A Variable-Density Fictitious Domain Method for Particulate Flows with Broad Range of Particle-Fluid Density Ratios

Sourabh V. Apte[1a], and Justin R. Finn[a]

[a]School of Mechanical, Industrial, and Manufacturing Engineeing, Oregon State University, Corvallis, OR 97331

Abstract

A numerical scheme for fully resolved simulation of fluid-particle systems with freely moving rigid particles is developed. The approach is based on a fictitious domain method wherein the entire fluid-particle domain is assumed to be an incompressible fluid but with variable density. The flow inside the particle domain is constrained to be a rigid body motion using an additional rigidity constraint in a fractional step scheme. The rigidity constraint force is obtained based on the fast computation technique proposed by Sharma & Patankar [1]. The particle is assumed to be made up of material points moving on a fixed background mesh where the fluid flow equations are solved. The basic finite-volume solver is based on a co-located grid incompressible but variable density flow. The incompressibility constraint is imposed by solving a variable-coefficient pressure equation. Use of density-weighted reconstruction of the pressure gradients was found to give a stable scheme for high density ratio fluid-particle systems. Various verification and validation test cases on fixed and freely moving particles are performed to show that the numerical approach is accurate and stable for a wide range ($10^{-3} - 10^{6}$) of particle-fluid density ratios.

Key words: Fully resolved simulations, fictitious domain method, particulate flows, high-density ratio, particle-vortex interactions.

1. Introduction

Fully resolved simulations (FRS) of fluid-particle systems, wherein all scales associated with the fluid and particle dynamics are completely captured from first principles, are of great importance for understanding disperse particulate flows with applications in environmental engineering, biological flows, chemical reactors, and energy conversion systems; for example, sediment transport, aeolian transport, red blood cells, coal-particle combustors, bubbly flows in fluidized beds, catalytic reactors, among others. Many of these applications involve complex configurations and unsteady, often turbulent flows and their fundamental understanding is of critical importance. Such direct numerical simulation techniques are useful to obtain detailed data that can be used to develop subgrid and reduced order models used in other approaches for particulate flows such as Euler-Lagrange discrete element modeling, wherein the disperse particle dynamics is modeled through closure laws for drag, lift, added mass and other forces exerted by the fluid.

Several numerical schemes have been developed for fully resolved simulations of *freely moving, rigid particles* in a fluid flow. These can be categorized as (i) body-fitted, (ii) mesh-free, and (iii) fixed-grid methods. The body-fitted grid approach, such as boundary element [2] and arbitrary Lagrangian-Eulerian (ALE)-based finite-element approach on unstructured grids [3] method, use moving grids that conform to the shape of the immersed particles. Moving mesh algorithm based on space-time finite-element approach was also developed by Johnson and Tezduyar [4] to calculate falling particles in a tube. Such approaches, although provide an accurate solution at the fluid-particle interface, suffer from the complexity of the moving mesh and regeneration algorithms. Use of these techniques in three-dimensions significantly increase the computational cost and memory requirements. Smoothed particle hydrodynamics (SPH) is a *mesh-free* technique commonly used for multiphase flows with free-surfaces [5]. Fixed grid approaches, such as distributed Lagrange multiplier and fictitious-domain methods, immersed boundary method, lattice Boltzmann method, among others, are most popular for such simulations owing to their simplicity in computing motion of arbitrary shaped objects.

[1]Corresponding Author: sva@engr.orst.edu, Phone: 541-737-7335, Fax: 541-737-2600

Considerable work has been done on fully resolved simulations of freely moving particles in fluid flows on fixed grids. For example, distributed Lagrange multiplier/fictious domain (DLM) based methods ([6]) and Immersed Boundary method (IBM by [7, 8, 9, 10, 11]) have been developed and shown to be very effective in computing fluid-particle systems and fluid-structure interaction problems. Lattice Boltzmann method (LBM by [12]) has been developed and effectively used for simulations of rigid as well as deforming particles. Combination of the DLM, direct forcing based IBM, and Lattice-Boltzmann method (termed as *Proteus*) was recently developed [13]. A second-order accurate fixed grid method (PHYSALIS) has also been proposed [14], which gives good solutions for *spherical* particles by using local spectral representations of the solution near a spherical boundary.

