resolutions the Darcy number varies with the retraction distance in a similar way, which is approximately linear over the small range of r_d investigated. From this figure the difference between the effective and the actual particle diameter can be observed. For $r_d = 0$ the excess in effective

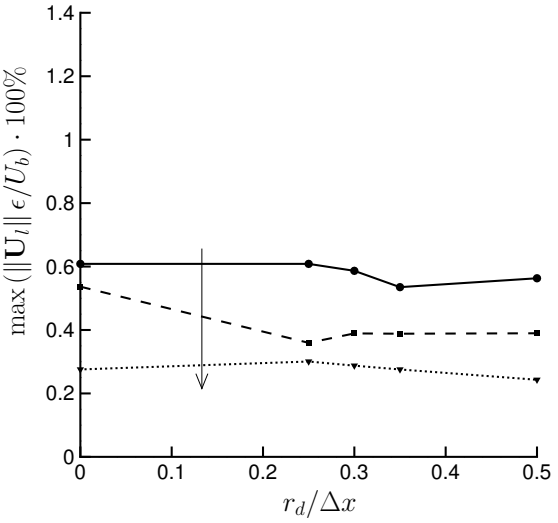

Figure 9: Maximum error in the norm of the Lagrangian velocity $\|\mathbf{U}_l\|$ as function of the retraction distance $r_d/\Delta x$ with the total number of force iterations fixed at $N_s = 2$. The error is given in percentage of the intrinsic bulk velocity. The lines represent different grid resolutions: —, $D/\Delta x = 16$; - - -, $D/\Delta x = 24$; $\cdots$, $D/\Delta x = 36$. Arrow points in direction of increasing grid resolution.

particle diameter relative to the actual particle diameter (D) is indicated by a vertical line for each of the three grid resolutions: $0.043D$ at $D/\Delta x = 16$, $0.027D$ at $D/\Delta x = 24$ and $0.018D$ at $D/\Delta x = 36$. Thus, for $r_d = 0$ the effective particle diameter appears to scale approximately linearly with the grid resolution.

Fig. 11 depicts the percentual error in the Darcy number as function of the grid resolution and various values of r_d. The retraction distance has a strong influence on the error as well as on the order of grid convergence of the IBM. The black line corresponds to the original IBM with zero retraction for which the method is approximately first-order accurate. This is consistent with the approximately linear relation between the effective particle diameter and the grid resolution for $r_d = 0$ as observed from Fig. 10.b. Interestingly, for $r_d = 0.3\Delta x$ the method becomes approximately second-order accurate. At a grid resolution of $D/\Delta x = 16$ the error is $\approx 1.5\%$, which is almost 7 times smaller than the error of $\approx 9.8\%$ at zero retraction. Along with the increase in accuracy, there is also a slight increase in the efficiency of the method. At a resolution of $D/\Delta x = 16$ the number of Lagrangian grid

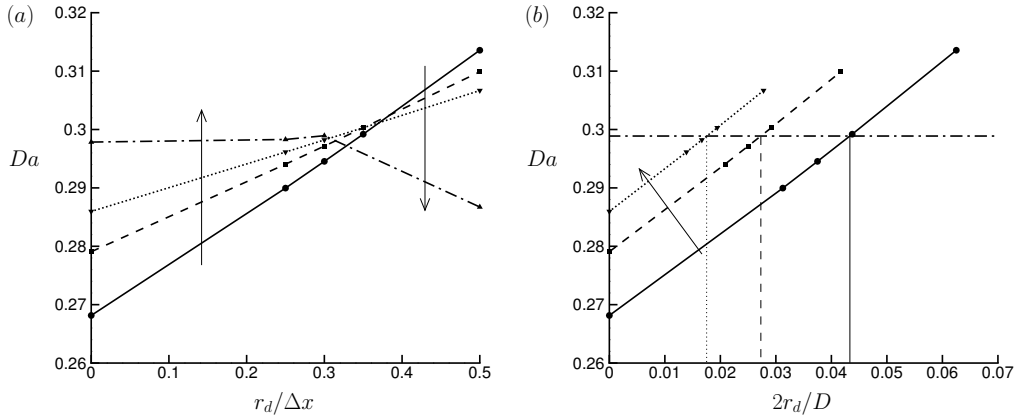

Figure 10: Darcy number Da as function of retraction distance r_d. The lines represent different grid resolutions: —, $D/\Delta x = 16$; - - -, $D/\Delta x = 24$; $\cdots$, $D/\Delta x = 36$; –·–·–, Da_r obtained from Richardson extrapolation. Arrow points in direction of increasing grid resolution. (a) Da as function of $r_d/\Delta x$. (b) Da as function of $2r_d/D$. The dash-dotted line corresponds to the value Da_r found from Richardson extrapolation at $r_d/\Delta x = 0.3$. The vertical lines indicate for each grid resolution at $r_d = 0$ the excess in effective sphere diameter with respect to the actual sphere diameter D.

points decreases from $N_L = 805$ to $N_L = 746$ when the retraction distance is increased from $r_d = 0$ to $r_d = 0.3\Delta x$, which corresponds to a drop in N_L of about 7.3%. For a retraction distance of $r_d \gtrsim 0.35$ the error does not display a power-law dependence of the grid resolution.

Based on Figs 10 and 11 it is suggested that a retraction distance of $r_d = 0.3\Delta x$ is close to optimal for maximizing the accuracy of the present IBM. Interestingly, Höfler and Schwarzer [11] suggested for their IBM the same value for the retraction distance in case of spheres. Furthermore, Yu and Shao [36] mentioned a similar value of $\Delta x/3$ as a good choice for the retraction distance in their direct-forcing fictitious domain method.

Yu and Shao [36] showed for their method that the optimal retraction distance depends on the Reynolds number and the shape of the particle. For the present IBM the effect of the Reynolds number appears small. Simulations of flow through a simple cubic lattice of fixed spheres at $Re \approx 60$ indicated no significant change in the optimal retraction distance, though the flow pattern exhibits 2 recirculation cells in between the spheres (not shown) and is thus quite different from the Stokes-flow pattern depicted in Fig. 4. The shape of the particle, however, does have some influence. Based on simulations of Stokes flow through a simple cubic lattice of fixed *cubes* at

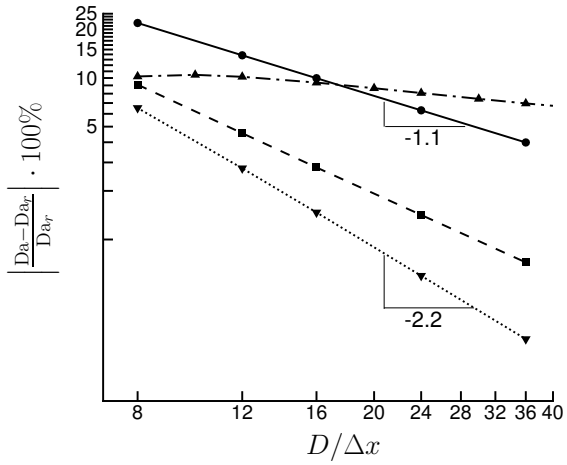

Figure 11: Percentual error in Darcy number Da as function of the grid resolution at a number of force iterations of $N_s = 2$. The error is relative to the value of the Darcy number Da_r obtained from Richardson extrapolation using the data points of $D/\Delta x = 16$, 24 and 36. The lines show the effect of the retraction distance: —, $r_d/\Delta x = 0$; - - - , $r_d/\Delta x = 0.25$; $\cdots$, $r_d/\Delta x = 0.3$; $-\cdot-\cdot-$, $r_d/\Delta x = 0.5$.

$\epsilon = 0.875$, it was found that the optimal retraction distance is slightly larger and close to $0.4\Delta x$ (not shown).

As already mentioned in the introduction section, the first-order accuracy of the original IBM for which $r_d = 0$ originates from the first-order accuracy of the interpolation/spreading operations for a non-smooth velocity field. However, for $r_d > 0$ the velocity field around the interface of a particle remains non-smooth, see for example Fig. 4 for $r_d = 0.25\Delta x$. This poses the question how to interpret the increased accuracy of the present IBM for this case. Retraction apparently cancels in part the interpolation/spreading errors such that for $r_d = 0.3\Delta x$ second-order accuracy is obtained.

6.2. Freely moving sphere in plane Poiseuille flow

To demonstrate the accuracy of the present IBM for moving particles, the case is considered of a freely moving sphere in upward plane Poiseuille flow. The present case has been simulated earlier by Uhlmann [32]. The flow domain is bounded by a solid wall at $z = 0$ and a plane free surface at $z = h$. The flow is driven by a vertical pressure gradient $(-dp_e/dy > 0)$ and dynamically adjusted in time such that the Reynolds number remains constant at $Re \equiv V_b h/\nu_f = 1000$ with V_b the superficial bulk velocity. At

this value of the Reynolds number the flow is laminar. The dimensions of the flow domain are $0.5h \times 0.5h \times h$ in the x-, y- and z-direction, respectively. The boundary conditions for the (prediction) velocity are periodic in the x- and y-direction, the no-slip/np-penetration conditions at the solid wall and the free-slip/no-penetration conditions at the plane free surface. The boundary conditions for the (correction) pressure are periodic in the x- and y-direction and the homogeneous Neumann condition at the solid wall and plane free surface. The initial conditions for the flow field are $u = w = 0$ and $v(z)/V_b = (3/2)\,(z/h)\,(2 - z/h)$.

At $t = 0$ a sphere with diameter $D/h = 1/20$ is placed in the center of the flow domain at $x_c/h = y_c/h = 0.25$ and $z_c/h = 0.5$. The particle-fluid mass density ratio is set to $\rho_p/\rho_f = 4.17$. The gravitational acceleration in the y-direction is set to $g_y D/V_b^2 = -1.1036$. The initial translational velocity of the particle is set to the velocity of the fluid phase at $y/h = 0.5$, $v_c/V_b = 9/8$. Similarly, the initial angular velocity of the particle is set to the angular velocity (= half the vorticity) of the fluid phase at $y/h = 0.5$, $\omega_{cx} h/V_b = -3/4$. It is remarked that the present initial condition for the angular velocity is different from Uhlmann's choice of $\omega_{cx} = 0$, which is not compatible with the angular velocity of the fluid phase and not consistent with the choice for the translational velocity.

Fig. 12 illustrates the flow pattern at $t = t_*$ with $t_* V_b/h = 0.9990$, which corresponds to 1488 time steps in this case with $D/\Delta x = 16$, $N_s = 2$ and $r_d = 0.3\Delta x$. The plane-averaged vertical velocity profile in Fig. 12.a is distorted near the position of the particle, but remains parabolic in the rest of the flow domain. Fig. 12.b shows a cross-section of the flow field around the sphere in a frame moving with the velocity of the sphere. The particle Reynolds number based on the particle diameter and the velocity difference of the particle relative to the parabolic velocity profile is ≈ 130 at $t = t_*$ and thus inertial effects dominate the flow near the particle. The contours of the vertical velocity nicely show the extent of the wake at the rear of the particle. Within the particle two counter-rotating recirculation cells are present like observed earlier for a fixed particle in Fig. 4. The particle boundary nearly collapses with a velocity contour, which indicates the IBM's capability of enforcing the ns/np conditions at the particle boundary.

In Figs 13–19 simulation results from the present IBM with $r_d/\Delta x = 0.3$ and $N_s = 2$ are compared with the results from the original IBM of Uhlmann [31, 32] for which $r_d = 0$ and $N_s = 0$. Recall that the original IBM assumes rigid-body motion for the fluid on the particle surface and within the interior

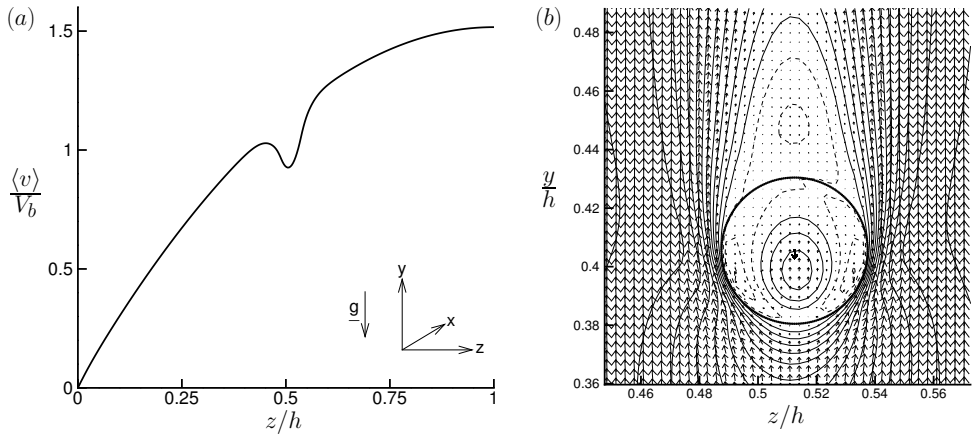

Figure 12: Freely moving sphere in upward plane Poiseuille flow at $tV_b/h = 0.9990$ with $D/\Delta x = 16$, $N_s = 2$ and $r_d = 0.3\Delta x$. (a) Plane-averaged velocity profile $\langle v \rangle / V_b$ as function of the distance to the wall z/h. (b) Flow field around the sphere at $x/h = 0.2516$ in a frame of reference moving with the velocity of the sphere indicated by the black arrow in the center of the sphere. The thin lines are contours of the vertical velocity with a contour interval of 0.2 V_b and are dashed when smaller or equal to zero.

of the particle in order to simplify the volume integrals in Eqs. (8a) and (8b), see section 4.3. Simulations with the original IBM indicate that this assumption is justified for the present flow problem in which the particle-fluid mass density ratio is relatively high; results will be shown later. The differences in the simulation results of the present and the original IBM are therefore related only to the effect of retraction and multidirect forcing. The choice of $r_d/\Delta x = 0.3$ and $N_s = 2$ for the present IBM is based on the optimal values found in the previous section for Stokes flow through a simple cubic lattice of spheres. The comparison between the two methods focusses on the motion of the particle since the aim of this section is to demonstrate the accuracy of the present IBM for moving particles.

Fig. 13 depicts the history of the vertical translational particle velocity v_c from $tV_b/h = 0$ till 1. The lines correspond to different spatial resolutions, ranging from $D/\Delta x = 16$ till 54. The number of grid cells between the highest and lowest resolution differs by a factor of about 38 and is equal to $540 \,\mathrm{x}\, 540 \,\mathrm{x}\, 1080 \approx 314.9 \cdot 10^6$ grid cells at $D/\Delta x = 54$. Because the particle is heavier than the surrounding fluid, the velocity of the particle decreases immediately at the beginning of the simulation. At $tV_b/h \approx 0.109$ the vertical velocity changes sign and is moving in the opposite direction

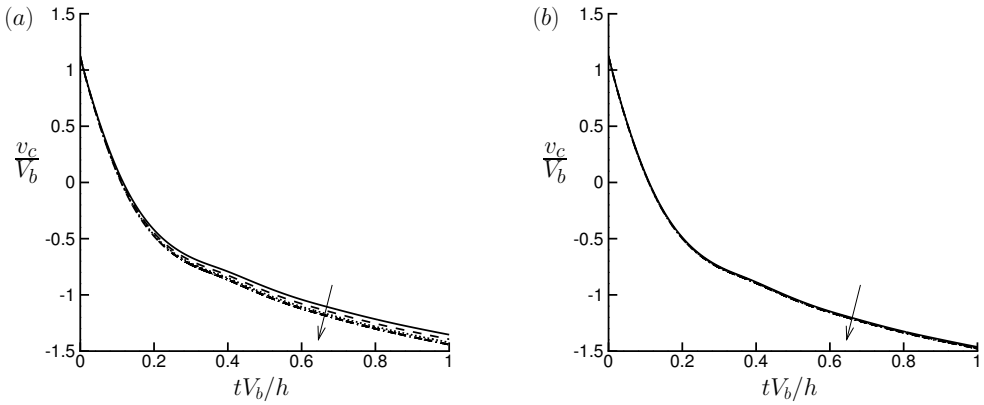

Figure 13: Vertical translational particle velocity v_c/V_b as function of time tV_b/h. The lines represent different grid resolutions: —, $D/\Delta x = 16$; - - -, $D/\Delta x = 24$; $\cdots$, $D/\Delta x = 36$; $-\cdot-\cdot-$, $D/\Delta x = 54$. Arrow points in direction of increasing grid resolution. (a) Original method corresponding to $r_d = 0$ and $N_s = 0$. (b) Present IBM with $r_d/\Delta x = 0.3$ and $N_s = 2$.

henceforth. The results for the original (Fig. 13.a) and the present IBM (Fig. 13.b) are similar, though the present IBM is less sensitive to the grid resolution. This is already an indication of the higher accuracy of the present IBM over the original method.

Fig. 14 shows the time history of the wall-normal translational particle velocity w_c. Except for the original method with $D/\Delta x = 16$, the results show that the wall-normal velocity is positive in the plotted time interval and thus the particle moves away from the solid wall. The wall-normal velocity displays an oscillatory behavior. The influence of the grid resolution on the amplitude of the oscillation becomes apparant for $tV_b/h \gtrsim 0.2$, but the grid resolution has little influence on the period of the oscillation. At $tV_b/h = 1$ the difference between the original and the present IBM is large, though the results from the two methods converge to each other with increasing grid resolution.

Fig. 15 shows the time history of the lateral angular particle velocity ω_{cx}. Initially, the angular velocity varies only little, but from $tV_b/h \approx 0.05 - 0.4$ it's magnitude decreases rapidly by more than a factor 2. The grid resolution has a clear influence on the angular velocity. At $tV_b/h = 1$ the differences between the original and the present IBM are large at the low resolutions: for $D/\Delta x = 16$ the original method predicts that the angular velocity is slightly

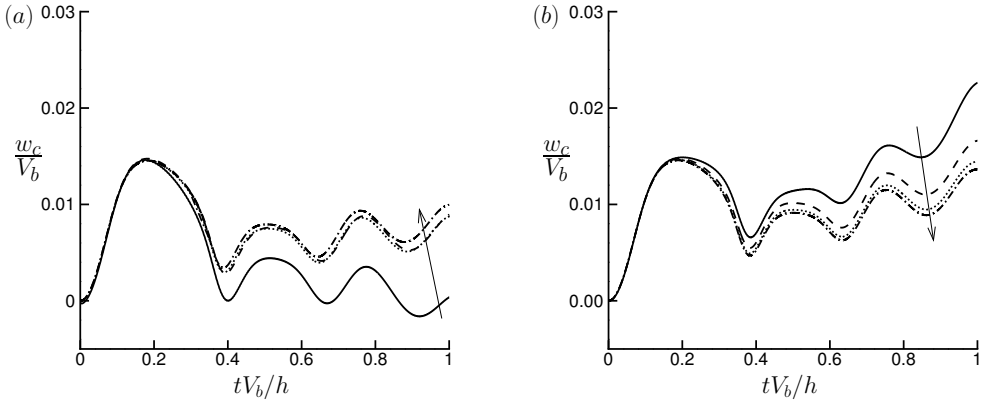

Figure 14: Idem as Fig. 13, but for wall-normal translational particle velocity w_c/V_b.

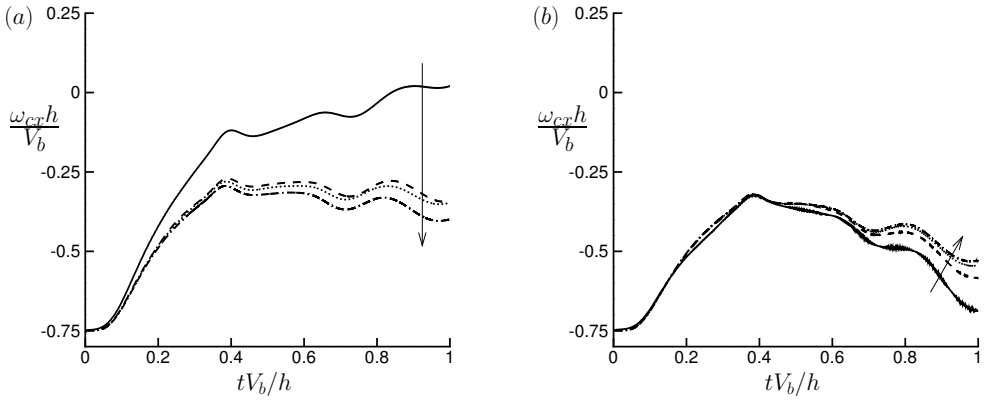

Figure 15: Idem as Fig. 13, but for lateral angular particle velocity $\omega_{cx}h/V_b$.

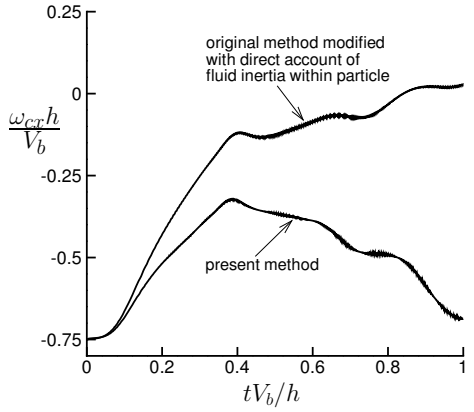

Figure 16: Lateral angular particle velocity $\omega_{cx}h/V_b$ as function of tV_b/h. The upper line denotes the original method modified with a direct account of the fluid inertia within the particle as described in section 4.3. The lower line denotes the present method with $r_d/\Delta x = 0.3$ and $N_s = 2$. Grid resolution is $D/\Delta x = 16$ in both cases.

positive, while the present IBM predicts $\omega_{cx}h/V_b \approx -0.69$.

For $D/\Delta x = 16$ small-amplitude, high-frequency oscillations are visible in Fig. 15.b for the present IBM. These spurious oscillations are actually present in all simulations with both methods and originate from *grid locking* as described in section 1. This explanation is substantiated by the period of the oscillations, which appears to be well described by the time it takes for the particle to travel a distance of one grid spacing ($\Delta x/\|\mathbf{U}_p\|$). For the present IBM grid locking is somewhat more pronounced than in the original method because of an additional effect of the motion of the Lagrangian grid on the computation of the volume integrals on the right-hand side of Eqs. (8a) and (8b). This effect is illustrated in Fig. 16 where for $D/\Delta x = 16$ the present method is compared to the original method modified with a direct account of fluid inertia within the particle instead of assuming rigid-body motion of the fluid within the particle. Now both methods display spurious oscillations of similar amplitude, suggesting that errors in the computation of the volume integrals contribute the most to the amplitude of the oscillations. The amplitude of the oscillations decreases with increasing grid resolution; for $D/\Delta x > 16$ the oscillations are hardly visible in Fig. 15.b. This substantiates the effectiveness of the IBM in suppressing grid locking, as already mentioned in the introduction section.

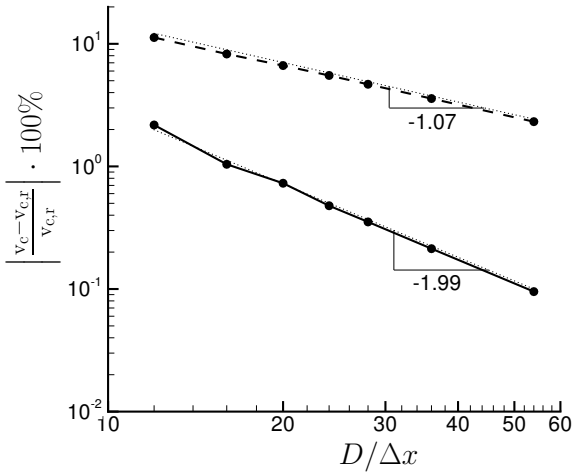

Figure 17: Percentual error in vertical translational particle velocity v_c at $t = t_*$. The exact value $v_{c,r}$ is estimated from Richardson extrapolation using the data points of $D/\Delta x = 24$, 36 and 54. Solid line: present IBM with $r_d/\Delta x = 0.3$ and $N_s = 2$. Dashed line: original IBM corresponding to $r_d = 0$ and $N_s = 0$. Dotted lines show the grid convergence behavior determined from Richardson extrapolation.