The Immersed Boundary Method has traditionally been used for fluid-structure interaction problems with the motion of the immersed object specified (stationary, forced rigid motion, or elastically deforming objects). For such applications, two different implementations are typically used involving 'direct continuous forcing' wherein a continous forcing function around the particle boundary is added to the Navier Stokes equations [7, 15, 16] or 'discrete forcing' wherein forcing is either explicitly or implicitly applied to the discrete equations [17, 18, 19, 20, 10, 11]. The former is a straightforward approach that can be implemented in any Navier-Stokes solver with relative ease, however, diffuses the interface boundary proportional to the grid spacing owing to interpolation functions. The latter allows precise satisfaction of the boundary condition at the immersed surface maintaing a sharp interface representation, however, its implementation for arbitrary shaped objects can become fairly involved. Recently, Kim and Choi [10] developed a new immersed boundary method using the conservative form of Navier-Stokes and continuity equations in the non-inertial frame of reference and applied to fluid-structure interactions problems with the motion of the immersed objects specified (forced) or predicted as for freely moving rigid particles. With sufficient grid resolutions, both approaches have been shown to provide grid-convergent and accurate results.

For freely moving particle-laden flows, use of relatively coarse grids near the interface is necessary especially if the approach is used to study large number of particles (on the order of 1000) in complex flows. In such flows, use of grid resolution with more than 20-25 grid points per particle can become prohibitively expensive. Direct forcing techniques may result in large oscillations in the forces exerted by the fluid on the particle if the particle moves in such a way that the local stencil near the boundary changes abruptly. This, although not an issue for specified motion fluid-structure interaction problems, can cause problems to freely moving particulate flow problems. It is especially important for simulations with relatively coarse resolution of the interface between the particle and the fluid. Continuous forcing immersed boundary approaches on the other hand do not seem to show such an issue as the forces are regularized prior to discretization [9]. The relative ease of implementation for continuous forcing methods makes them attractive for freely moving large number of disperse particles in complex turbulent flows. However, the numerical approach by Uhlmann [9] has been found to be stable only for particle-to-fluid density ratio ($\rho_P/\rho_F > 1$) and has only been used for density ratios up to 10, whereas for particle-air systems the density ratios can easily range on the order of 10^3. For lighter than fluid particles or neutrally buoyant particles, the scheme has been found to become unstable. Recently, Kempe & Frohlich [21] suggested modifications to the approach that increased the range of applicability of the method to lighter than fluid particles and tested the scheme for $0.3 < \rho_P/\rho_F < 10$.

Taira & Colonius [22] proposed a new implementation of the immersed boundary method to achieve second-order accuracy. They compared IBM with fictitious-domain based methods to point out subtle differences when the immersed objects are constrained to undergo specified motion. In the fictititious domain/DLM method (see [6, 23]), the entire fluid–particle domain is assumed to be a fluid and the flow inside the entire particle domain is constrained to be a rigid-body motion through rigidity constraint in terms of a stress or a force. The rigidity constraint force is applied over the entire particle domain as opposed to the continuous forcing immersed boundary method of Uhlmann [9] where the forcing function is present very close to the interface, giving rise to fluid-like flowfield inside the particle region. Sharma & Patankar [1] proposed a fast technique to obtain the rigidity constraint force that eliminated the need for an iterative procedure to solve for the rigid body motion in laminar flows. Recently, Veeramani *et al.* [24] proposed a similar approach in the context of finite-element methods and used constant fluid density even within the particle domain. Apte *et al.* [25] further developed the finite-volume based fictitious domain approach by [1] to large number of particles in complex turbulent flows on co-located grids and improved the temporal and spatial accuracy. Their approach [25] uses the true local density at a control volume, equal to the fluid in the fluid region and equal to the particle in the particle region, and constant coefficient Poisson solvers based on multigrid approaches for fast convergence. This approach does not suffer the stability issue as in Uhlmann [9] and has been used for particle-to-fluid density ratios over the range of 0.1-20.