For $D/\Delta x = 16$ the result from the original method with the assumption of rigid-body motion is very similar to the result from the original method modified with the direct account of fluid inertia, cf. Figs 15.a and 16. This justifies the assumption of rigid-body motion in the original method for simplifying Eqs. (8a) and (8b), at least for the present flow problem.

Fig. 17 depicts for both the original method and the present IBM the percentual error in the vertical translational particle velocity v_c at $t = t_*$. The error is computed relative to the exact value $v_{c,r}$ estimated from Richardson extrapolation of the data points at resolutions $D/\Delta x = 24$, 36 and 54. For both methods $v_{c,r}/V_b \approx -1.48$. The grid convergence is approximately first order for the original method, while second order for the present IBM. This result is similar to the order of grid convergence found for flow through a simple cubic lattice of fixed spheres, cf. Fig. 11. At $D/\Delta x = 16$ the percentual error is 8.3 % for the original method and 1.0 % for the present IBM, a difference of about a factor 8.

Fig. 18 shows the error in the wall-normal translational particle velocity w_c at $t = t_*$. The error of the original IBM does not behave according to a power law of the grid resolution. For that reason the exact value is

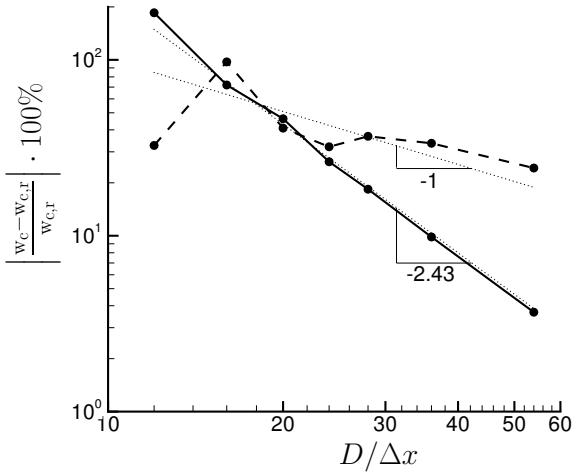

Figure 18: Idem as Fig. 17, but for wall-normal translational particle velocity w_c. The error for the case with *zero* retraction is computed relative to $w_{c,r}$ obtained from the case *with* retraction.

estimated from Richardson extrapolation of the results for the present IBM, $w_{c,r}/V_b \approx 0.013$. The order of grid convergence seems to be roughly 1 for the original method, while it is about 2.4 for the present method. For resolutions of $D/\Delta x = 16$ till 24 the relative error of both methods is comparable and quite large compared to the relative error in v_c at the same resolutions. For $D/\Delta x > 24$ the error of the present IBM decreases much more rapidly than for the original IBM. At $D/\Delta x = 54$ the error of the present IBM amounts 3.7 %, a factor of about 7 smaller than the error of the original IBM.

Fig. 19 depicts the error in the lateral angular particle velocity ω_{cx} at $t = t_*$. Again, the error of the original IBM does not behave according to a power law of the grid resolution and therefore the exact value $\omega_{cx,r}h/V_b \approx -0.52$ has been estimated from the data of the present IBM. The order of grid convergence seems to be roughly 1 for the original method, while it is about 2.5 for the present IBM. At $D/\Delta x = 54$ the error is 23.6 % for the original method and 1.6 % for the present method, a difference by a factor of about 15.

Finally, Fig. 20 shows the maximum error in the norm of the Lagrangian velocity $\|\mathbf{U}_l\|$ at $t = t_*$ as function of the computational time step. As in Fig. 6, the time step was varied by varying the spatial grid resolution. For both the present IBM and the original IBM the order of convergence is the

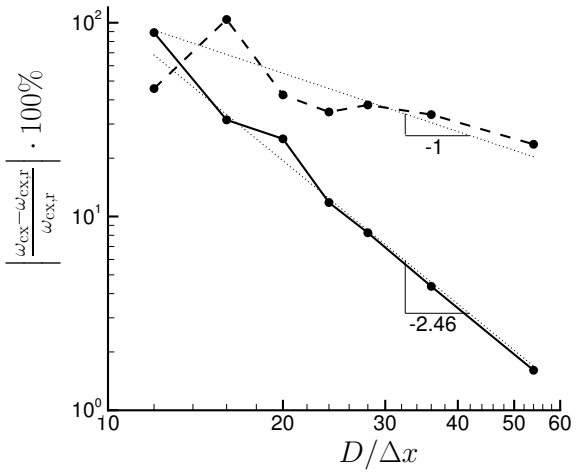

Figure 19: Idem as Fig. 17, but for the lateral angular particle velocity ω_{cx}. The error for the case with zero retraction is computed relative to $\omega_{cx,r}$ obtained from the case with retraction.

same with the error scaling with the computational time step to a power of approximately 0.4. The error in the present IBM with $N_s = 2$ is, however, significantly smaller compared to the original method corresponding to $N_s = 0$. It is remarked that the error fluctuates a little bit in time, presumably related to the change in orientation of the Lagrangian grid relative to the underlying Eulerian grid when the particle moves in space. This might explain the slight scatter in the data points.

6.3. Inertial migration of a sphere in tube Poiseuille flow

In section 4.3 it was mentioned that the numerical stability of the IBM has been improved by a direct account of the inertia of the fluid contained within the particles. This removes the singularity of the Newton-Euler equations for a mass density ratio ρ_p/ρ_f close to unity in the original IBM. In this section the improved stability will be demonstrated in a study of the migration of a sphere in tube Poiseuille flow for various mass density ratios in the range of $0.35 - 1.01$. But first the simulation of Poiseuille flow in a round tube in the absence of any sphere need to be discussed.

6.3.1. Tube Poiseuille flow

In order to simulate Poiseuille flow in a round tube on a Cartesian grid, a tube with radius R_t is placed in the center of a square channel. The present

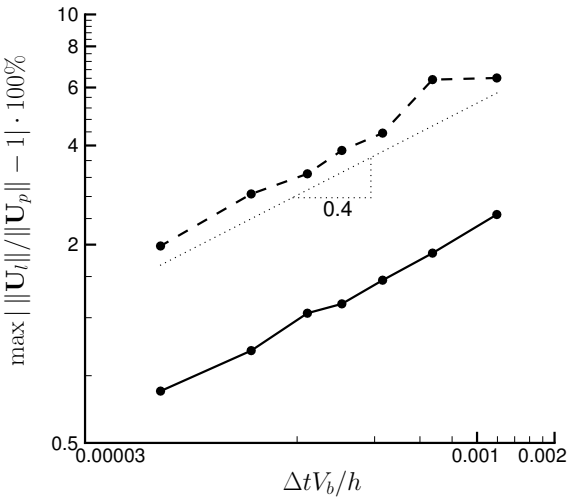

Figure 20: Maximum error in the norm of the Lagrangian velocity $\|\mathbf{U}_l\|$ as function of the computational time step at $t = t*$. The error is given in percentage of the norm of the instantaneous particle velocity $\|\mathbf{U}_p\|$. Lines: —, present IBM with $r_d = 0.3\Delta x$ and $N_s = 2$; - - -, original IBM corresponding to $r_d = 0$ and $N_s = 0$. Note that the time step was varied by varying the spatial grid resolution.

IBM is used to enforce the ns/np boundary conditions on the tube wall. Similar to the spheres in the previous sections, the tube wall is covered with a uniform 'Lagrangian tube grid'. In this case, the Lagrangian tube grid is not moving and thus $\mathbf{U}_l = \mathbf{0}\ \forall\ l$. The Lagrangian tube grid points are slightly retracted from the tube wall and located at radial position $R_t - r_{dt}$, where r_{dt} is referred to as the tube retraction distance. The spacing of the Lagrangian tube grid in the streamwise direction is the same as that of the Eulerian (Cartesian) grid. The streamwise coordinates of the Lagrangian tube grid points coincide with the streamwise coordinates of the Eulerian grid cell centers. Furthermore, the Lagrangian tube grid points are evenly distributed along the circumference of the tube. The number of Lagrangian tube grid points, N_{LT}, and their cell volume, ΔV_{lt}, are determined as follows:

$$N_{LT} = \left[\frac{(R_t - r_{dt} + \Delta x/2)^2 - (R_t - r_{dt} - \Delta x/2)^2}{\Delta x^3/(\pi L_t)}\right], \tag{18a}$$

$$\Delta V_{lt} = \frac{(R_t - r_{dt} + \Delta x/2)^2 - (R_t - r_{dt} - \Delta x/2)^2}{N_{LT}/(\pi L_t)}, \tag{18b}$$

where the square brackets in Eq. (18a) denote the nearest integer value of the enclosed expression and L_t is the streamwise extent of the tube (which is the same as the streamwise extent of the square channel).

First the optimal tube retraction distance is determined from simulations of tube Poiseuille flow in the absence of a sphere. In the simulations the number of iterations of the multidirect forcing scheme for the tube Lagrangian grid, N_{st}, is fixed at 2. The flow is driven by a streamwise pressure gradient equal to $dp_e/dz = -4\rho_f\nu_f^2/R_t^3$. According to the analytical solution for tube Poiseuille flow, this corresponds with a centerline velocity $W_{cl} = \nu_f/R_t$ and thus a Reynolds number $W_{cl}R_t/\nu_f = 1$. The centerline velocity obtained from a numerical simulation will somewhat deviate from this value due the approximation of the ns/np conditions at the tube wall in the IBM and numerical errors related to the used numerical scheme. The dimensions of the square channel are $2.5R_t$ x $2.5R_t$ x $1.5R_t$ in the x-, y- and z-direction, respectively, where the z-direction corresponds to the streamwise direction. The boundary conditions for the velocity and pressure at the solid side walls of the square channel are the ns/np conditions and the homogeneous Neumann condition, respectively. Periodic boundary conditions are imposed on both velocity and pressure in the streamwise direction. The initial condition for the velocity field is zero velocity. The simulations are run until the flow

reaches a steady state.

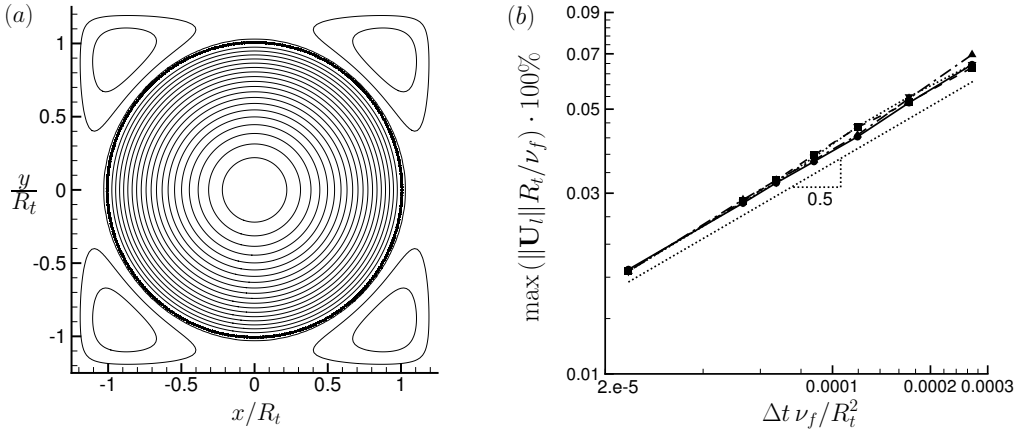

Figure 21: (a) Cross-section of Poiseuille flow in a round tube for $R_t/\Delta x = 54$, $r_{dt}/\Delta x = -0.35$ and $N_{st} = 2$. Contours represent isolines of the streamwise velocity with the contour interval equal to $0.05\,\nu_f/R_t$. The thick black line shows the location of the Lagrangian grid points on the wall of the tube. (b) Maximum error in the norm of the Lagrangian velocity $\|\mathbf{U}_l\|$ in percentage of ν_f/R_t and as function of the computational time step for various values of the tube retraction distance r_{dt} with $N_{st} = 2$ fixed. Note that the time step was varied by varying the spatial grid resolution. Lines: —, $r_{dt}/\Delta x = 0$; - - -, $r_{dt}/\Delta x = -0.3$; $\cdots$, $r_{dt}/\Delta x = -0.35$; – · – ·–, $r_{dt}/\Delta x = -0.5$.

Fig. 21.a shows a cross-section of the flow field at resolution $R_t/\Delta x = 54$ and a tube retraction distance $r_{dt}/\Delta x = -0.35$. Note that a negative tube retraction distance means that the radial position of the tube Lagrangian grid points is slightly *larger* than the tube radius. As expected, the contours of the streamwise velocity are circular. Since the streamwise pressure gradient is applied everywhere in the computational domain, also a flow exists in the space in between the tube wall and the side walls of the square channel; although this is not of particular interest, it is shown here for completeness.

Fig. 21.b shows the maximum error in the norm of the Lagrangian velocity $\|\mathbf{U}_l\|$ in percentage of ν_f/R_t (the centerline velocity expected from the analytical solution) as function of the computational time step for various values of the tube retraction distance. The error appears to be very small (less than 0.07% in all investigated cases) and scales with Δt to a power of approximately 0.5. The tube retraction distance has very little influence on the error.

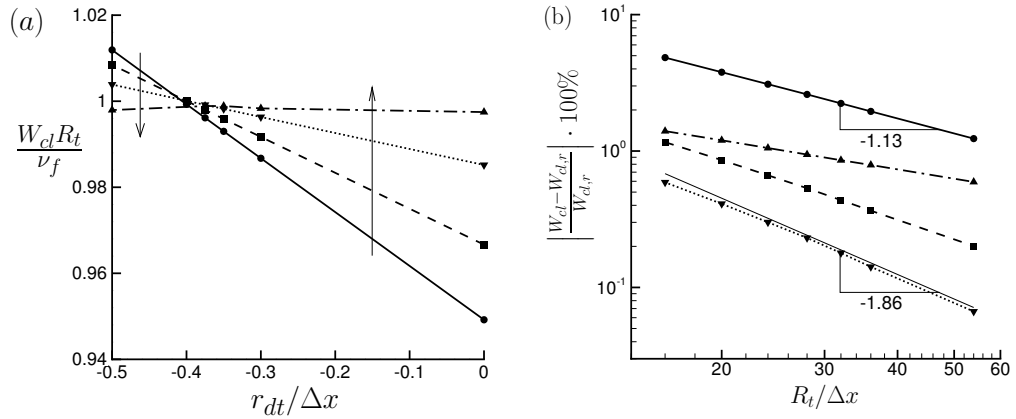

Figure 22: (a) Normalized centerline velocity $W_{cl}R_t/\nu_f$ of Poiseuille flow in a round tube as function of the normalized tube retraction distance $r_{dt}/\Delta x$ with $N_{st} = 2$ fixed. The lines lines represent different grid resolutions: —, $R_t/\Delta x = 16$; - - -, $R_t/\Delta x = 24$; $\cdots$, $R_t/\Delta x = 54$; $-\cdot-\cdot-$, $W_{cl,r}$ obtained from Richardson extrapolation. Arrow points in direction of increasing grid resolution. (b) Percentual error in centerline velocity as function of the grid resolution. The error is relative to the value of the centerline velocity $W_{cl,r}$ obtained from Richardson extrapolation using the data points of $R_t/\Delta x = 24$, 36 and 54. The lines show the effect of the tube retraction distance with $N_{st} = 2$ fixed: —, $r_{dt}/\Delta x = 0$; - - -, $r_{dt}/\Delta x = -0.3$; $\cdots$, $r_{dt}/\Delta x = -0.35$; $-\cdot-\cdot-$, $r_{dt}/\Delta x = -0.5$.

In figure 22.a the streamwise centerline velocity is plotted as function of retraction distance. The lines correspond to different spatial grid resolutions. The dash-dotted line shows the centerline velocity obtained from Richardson extrapolation of the simulations, $W_{cl,r}$. For the range of investigated tube retraction distances, $W_{cl,r}$ appears to vary very little with r_{dt}. At $r_{dt}/\Delta x = -0.35$, $W_{cl,r}R_t/\nu_t = 0.999$, an error of 0.1% compared to the value 1 of the analytical solution. From Fig. 22.a it is observed that for $r_{dt}/\Delta x \gtrsim -0.39$ the centerline velocity is underestimated in the simulations, while for $r_{dt}/\Delta x \lesssim -0.41$ it is overestimated. In the range of $-0.42 \lesssim r_{dt}/\Delta x \lesssim -0.4$, the centerline velocity does not vary monotonically with the grid resolution and is nearly independent of the grid resolution.

Fig. 22.b shows the percentual error in the centerline velocity as function of spatial grid resolution. The error is relative to the value obtained from Richardson extrapolation. The lines represent different values of the tube retraction distance. For $r_{dt} = 0$ the order of convergence is approximately 1, while near second-order convergence is obtained for $r_{dt}/\Delta x = -0.35$. At a

grid resolution of $R_t/\Delta x = 54$ the percentual error in the centerline velocity is about 1.23% for $r_{dt} = 0$, while only 0.067% for $r_{dt}/\Delta x = -0.35$, a difference of more than a factor 18.

Based on Figs 22.a and 22.b, it is concluded that a tube retraction distance of $r_{dt} = -0.35\Delta x$ is close to optimal for maximizing the accuracy of the IBM. Note that this is close to the retraction distance of $0.3\Delta x$ found for a sphere in section 6.1.2 (the minus sign for r_{dt} is a matter of definition; for both spheres and the tube wall, the Lagrangian grid points need to be displaced towards the interior of the solid particle/wall). This suggests once more that the optimal retraction distance does not depend strongly on the shape of arbitrarily shaped solid particles or solid boundaries.

6.3.2. Inertial migration of a neutrally-buoyant sphere

The inertial migration of a sphere in tube Poiseuille flow has been studied extensively in literature, in particular the case of a neutrally-buoyant sphere. When a neutrally-buoyant sphere is released in tube Poiseuille flow, it will slowly migrate towards an equilibrium radial position equal to about 0.6 times the tube radius. This effect has first been measured by Segré and Silberberg [27, 26] and is now known as the Segré-Silberberg or tubular pinch effect. More measurements have been reported by Jeffrey and Pearson [13], Karnis et al. [14] and Matas et al. [18]. The effect was analyzed theoretically by Schonberg and Hinch for plane Poiseuille flow [25], among others. In recent years there have been quite a number of numerical studies of the Segré-Silberberg effect in tube Poiseuille flow with different numerical methods [35, 20, 33, 36, 3].

In order to compare the present IBM with the mentioned numerical studies, a neutrally-buoyant sphere is considered with a diameter $D = 0.3R_t$. The tube Poiseuille flow is mimicked along the lines explained in the previous subsection. The flow is driven by a streamwise pressure gradient equal to $dp_e/dz = 200\rho_f\nu_f^2/R_t^3$, which according to the analytical solution for tube Poiseuille flow corresponds to a centerline velocity $W_{cl} = -50\nu_f/R_t$ and thus a Reynolds number $|W_{cl}|\, R_t/\nu_f = 50$. Note that the pressure gradient is directed in the positive z-direction and thus the flow is directed downwards. This is of importance when considering the case of a *buoyant* sphere in the next subsection. The tube is placed in a square channel with dimensions $2.1R_t \,\mathrm{x}\, 2.1R_t \,\mathrm{x}\, 4.05R_t$ in the x-, y- and z-direction, respectively, where the z-direction corresponds to the streamwise direction. The boundary conditions for the velocity and pressure are given in the previ-

ous subsection. The initial condition for the velocity field are $u = v = 0$ and $w(r)R_t/(50\nu_f) = (r/R_t)^2 - 1$, where r is the radial position relative to the centerline. The spatial grid resolution is fixed in the simulations at $D/\Delta x = 16$. The number of force iterations is fixed at $N_s = N_{st} = 2$ for both the sphere and tube Lagrangian grid. The retraction distances are $r_d/\Delta x = 0.3$ and $r_{dt}/\Delta x = -0.35$.

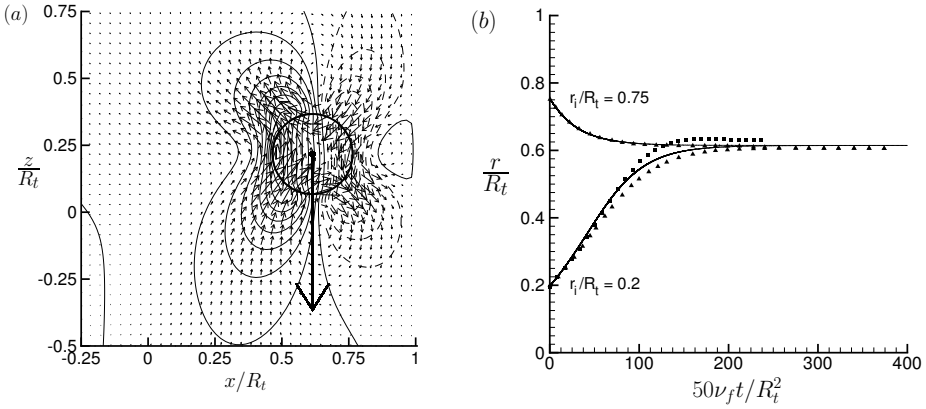

Figure 23: (a) Motion of a neutrally-buoyant sphere in tube Poiseuille flow at its final equilibrium radial position. Light vectors represent the deviations in the flow field around the sphere from the analytical solution for pure Poiseuille flow. The number of vectors is reduced by a factor 2 in each direction. Contours represent isolines of the streamwise velocity deviations with the contour interval equal to $0.5\nu_f/R_t$; dashed contours represent negative values. The thick black vector shows the streamwise velocity of the sphere centroid (in absolute frame of reference). (b) Radial position of sphere centroid r/R_t as function of time $50\nu_f t/R_t^2$. The two lines correspond to two different initial radial positions. The triangles represent data from Pan and Glowinski [20] and the squares data from Choi and Kim [3].

At $t = 0$ the sphere is released in the tube. Two different initial radial positions (r_i) have been simulated. In the first case $x_c/R_t = 0.2$ and $y_c = 0$, while in the second case $x_c/R_t = 0.75$ and $y_c = 0$, where x_c and y_c are the coordinates of the sphere centroid relative to the centerline. The initial radial positions are thus $r_i/R_t = 0.2$ and $r_i/R_t = 0.75$, respectively. The initial translational (w_c) and angular velocity (ω_{cy}) of the particle are determined from the background velocity profile and equal to the local values of w and $-(1/2)\partial w/\partial r = -50\nu_f r/R_t^3$ at the radial position of the sphere centroid, respectively.