All of the above approaches have only been applied to particle-fluid systems with relatively low range of density ratios between the two-phases ($\mathcal{O}[10^{-1} - 10]$). Large density ratios are common in many practical applications involving complex flows; for example coal particles in a oxycoal boiler, aeolian particle transport, aerosol transport, microfluidics, among others. Sharp gradients in density across the fluid-particle interface in turbulent flows, for example in *gas-solid* systems such as aeolian transport, chemical reactors ($\mathcal{O}[10^3]$) or lighter than fluid solid-liquid or bubbly flow systems, can cause numerical 'ringing' of the solution and lead to numerical instabilities when using the fictitious domain approach with fast computation of the rigidity constraint ([1, 25]). In the present work, we extend this numerical approach to account for fluid-particle systems over a broad range of density ratios of $\mathcal{O}[10^{-3} - 10^6]$. A density-weighted reconstruction procedure for velocity and pressure gradient fields is used to obtain stable results. The stability and accuracy of the approach is investigated rigorously indicating good properties over a wide range of density ratios.

The paper is arranged as follows. A mathematical formulation of the fictitious domain scheme is first described. Numerical issues with the original formulation for high density ratios and potential remedies are discussed. A new stable approach for broad range of fluid-particle density ratios is implemented in a co-located grid finite volume method. The approach is first validated for basic test cases to show good predictive capability. Namely, flows over a fixed cylinder and sphere are first investigated to quantify the accuracy of the scheme. Next, freely falling/rising spherical particles at different Reynolds numbers are considered and compared with available experimental data at relatively low fluid-particle density ratios. The density ratios are varied over $10^{-3} - 10^6$ for the freely rising/falling particle to show stable and accurate solution. Finally, interactions of a lighter than fluid sphere with a stationary Gaussian vortex is simulated to show the capability of the approach to study particle-vortex interactions.

2. Mathematical Formulation

Let Γ be the computational domain which includes both the fluid ($\Gamma_F(t)$) and the particle ($\Gamma_P(t)$) domains. Let the fluid boundary not shared with the particle be denoted by $\mathcal{B}$ and have a Dirichlet condition (generalization of boundary conditions is possible). For simplicity, let there be a single rigid object in the domain and the body force be assumed constant so that there is no net torque acting on the object. The basis of fictitious-domain based approach is to extend the Navier-Stokes equations for fluid motion over the entire domain Γ inclusive of immersed object. The natural choice is to assume that the immersed object region is filled with a Newtonian *fluid* of density equal to the object density (ρ_P) and some fluid viscosity (μ_F). Both the real and fictitious fluid regions will be assumed as incompressible and thus incompressibility constraint applies over the entire region. In the numerical approach presented by [1, 25], the particle region is identified by an indicator (color) function Θ which has unit value inside the particle region and vanishes in the fluid region. Owing to finite number of grid cells, the boundary region of the particle typically is smeared with $0 \leq \Theta \leq 1$. The density field over the entire domain is then given as,

$$\rho = \rho_P \Theta + \rho_F \left(1 - \Theta\right). \tag{1}$$

The indicator function moves with the particle resulting in

$$\frac{D\Theta}{Dt} = 0, \tag{2}$$

where D/Dt is the material derivative. The fluid velocity field is constrained by the conservation of mass over the entire domain given as

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0. \tag{3}$$

The conservation of mass together with the indicator function advection implies that for an incompressible fluid,

$$\nabla \cdot \mathbf{u} = 0, \tag{4}$$

over the entire domain.

The momentum equation for fluid motion applicable in the entire domain Γ in the conservative form is then given by:

$$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u}\mathbf{u}) \quad = \quad -\nabla p + \nabla \cdot \left(\mu_F \left(\nabla \mathbf{u} + (\nabla \mathbf{u})^T \right) \right) + \rho \mathbf{g} + \tilde{\mathbf{f}}, \tag{5}$$

where ρ is the density field, $\mathbf{u}$ the velocity vector, p the pressure, μ_F the fluid viscosity, $\mathbf{g}$ the gravitational acceleration, and $\tilde{\mathbf{f}}$ is an additional body force that enforces rigid body motion within the immersed object region Γ_P.