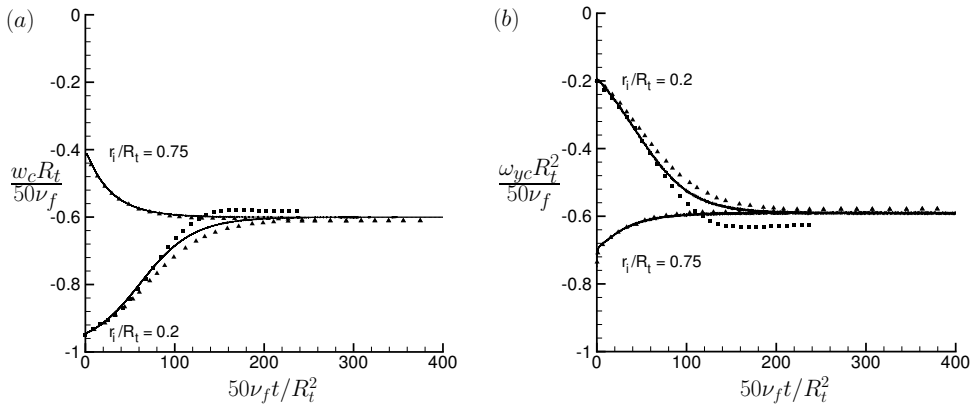

Figure 24: (a) Streamwise velocity of sphere centroid $w_c R_t/50\nu_f$ as function of time $50\nu_f t/R_t^2$. The two lines correspond to two different initial radial positions. The triangles represent data from Pan and Glowinski [20] and the squares data from Choi and Kim [3]. (b) Idem as for (a), but now showing the sphere angular velocity $\omega_{cy} R_t^2/(50\nu_f)$.

In both simulations the sphere reaches the same equilibrium radial position of $r/R_t = 0.6146$ after sufficiently long time. A snapshot of the flow field around the sphere at this equilibrium position is shown in Fig. 23.a, where the vectors represent the deviations in the flow field from pure Poiseuille flow. The velocity deviations nicely show the anti-clockwise rotation of the sphere. Note furthermore that the velocity deviations are confined to the vicinity of the sphere. This suggests that the sphere is not influenced by the limited streamwise extent of the tube and the imposed periodic boundary conditions in this direction. This was also confirmed by a test simulation in which the streamwise extent of the tube was increased by more than a factor 1.5; the results from that simulation were indistinguishable from the present results.

Figs 23.b, 24.a and 24.b depict the histories of the radial position of the sphere, the streamwise sphere velocity and the sphere angular velocity, respectively. For comparison with literature, data is included from Pan and Glowinski (DLM/FD method) [20] and Choi and Kim (ALE technique) [3]. The present results are in agreement with the literature data, in particular the data of Pan and Glowinski [20] who used a (velocity) mesh size in their simulations comparable to the present Eulerian grid spacing ($D/\Delta x = 15$ and 16, respectively).

The sphere migrates very slowly from its initial radial position towards

its equilibrium position at which the net radial force on the sphere is equal to zero. The maximum (absolute) radial velocity is achieved by the sphere with initial position $r_i/R_t = 0.2$, but this maximum value is still smaller than 0.415% of the undisturbed centerline velocity ($50\nu_f/R_t$). The sphere velocity at the equilibrium position appears slightly different from the undisturbed Poiseuille flow far upstream/downstream at the same radial position, -0.600 versus -0.622 (velocity normalized by $50\nu_f/R_t$). Also the sphere angular velocity at the equilibrium position appears to be slightly different from the undisturbed Poiseuille flow, -0.590 versus -0.615 (angular velocity normalized by $50\nu_f/R_t^2$). This indicates a small finite-size effect of the sphere on the sphere and flow dynamics.

6.3.3. Inertial migration of a buoyant sphere

The simulations of the previous subsection have been repeated for the case of buoyant spheres. The gravitational acceleration in the z-direction is set to $g_z = -2500\nu_f^2/(R_t^2 D)$. Recall that the flow is directed downwards. Five different mass density ratios are considered: $\rho_p/\rho_f = 1.01$, 1, 0.99, 0.675 and 0.35. Somewhat below a density ratio of 0.35 the numerical solution becomes unstable due to the fully explicit coupling of the Navier-Stokes and Newton-Euler equations as discussed in section 5. It is remarked that for the lowest two mass density ratios a significant wake region exists behind the sphere (not shown). This indicates that these simulations should actually be run for a much longer streamwise extent of the tube in order to eliminate significant effects from the (artificial) periodic boundary conditions in the streamwise direction. The purpose of this subsection is, however, to demonstrate the enhanced numerical stability of the present IBM rather than a detailed (quantitative) study of buoyancy effects on sphere migration.

Fig. 25 depicts the time history of the radial position of a buoyant sphere in tube Poiseuille flow for different values of the mass density ratio and the initial radial position. A slight deviation in the mass density ratio has a significant effect on the final radial equilibrium position of the sphere. Compared to a neutrally-buoyant sphere ($\rho_p/\rho_f = 1$), a slightly denser sphere migrates further towards the wall, while a slightly lighter sphere ends up closer to the centerline. This behavior is in line with the experiments of Jeffrey and Pearson [13]. Note that in the simulations the flow is directed downwards; for upward flow the opposite effect of the mass density ratio on the equilibrium radial position is expected. For $\rho_p/\rho_f = 0.35$ and $\rho_p/\rho_f = 0.675$ the final equilibrium position of the sphere is at the centerline of the

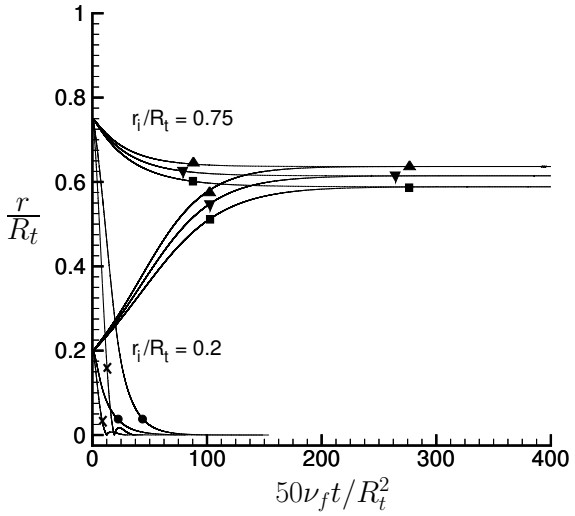

Figure 25: Radial position of a buoyant sphere in tube Poiseuille flow as function of time. The lines correspond to different values for the mass density ratio (ρ_p/ρ_f) and the initial radial position ($r_i/R_t = 0.2$ or 0.75). Symbols: deltas, $\rho_p/\rho_f = 1.01$; gradients, $\rho_p/\rho_f = 1$; squares, $\rho_p/\rho_f = 0.99$; circles, $\rho_p/\rho_f = 0.675$; crosses, $\rho_p/\rho_f = 0.35$.

tube, irrespective of the initial radial position. Furthermore, the lighter the sphere, the faster the sphere moves towards the centerline.

6.4. Slow normal approach between two equal spheres

Brenner [1] derived an analytical solution for the slow normal approach between two equal spheres in free space. This is therefore an interesting benchmark case involving more than just one particle for validation of the present IBM. In the simulations discussed below, the particle Reynolds number, $\|\mathbf{U}_p\|D/\nu_f$, is equal to 0.1. The surrounding flow field is thus in the Stokes regime and can be considered as quasi-stationary. This allows to keep the spheres fixed in space with a prescribed approach velocity. Note that in this benchmark case the Newton-Euler equations do not need to be solved. The dimensions of the computational domain are $5.5D \times 5.5D \times 9D$ in the x-, y- and z-direction, respectively. The spheres are positioned at $x_c/D = y_c/D = 2.75$ and $z_c/D = 4.375$ and 4.625, respectively. The gap width between the tip of the spheres is thus equal to $D/4$. The vertical centroid velocity of the spheres is prescribed at $w_c/\|\mathbf{U}_p\| = +1$ and -1, re-

spectively. The boundary conditions for the fluid phase are as follows: free-slip conditions for the boundary-parallel velocities in combination with zero pressure at the domain boundaries in the x- and y-directions and periodic boundary conditions for the velocity and pressure in the z-direction.

The simulations are initiated with a flow field at rest. With time the flow adapts itself to the velocities of the spheres and reaches a steady state. As noted before in section 6.1, at steady state part of the error in enforcing the ns/np condition at the surface of the spheres is eliminated as the *second prediction* is then equal to the *actual* fluid velocity.

Fig. 26 depicts a cross-section of the flow field at a resolution of $D/\Delta x = 16$, $N_s = 2$ force iterations and a retraction distance $r_d/\Delta x = 0.3$. The zero contour of the y-component of the fluid velocity nearly coincides with the surface of the spheres, demonstrating the success of the present IBM in enforcing the ns/np conditions at the surface of the spheres in this simulation.

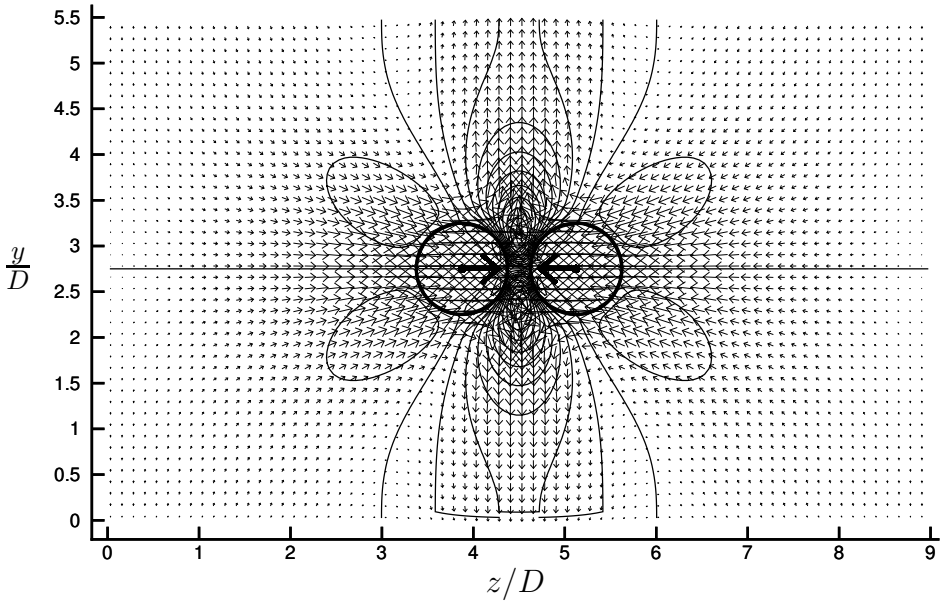

Figure 26: Cross-section of flow field at $x/D = 2.75$ for the slow normal approach between two equal spheres at a gap width of $D/4$. Simulation parameters: $D/\Delta x = 16$, $N_s = 2$ and $r_d/\Delta x = 0.3$. The number of vectors is reduced by a factor 2 in each direction. The thin lines are contours of the velocity component in the y-direction with a contour interval of $0.1\|\mathbf{U}_p\|$ and the thick arrows indicate the prescribed sphere velocities.

Similar to Fig.6, Fig. 27.a shows the maximum error in the norm of

the Lagrangian velocity $\|\mathbf{U}_l\|$ as function of the computational time step and for different values of the retraction distance and the number of force iterations. Note that the time step was varied by varying the spatial grid resolution. The error is smallest for $r_d/\Delta x = 0.3$ and $N_s = 2$, while largest for the original method with $r_d/\Delta x = 0$ and $N_s = 0$. Increasing the number of force iterations from 0 to 2 in the original method reduces the error in the ns/np condition strongly. Furthermore, the results indicate that r_d has relatively little effect on the error compared to N_s. For all three investigated combinations of r_d and N_s the order of convergence is roughly 0.5 in the computational time step.

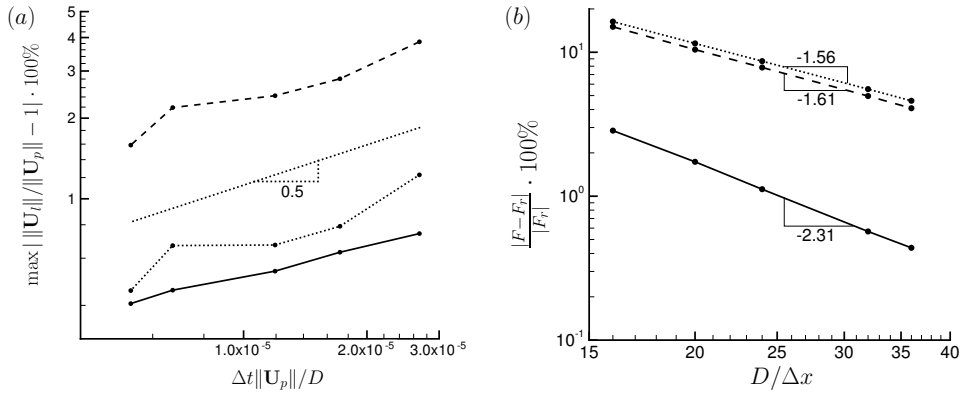

Figure 27: (a) Maximum relative error in the norm of the Lagrangian velocity $\|\mathbf{U}_l\|$ as function of the normalized computational time step $\Delta t\|\mathbf{U}_p\|/D$. The error is given in percentage of the norm of the approach velocity of the spheres $\|\mathbf{U}_p\|$. The lines with symbols correspond to: —, $r_d/\Delta x = 0.3$, $N_s = 2$; - - -, $r_d/\Delta x = 0$, $N_s = 0$; $\cdots$, $r_d/\Delta x = 0$, $N_s = 2$. (b) Percentual error in the norm of the force acting on the spheres as function of grid resolution $D/\Delta x$. The lines correspond to: —, $r_d/\Delta x = 0.3$, $N_s = 2$; - - -, $r_d/\Delta x = 0$, $N_s = 0$; $\cdots$, $r_d/\Delta x = 0$, $N_s = 2$.

Fig. 27.b depicts the percentual error in the force acting on the spheres as function of resolution, the retraction distance and number of force iterations. The error is computed relative to the force estimated from Richardson extrapolation of the data points at resolutions $D/\Delta x = 16$, 24 and 36. The present method with $r_d/\Delta x = 0.3$ and $N_s = 2$ has an order of grid convergence of about 2.3, while the original method with $r_d/\Delta x = 0$ has an order of grid convergence of about 1.6 and consequently significantly larger error. Increasing the number of force iterations from 0 to 2 in the original method

has a very small effect and even slightly increases the error in the force.

The values for the force obtained from Richardson extrapolation are in close agreement with the exact value given by Brenner's solution. According to the exact solution the force increases by a factor of 2.77 at a gap width of $D/4$ as compared to the force on a slowly moving sphere in free space (equal to $3\pi\mu_f D\|\mathbf{u}_c\|$). The present method with $r_d/\Delta x = 0.3$ and $N_s = 2$ predicts a multiplication factor of 2.75 (-0.87% off), while the original method with $r_d = 0$ and $N_s = 0$ or 2 predicts a multiplication factor of 2.80 (1.08% off).

6.5. Two equal spheres sedimenting in a closed container filled with a viscous fluid

In this section the case is considered of two equal spheres sedimenting in a closed container filled with a viscous fluid. This case has been studied before by Glowinski et al. [8] and Sharma and Patankar [28], among others. Following the latter authors, the specifications of the flow geometry are given in physical units. The dimensions of the container are 0.01 m x 0.04 m x 0.01 m in the x-, y- and z-direction, respectively. Gravity acts in the y-direction with the gravitational acceleration $g_y = -9.81$ m/s^2. The diameter of the two spheres is $D = 1.67 \cdot 10^{-3}$ m. The mass density of the spheres and the fluid is given by $\rho_p = 1140$ and $\rho_f = 1000$ kg/m^3, respectively, so the particle-fluid mass density ratio is equal to 1.14. The dynamic viscosity of the fluid is $\mu_f = 10^{-3}$ kg/(ms).

At $t = 0$ s the two spheres are placed in the container above each other at $y_c = 0.0316$ and $y_c = 0.035$ m, respectively. A slight offset is introduced in the horizontal position of the spheres to trigger the drafting-kissing-tumbling phenomenon [7, 5] discussed later. The upper sphere has horizontal position $x_c = z_c = 0.0202$ m, the lower sphere is placed at $x_c = z_c = 0.0198$ m. The fluid and the spheres are initially at rest. The boundary conditions for the fluid velocity and pressure at the solid walls are the ns/np conditions and the homogeneous Neumann condition, respectively.

Since a detailed investigation of particle collisions is not the focus of the present study, the simple collision model of Glowinski et al. [8] is adopted to accommodate collisions between the spheres in the simulations. Let $\mathbf{x}_{c,i}$ be the position vector of the centroid of a sphere labelled i. Define $\mathbf{d}_{ij} = \mathbf{x}_{c,j} - \mathbf{x}_{c,i}$ as the distance vector between the centroid of sphere i and the centroid of another sphere labelled j. The distance vector points from sphere i towards sphere j. The collision model becomes active when the distance $\|\mathbf{d}_{ij}\|$ between the centroids of the two spheres is smaller than $D + d_c$, where

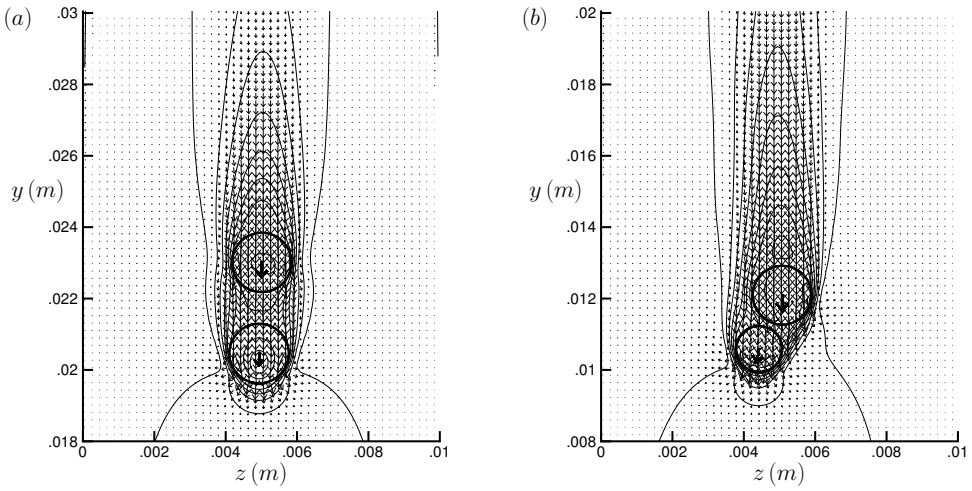

Figure 28: Two equal spheres sedimenting in a viscous fluid in a closed cavity. Cross-section at $x = 0.00495$ m of the flow field at two different times. Simulation parameters: $r_d/\Delta x = 0.3$, $N_s = 2$ and $D/\Delta x = 16$. The number of vectors is reduced by a factor 2 in each direction. Contours represent isolines of the vertical velocity with the contour interval equal to 0.01 m/s. (a) $t = 0.28646$ s. (b) $t = 0.44759$ s.

d_c is a threshold distance that has to be specified. In case the threshold distance is exceeded, the collision force on sphere i due to a collision with a sphere j is computed according to:

$$\mathbf{F}_{c,ij} \;=\; -\frac{\rho_p V_p \|\mathbf{g}\|}{\epsilon_c} \left(\frac{\|\mathbf{d}_{ij}\| - D - d_c}{d_c} \right)^2 \frac{\mathbf{d}_{ij}}{\|\mathbf{d}_{ij}\|}, \tag{19}$$

where ϵ_c is a nondimensional parameter that has to be specified. In the simulations presented below $\epsilon_c = 10^{-4}$ (same as in Glowinski et al. [8]) and $d_c/D = 1/16$ (corresponding to 1 Eulerian grid spacing at $D/\Delta x = 16$). The collision force is added to the right-hand side of Eq. (3a). The collision force is numerically integrated with the second-order accurate Crank-Nicolson scheme, see Eq. (16a). The collision force at the Runge-Kutta step q is iteratively determined as function of the sphere positions at the same step. The number of iterations is fixed at 15 and under-relaxation with a factor of 0.5 is used; using these settings the collision force, sphere centroid velocity and sphere centroid position quickly converge towards their equilibrium values with an exponential-like behavior as function of the number of iterations. The collision torque is neglected in the present study, so $\mathbf{T}_c = \mathbf{0}$

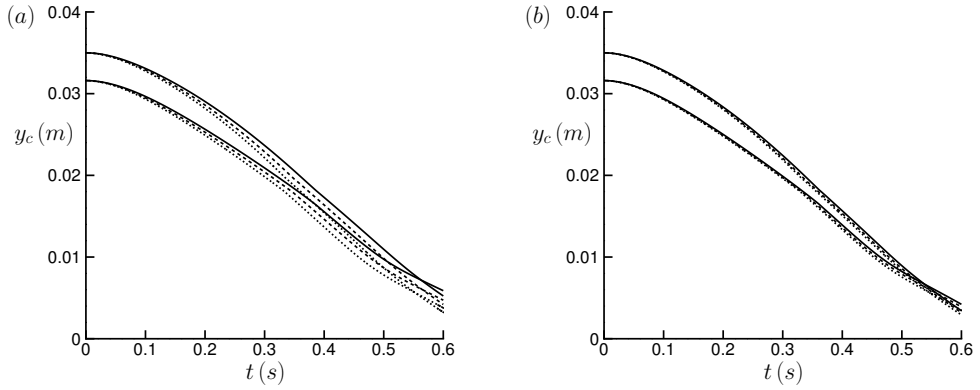

Figure 29: Two equal spheres sedimenting in a viscous fluid in a closed cavity. Vertical component of the sphere centroid positions as function of time. The lines represent different spatial grid resolutions: —, $D/\Delta x = 10$; - - -, $D/\Delta x = 16$; $\cdots$, $D/\Delta x = 54$. (a) Original method corresponding to $r_d = 0$ and $N_s = 0$. (b) Present IBM with $r_d/\Delta x = 0.3$ and $N_s = 2$.

in Eq. (16c).

After the spheres are released in the container at $t = 0$ s, they fall downwards under gravity. Since the upper sphere is shielded in the wake of the lower sphere, it experiences less drag from the surrounding fluid and falls faster. As a consequence the upper sphere drafts towards the lower sphere. Fig. 28.a shows a snapshot of the flow field during this *drafting* stage. After some time the sphere touch each other, referred to as *kissing*. The vertical alignment during this stage is unstable and the sphere move quickly around each other towards a more horizontal alignment. This is illustrated in Fig. 28.b. At the end of the kissing stage the spheres move apart from each other, referred to as the *tumbling* stage.

Fig. 29–31 show the time history of the vertical sphere centroid positions, the vertical sphere centroid velocities and the (shortest) distance between the two spheres, respectively. In the left panels the results are shown for the original IBM corresponding to $r_d = 0$ and $N_s = 0$. The mass density ratio in the present case is 1.14 for which the original IBM is actually unstable, so the original IBM was modified by a direct account of inertia of the fluid contained within the spheres as discussed in section 4.3. In the right panels the results are depicted for the present IBM with $r_d/\Delta x = 0.3$ and $N_s = 2$. The results for both the original and the present IBM are qualitatively similar,

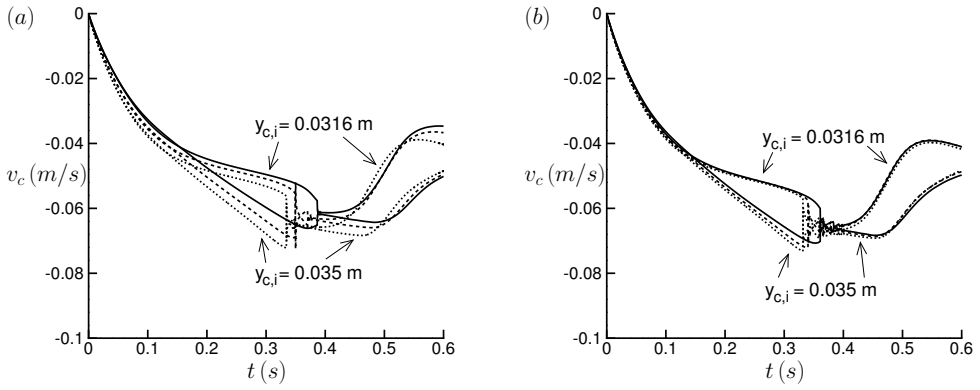

Figure 30: Idem as Fig. 29, but now showing the vertical component of the sphere centroid velocity as function of time. $y_{c,i}$ represents the initial vertical centroid position of the spheres at $t = 0$ s. (a) Original method. (b) Present IBM.