With the above variable density field formulation, sharp changes in density over a single grid cell can lead to numerical instabilities when the momentum and continuity equations are solved in the above conservative form. This was shown to be a problem in volume of fluid formulations for two immiscible fluids by Rudman [26]. This problem can be remedied by performing consistent flux constructions for mass and momentum fluxes at the control volume faces [26, 27, 28]. Inconsistencies in flux calculations for mass and momentum leads to incorrect accelerations of fluids near interfaces leading to numerical instability at high density ratios. Alternatively, one option is to keep constant density (equal to the fluid density) over the entire domain, including the particle domain [24]. This works well for small density ratios, however, gives numerical instabilities for large density ratios owing to large variations in the explicit rigid body force, especially when the rigid body is accelerating or decelerating.

An alternative approach, that is commonly followed in interface tracking schemes based on level set methods [29, 30], is to solve the above equations in a rearranged non-conservative form, wherein computation of density jumps across cell faces are not required especially for co-located grid formulations. However, level set methods suffer from the loss of mass owing to the non-conservative advection of the signed-distance function especially for deforming interfaces. For the present work on rigid body motion, the interface between the fluid and the particle is advanced by motion of Lagrangian marker points [25] and the surface area of the interface between the fluid and particles remains constant over time as interface deformation is absent. Since the rigid particles are advanced in a purely Lagrangian frame their mass is conserved discretely in a numerical formulation. When cast in the non-conservative form, and making use of the fact that the fluid velocity is divergence free over the entire domain, the momentum equation can be written as,

$$\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot (\mathbf{u}\mathbf{u}) \quad = \quad -\frac{1}{\rho}\nabla p + \frac{1}{\rho}\nabla \cdot \left(\mu_F \left(\nabla \mathbf{u} + (\nabla \mathbf{u})^T\right)\right) + \mathbf{g} + \frac{1}{\rho}\tilde{\mathbf{f}}. \tag{6}$$

In the present work, we solve the momentum equation in the above form together with the incompressibility constraint $\nabla \cdot \mathbf{u} = 0$ over the entire domain. The density field ρ varies depending on whether the control volume is in the fluid or particle domain. In order to enforce that the material inside the immersed object moves in a rigid fashion, a rigidity constraint force is required that leads to a non-zero forcing function $\tilde{\mathbf{f}}$. For simplicity, the rigidity constraint force per unit density is labeled as $\mathbf{f}$ and is used in enforcing the rigid body motion in the formulation,

$$\mathbf{f} = \frac{1}{\rho}\tilde{\mathbf{f}}. \tag{7}$$

The fractional step scheme can be summarized as follows:

1. In this first step, the rigidity constraint force $\tilde{\mathbf{f}}$ in equation 6 is set to zero and the equation together with the incompressibility constraint is solved by standard fractional-step schemes over the entire domain. A *variable coefficient* pressure Poisson equation is derived and used to project the velocity field onto an incompressible solution. A *density weighted pressure reconstruction* procedure is used to obtain the pressure gradients at the center of the control volumes which are then used to correct the velocity field. The obtained velocity field is denoted as $\mathbf{u}^{n+1}$ inside the fluid domain and $\hat{\mathbf{u}}$ inside the object.
2. The velocity field for a freely moving object is obtained in a second step by projecting the flow field onto a rigid body motion. Inside the object:
$$\left(\frac{\mathbf{u}^{n+1} - \hat{\mathbf{u}}}{\Delta t}\right) = \frac{1}{\rho}\tilde{\mathbf{f}} = \mathbf{f}. \tag{8}$$
To solve for $\mathbf{u}^{n+1}$ inside the particle region we require $\mathbf{f}$. This is obtained by first finding the rigid body motion inside the particle region. The velocity field in the particle domain involves only translation and angular velocities. Thus $\hat{\mathbf{u}}$ is split into a rigid body motion ($\mathbf{u}^{RBM} = \mathbf{U} + \Omega \times \mathbf{r}$) and residual non-rigid motion ($\mathbf{u}'$). The translational and rotational components of the rigid body motion are obtained by conserving the linear and angular momenta and are given as:
$$M_P \mathbf{U} \quad = \quad \int_{\Gamma_P} \rho_P \hat{\mathbf{u}} d\mathbf{x}; \tag{9}$$