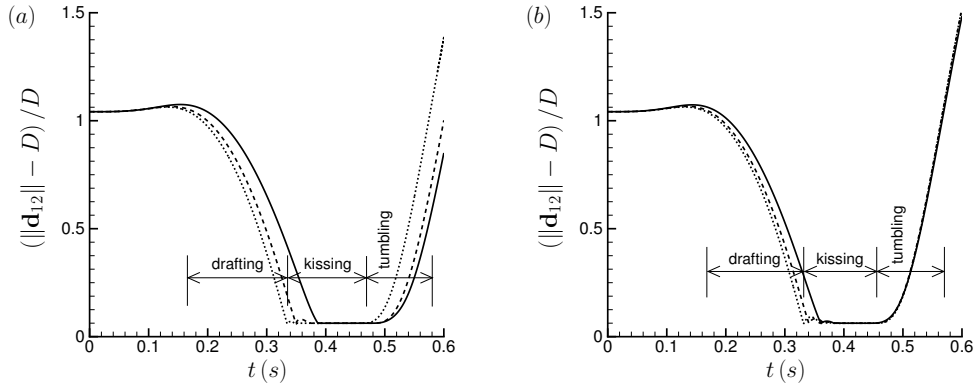

Figure 31: Idem as Fig. 29, but now showing the (shortest) distance between the two spheres as function of time. The drafting, kissing and tumbling stages are indicated for the simulation with the highest resolution ($D/\Delta x = 54$). (a) Original method. (b) Present IBM.

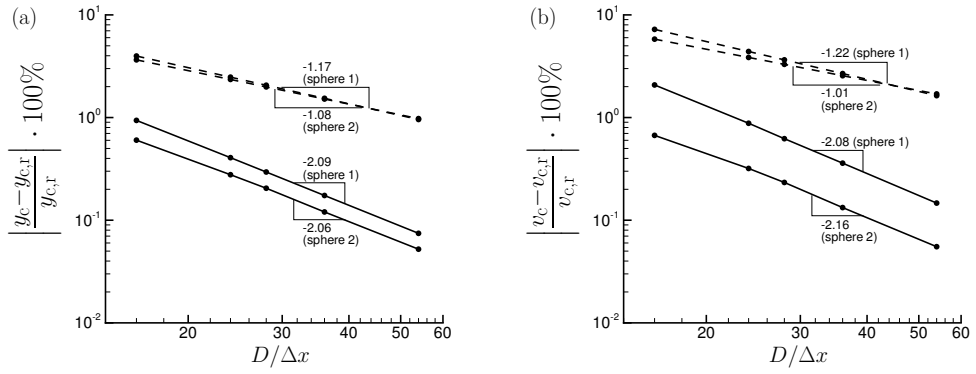

Figure 32: Two equal spheres sedimenting in a viscous fluid in a closed cavity. (a) Percentual error in the vertical position of the particle centroid y_c at $t = 0.28646$ s (at the end of the drafting stage, prior to kissing). The exact value y_{cr} is estimated from Richardson extrapolation using the data points of $D/\Delta x = 24$, 36 and 54. Dotted line with dots: original method corresponding to $r_d = 0$ and $N_s = 0$. Solid line with dots: present IBM with $r_d/\Delta x = 0.3$ and $N_s = 2$. (b) Idem as (a), but for the vertical translational particle velocity v_c.

but the quantitative differences are significant. Furthermore, the results of the original method are more susceptible for the spatial grid resolution as compared to the present IBM, hinting at a higher order of grid convergence for the present IBM. Fig. 29 shows that during the kissing and tumbling stage the upper particle overtakes the lower particle and reaches the bottom of the container first (the collision with the bottom wall is not shown). Fig. 30 nicely illustrates when kissing starts: at this point in time the vertical sphere velocities suddenly become equal. Fig. 31 indicates the duration of the kissing stage during which $\|\mathbf{d}_{12}\| - D \leq d_c = D/16$. It is remarked that the results of Glowinski et al. [8] compare best with the results from the original IBM at a comparable resolution of $D/\Delta x = 10$; differences between the results become significant at the end of the drafting stage and kissing occurs later in the present simulations (not shown).

Figs 32.a and 32.b show the percentual error in, respectively, the vertical centroid position and the vertical centroid velocity of the spheres as function of the spatial grid resolution at $t = 0.28646$ s. This point of time is taken at the end of the drafting stage, see Fig. 31. The exact values for the vertical centroid positions and vertical centroid velocities have been estimated from Richardson extrapolation of the simulation data. The results show that the

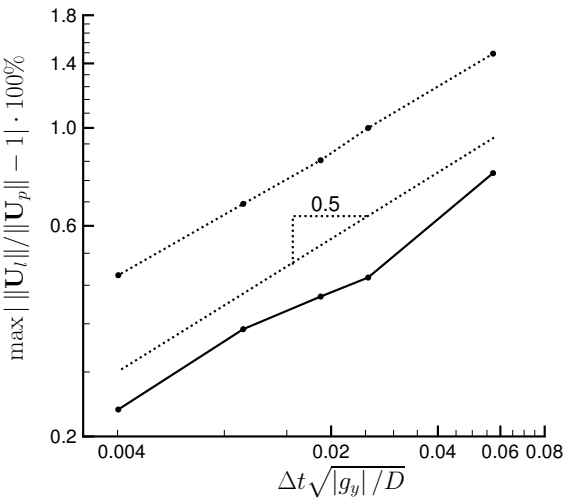

Figure 33: Two equal spheres sedimenting in a viscous fluid in a closed cavity. Maximum error in the norm of the Lagrangian velocity $\|\mathbf{U}_l\|$ as function of the computational time step. The error is given in percentage of the norm of the local target velocity, $\|\mathbf{U}_p\|$, at the Lagrangian grid point with index l. The computational time step was varied by varying the spatial grid resolution. Dotted line with dots: original method corresponding to $r_d = 0$ and $N_s = 0$. Solid line with dots: present IBM with $r_d/\Delta x = 0.3$ and $N_s = 2$.

present IBM is much more accurate than the original method; the order of grid convergence is roughly second order for the present IBM, while roughly first order for the original method. This result holds for both spheres.

Finally, Fig. 33 shows the maximum error in the norm of the Lagrangian velocity $\|\mathbf{U}_l\|$ as function of the computational time step (where both spheres were considered for determining the maximum). The error is given in percentage of the local target velocity at the specific Lagrangian grid point were the maximum was found. The error appears to scale with the computational step to a power of about 0.5.

7. Conclusions and discussion

The IBM of Uhlmann [31] has been modified to increase the numerical accuracy of the method and to improve its stability for a particle-fluid mass density ratio near unity. The numerical accuracy is increased by the implementation of a multidirect forcing scheme, in which the IBM force distribution around a particle is iteratively determined, and a slight inward

retraction of the Lagrangian grid towards the interior of a particle. The numerical stability of the method is improved by the direct account of the rate of change of linear and angular momentum of the fluid contained within a particle. It was demonstrated that the modified method is numerically stable for a particle-fluid mass density ratio as low as 0.35.

The numerical accuracy of the modified IBM is influenced by several errors:

1. The error associated with forcing the second prediction velocity instead of the actual fluid velocity. This results in an error in the fluid velocity at the Lagrangian grid points, which is expected to be $\|\mathbf{U}^{n+1} - \mathbf{U}^{**}\| = O(\Delta t^2)$. This error thus decreases with decreasing Δt. It is zero for a stationary flow for which the correction pressure is zero.
2. The error associated with a finite number of force iterations N_s to enforce the condition $\mathbf{U}^{**} \approx \mathbf{U}_p$ at the Lagrangian grid points. The higher the total number of force iterations N_s, the smaller the error in this condition. This error is smaller for smaller Δt, since a decrease in Δt effectively increases the number of force iterations per unit time. For the present numerical method, based on the three-step Runge-Kutta method of Wray [29] for the integration in time, the error scales with Δt to a power in the range of 0.4-0.6.
3. The error associated with the fully explicit coupling of the Navier-Stokes and Newton-Euler equations. In the IBM the second prediction velocity at Runge-Kutta step q is enforced to obey $\mathbf{U}^{**} \approx \mathbf{U}_p^{q-1}$. The error in enforcing the second prediction velocity to the particle velocity at Runge-Kutta step $q-1$ instead of q is expected to be of $O(\Delta t)$. As discussed in section 5, the fully explicit coupling causes numerical instabilities for a particle-fluid mass density ratio of $\rho_p/\rho_f \lesssim 0.3$. A remedy for this problem is to switch to a fully implicit coupling in which the condition $\mathbf{U}^{**} \approx \mathbf{U}_p^q$ is enforced iteratively, but this is computationally much more expensive than the present fully explicit coupling.
4. The error associated with the difference between the effective and actual particle diameter. The effective particle diameter depends on the retraction distance (radial position of the Lagrangian grid) and the width of regularized delta function used for the interpolation/spreading operations (width of the porous shell surrounding a particle). By choosing the retraction distance r_d proportional to a fraction of the width of the regularized delta function and thus proportional to Δx, this error

becomes grid dependent and decreases with decreasing Δx.

5. The error associated with a finite temporal and spatial grid resolution. Since Δt is determined from the stability constraints for the Runge-Kutta scheme, it is coupled to Δx. An increase in the spatial grid resolution thus also increases the temporal accuracy. Note that smaller Δx and Δt reduce all previously mentioned errors.

The modified IBM contains two parameters by which the accuracy of the method can be controlled: the number of force iterations N_s and the retraction distance r_d. The original method corresponds to $N_s = 0$ and $r_d = 0$. The effect of N_s and r_d on the numerical accuracy has been demonstrated for five different flows with either fixed or moving spheres, with just one or two interacting spheres and a particle Reynolds number of up to $O(10^2)$.

The simulation results indicate that N_s has a strong influence on the enforcement of the ns/np condition on the surface of a particle ($\mathbf{U}^{**} \approx \mathbf{U}_p$), while it has little effect on the effective particle diameter. By contrast, r_d has little influence on the enforcement of the ns/np condition, but a strong influence on the effective particle diameter. Furthermore, the results show that N_s has a weak effect on the order of grid convergence, while r_d has a strong effect on it. The choice of $N_s = 2$ and $r_d = 0.3\Delta x$ seems optimal for both fixed and freely moving spheres for increasing the accuracy and retaining the computational efficiency of the method. For this choice of r_d approximate second-order spatial accuracy is obtained for both the fluid and particle-related quantities, while the original method with $r_d = 0$ exhibits first-order accuracy.

The choice of $r_d = 0.3\Delta x$ reduces the number of Lagrangian grid points on the surface of a particle compared to the original method with $r_d = 0$. This somewhat reduces the amount of computational work required for the interpolation/spreading operations. On the other hand, the choice of $N_s = 2$ increases the work related to the interpolation/spreading operations by a factor 3 compared to the original method with $N_s = 0$. This factor is, however, not dependent of the grid resolution and the number of particles N_P, so that the work related to interpolation/spreading operations remains $O(N_P N_L)$ in the present IBM.

The overall conclusion is that the accuracy of the original IBM is raised from first to second order for $r_d = 0.3\Delta x$ and that the error in the ns/np condition is strongly reduced for $N_s = 2$, while maintaining the suppression of grid locking and the computational efficiency of the method.

The optimal value for r_d for obtaining second-order accuracy of the present IBM was initially expected to depend on the characteristic Reynolds number for the flow/particle and the shape of the particle. While the influence of the Reynolds number appears to be small within the range investigated, some effect has been found of the particle geometry. For a sphere the optimal value for $r_d/\Delta x$ is approximately 0.3, for a tube it is about 0.35 (absolute value) and for a cube it lies around 0.4. Simulations like the ones presented in this paper can be used to determine the optimal value for r_d for other particles. An optimal value in the same range of 0.3-0.4 is expected.

8. Acknowledgments

The author would like to thank Dr. W. Vermin from SARA (Stichting Academisch Rekencentrum Amsterdam) for support with the parallelization and performance optimization of the Fortran code (NCF Research Grant number NRG-2008-05).

Appendix A. Approximations of flow-induced force and torque acting on particle

In this appendix a derivation is given of Eqs. (8a) and (8b) for the translational and the angular velocity of a solid particle, respectively.

In order to relate the surface integral on the right-hand side of Eq. (3a) to the forcing of the IBM, let us integrate Eq. (5) over the volume occupied by a particle and rewrite the result in the following form:

$$\oint_{\partial V} \tau \cdot \mathbf{n} dA \;=\; -\rho_f \int_{V_p} \mathbf{f} dV + \rho_f \frac{d}{dt}\left(\int_{V_p} \mathbf{u} dV\right) + V_p \nabla p_e. \qquad \text{(A.1)}$$

Ideally, the flow outside the particle does not feel the difference between a true solid particle and the solidified fluid at the surface of the particle in the IBM. In that case the surface integral on the left-hand side of the above equation matches the flow-induced force experienced by a true solid particle.

If the forcing of the IBM is distributed only *within* the space occupied by the particle, then the first term on the right-hand side of Eq. (A.1) is related to the sum over *all* IBM forces according to:

$$-\rho_f \int_{V_p} \mathbf{f} dV \;=\; -\rho_f \sum_{ijk} \mathbf{f}_{ijk} \Delta x \Delta y \Delta z. \qquad \text{(A.2)}$$

Recall that for a solid sphere and the regularized delta function of Roma et al. [24], $\mathbf{f}$ is distributed over a spherical shell of thickness $3\Delta x$ and centered around the radial position of the Lagrangian grid points. When the Lagrangian grid points are located exactly on the surface of the sphere, as in the study by Uhlmann [31], the inner and outer radii of the shell are equal to $R - 3\Delta x/2$ and $R + 3\Delta x/2$, respectively. Thus, in this case about half of the distribution of $\mathbf{f}$ is located *outside* the sphere and Eq. (A.2) is invalid. As a remedy to this problem the Lagrangian grid can be retracted towards the interior of the particle, such that the force is entirely distributed within the particle. However, when the Lagrangian grid is retracted too much, the associated force distribution does not result in a good approximation of the desired ns/np condition on the surface of the particle and Eq. (A.1) is a bad approximation of the flow-induced force experienced by a true solid particle.

Motivated by the above considerations, let us integrate Eq. (5) over a sphere with volume V_{p2} and radius R_2 equal to the outer radius of the force distribution, i.e. $R_2 = R - r_d + 3\Delta x/2$ with r_d the retraction distance. We rewrite the result in the following form:

$$\oint_{\partial V_2} \tau \cdot \mathbf{n} dA - (V_{p2} - V_p)\nabla p_e - \rho_f \frac{d}{dt}\left(\int_{V_{p2}} \mathbf{u} dV - \int_{V_p} \mathbf{u} dV\right) = \\ - \rho_f \int_{V_{p2}} \mathbf{f} dV + \rho_f \frac{d}{dt}\left(\int_{V_p} \mathbf{u} dV\right) + V_p \nabla p_e. \tag{A.3}$$

We now postulate that the retraction distance can be chosen such that: (1) by good approximation the fluid satisfies the desired ns/np condition at radius R and (2) the right-hand side of Eq. (A.3) is a good approximation of the flow-induced force experienced by a true sphere of radius R. Replacing the surface integral in Eq. (3a) by the right-hand side of Eq. (A.3), we get:

$$\rho_p V_p \frac{d\mathbf{u}_c}{dt} \approx -\rho_f \int_{V_{p2}} \mathbf{f} dV + \rho_f \frac{d}{dt}\left(\int_{V_p} \mathbf{u} dV\right) \\ + (\rho_p - \rho_f) V_p \mathbf{g} + \mathbf{F}_c. \tag{A.4}$$

Using that the total forcing from the IBM is preserved in the interpolation and spreading operations [22], the first term on the right-hand side can be related to the sum over all IBM forces on the Lagrangian grid and Eq. (8a) is obtained.

Based on a similar analysis, it is postulated that the total flow-induced torque experienced by a true particle can be approximated by:

$$\oint_{\partial V_2} \mathbf{r} \times (\tau \cdot \mathbf{n}) dA - \rho_f \frac{d}{dt} \left(\int_{V_{p2}} \mathbf{r} \times \mathbf{u} dV - \int_{V_p} \mathbf{r} \times \mathbf{u} dV \right) = - \rho_f \int_{V_{p2}} \mathbf{r} \times \mathbf{f} dV + \rho_f \frac{d}{dt} \left(\int_{V_p} \mathbf{r} \times \mathbf{u} dV \right). \quad \text{(A.5)}$$

Replacing the surface integral in Eq. (3b) by the right-hand side of the above equation and using that the total torque is preserved in the interpolation and spreading operations [22], Eq. (8b) is obtained.

References

[1] H. Brenner, The slow motion of a sphere through a viscous fluid towards a plane surface, Chem. Eng. Sci. 16 (1961) 242–251.

[2] W.P. Breugem, B.J. Boersma, R.E. Uittenbogaard, The laminar boundary layer over a permeable wall, Transp. Porous Media 59 (2005) 267–300.

[3] C.R. Choi, C.N. Kim, Inertial migration and multiple equilibrium positions of a neutrally buoyant spherical particle in Poiseuille flow, Korean J. Chem. Eng. 27 (2010) 1076–1086.

[4] E.A. Fadlun, R. Verzicco, P. Orlandi, J. Mohd-Yusof, Combined immersed-boundary finite-difference methods for three-dimensional complex flow simulations, J. Comput. Phys. 161 (2000) 35–60.

[5] J. Feng, H.H. Hu, D.D. Joseph, Direct simulation of initial value problems for the motion of solid bodies in a newtonian fluid. Part 1. Sedimentation, J. Fluid Mech. 261 (1994) 95–134.

[6] J.H. Ferziger, M. Perić, Computational methods for fluid dynamics, Springer-Verlag, Berlin, 2002.

[7] A.F. Fortes, D.D. Joseph, T.S. Lundgren, Nonlinear mechanics of fluidization of beds of spherical particles, J. Fluid Mech. 177 (1987) 467–483.

[8] R. Glowinski, T.W. Pan, T.I. Hesla, D.D. Joseph, J. Périaux, A fictitious domain approach to the direct numerical simulation of incompressible viscous flow past moving rigid bodies: Application to particulate flow, J. Comp. Phys. 169 (2001) 363–426.

[9] F.H. Harlow, J.E. Welch, Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface, Phys. Fluids 8 (1965) 2182–2189.

[10] H. Hasimoto, On the periodic fundamental solutions of the stokes equations and their application to viscous flow past a cubic array of spheres, J. Fluid. Mech. 5 (1959) 317–328.

[11] K. Höfler, S. Schwarzer, Navier-Stokes simulation with constraint forces: finite-difference method for particle-laden flows and complex geometries, Phys. Rev. E 61 (2000) 7146–7160.

[12] H.H. Hu, N.A. Patankar, M.Y. Zhu, Direct numerical simulations of fluid-solid systems using the arbitrary Lagrangian-Eulerian technique, J. Comput. Phys. 169 (2001) 427–462.

[13] R.C. Jeffrey, J.R.A. Pearson, Particle motion in laminar vertical tube flow, J. Fluid Mech. 22 (1965) 721–735.

[14] A. Karnis, H.L. Goldsmith, S.G. Mason, The flow of suspensions through tubes: V. Inertial effects, Can. J. Chem. Engng 44 (1966) 181–193.

[15] T. Kempe, S. Schwarz, J. Fröhlich, Modelling of spheroidal particles in viscous flows, in: Proceedings of the Academy Colloquium Immersed Boundary Methods: Current Status and Future Research Directions (KNAW, Amsterdam, The Netherlands, 15–17 June 2009).

[16] S. Kriebitzsch, M. Van der Hoef, H. Kuipers, Direct numerical simulations of gas-particle flows using an immersed boundary method, in: Proceedings of the Academy Colloquium Immersed Boundary Methods: Current Status and Future Research Directions (KNAW, Amsterdam, The Netherlands, 15–17 June 2009).

[17] K. Luo, Z. Wang, J. Fan, K. Cen, Full-scale solutions to particle-laden flows: Multidirect forcing and immersed boundary method, Phys. Rev. E 76 (2007).

[18] J.P. Matas, J.F. Morris, É. Guazelli, Inertial migration of rigid spherical particles in Poiseuille flow, J. Fluid Mech. 515 (2004) 171–195.

[19] R. Mittal, G. Iaccarino, Immersed boundary methods, Ann. Rev. Fluid Mech. 37 (2005) 239–261.

[20] T.W. Pan, R. Glowinski, Direct simulation of the motion of neutrally buoyant balls in a three-dimensional Poiseuille flow, Comptes Rendus Mécanique 333 (2005) 884–895.

[21] C.S. Peskin, Flow patterns around heart valves: A numerical method, J. Comput. Phys. 10 (1972) 252–271.

[22] C.S. Peskin, The immersed boundary method, Acta Numer. 11 (2002) 479–517.

[23] M. Pourquie, W.P. Breugem, B.J. Boersma, Some issues related to the use of immersed boundary methods to represent square obstacles, Int. J. Mult. Comp. Eng. 7 (2009) 509–522.

[24] A.M. Roma, C.S. Peskin, M.J. Berger, An adaptive version of the immersed boundary method, J. Comput. Phys. 153 (1999) 509–534.

[25] J.A. Schonberg, E.J. Hinch, Inertial migration of a sphere in Poiseuille flow, J. Fluid Mech. 203 (1989) 517–524.

[26] G. Segré, A. Silberberg, Behaviour of macroscopic rigid spheres in Poiseuille flow. Part 2. Experimental results and interpretation, J. Fluid Mech. 14 (1961) 136–157.

[27] G. Segré, A. Silberberg, Radial particle displacements in Poiseuille flow of suspensions, Nature 189 (1961) 209–210.

[28] N. Sharma, N.A. Patankar, A fast computation technique for the direct numerical simulation of rigid particulate flows, J. Comp. Phys. 205 (2005) 439–457.

[29] P.R. Spalart, R.D. Moser, M.M. Rogers, Spectral methods for the Navier-Stokes equations with one infinite and two periodic directions, J. Comput. Phys. 96 (1991) 297–324.

[30] M. Uhlmann, Simulation of particulate flows on multi-processor machines with distributed memory, Technical Report No. 1039, CIEMAT, Madrid (Spain), ISSN 1135-9420, 2003.

[31] M. Uhlmann, An immersed boundary method with direct forcing for simulation of particulate flows, J. Comput. Phys. 209 (2005) 448–476.

[32] M. Uhlmann, Experience with DNS of particulate flow using a variant of the immersed boundary method, in: P. Wesseling, E. Oñate, J. Périaux (Eds.), Proceedings of the European Conference on Computational Fluid Dynamics (ECCOMAS CFD), TU Delft, The Netherlands, 2006.

[33] C. Veeramani, P.D. Minev, K. Nandakumar, A fictitious domain formulation for flows with rigid particles: A non-lagrange multiplier version, J. Comp. Phys. 224 (2007) 867–879.

[34] P. Wesseling, Principles of computational fluid dynamics, Springer series in computational mathematics, Vol. 29, Springer-Verlag, Berlin, 2001.

[35] B.H. Yang, J. Wang, D.D. Joseph, H.H. Hu, T.W. Pan, R. Glowinski, Migration of a sphere in tube flow, J. Fluid Mech. 540 (2005) 109–131.

[36] Z. Yu, X. Shao, A direct-forcing fictitious domain method for particulate flows, J. Comput. Phys. 227 (2007) 292–314.