$$\mathcal{I}_P\Omega = \int_{\Gamma_P} \mathbf{r} \times \rho_P \hat{\mathbf{u}} d\mathbf{x}, \tag{10}$$

where M_P is the mass of the particle and $\mathcal{I}_P = \int_{\Gamma_P} \rho_P[(\mathbf{r}\cdot\mathbf{r})\mathbf{I} - \mathbf{r}\otimes\mathbf{r}]d\mathbf{x}$ is the moment of inertia tensor. Knowing $\mathbf{U}$ and Ω for each particle, the rigid body motion inside the particle region $\mathbf{u}^{RBM}$ can be calculated.

3. The rigidity constraint force is then simply obtained as $\tilde{\mathbf{f}} = \rho(\mathbf{u}^{RBM} - \hat{\mathbf{u}})/\Delta t$. This sets $\mathbf{u}^{n+1} = \mathbf{u}^{RBM}$ in the particle domain. Note that the rigidity constraint is non-zero only inside the particle domain and zero everywhere else. This constraint is then imposed in a third fractional step.

The key advantage of the above formulation is that the projection step only involves straightforward integrations (simple discrete summations) in the particle domain. The actual implementation of the approach and the semi-descritization are described below.

3. Numerical Approach

The preceding mathematical formulation is implemented in a co-located, structured grid, three-dimensional flow solver based on a fractional-step scheme developed by Apte *et al.* [25]. Accordingly, in the present work the fluid-particle system is solved by a *three-stage fractional step scheme*. First the momentum equations are solved. Instead of dropping the pressure gradient and rigidity constraint forces, an old time-level estimate is used for both, thus allowing solution of the full momentum equation in the first step. The incompressibility constraint is then imposed by solving a *variable-coefficient Poisson equation* for pressure in the second step. Finally, the rigid body motion is enforced by constraining the flow inside the immersed object to translational and rotational motion. An inner iteration is generally used to minimize the splitting error in the fractional time-stepping scheme and to improve robustness, especially when using large time-steps. However, as shown later, for the present application of freely moving particulate flows, the time-steps used are generally small (CFL $\leq$ 1) and a single iteration is sufficient. The main steps of the numerical approach are given below.

3.1. Immersed Object Representation

In the numerical implementation, small material volumes of cubic shape are created that completely occupy the immersed object. Each material volume is assigned the properties of the immersed object (e.g. density etc.). The shape of the object can be reconstructed from these material volumes by computing an indicator or color function (with value of unity inside the object and zero outside) on a fixed background mesh used for flow solution. In this work, the material volumes are forced to undergo rigid motion, based on the translational and rotational velocities of the object, resulting in no relative motion among them. At each time step the material volumes are advanced to new locations. In the present approach, the boundary of the object is represented in a stair-stepped fashion and it is straightforward to create the material volumes using a bounding-box algorithm as described in [25].

3.2. Discretized Equations and Numerical Algorithm

Figure 1 shows the schematic of variable storage in time and space. All variables are stored at the control volume (cv) center with the exception of the face-normal velocity u_{N}, located at the face centers. The face-normal velocity is used to enforce continuity equation. Capital letters are used to denote particle fields. The time-staggering is done so that the variables are located most conveniently for the time-advancement scheme. The collocated spatial arrangement for velocity and pressure field ([31], [32]) is used. Accordingly, the particle positions (X_i), density (ρ), volume fraction (Θ), viscosity (μ), and the pressure (p) are located at time level $t^{n-1/2}$ and $t^{n+1/2}$ whereas the velocity fields (u_i, u_{N}, and U_i) and the rigid body constraint force $f_{i,R}$, are located at time level t^n and t^{n+1}. This makes the discretization symmetric in time, a feature important to obtain good conservation properties.

For each time-step, multiple inner iterations can be performed to improve the temporal accuracy and reduce the splitting error in fractional time-stepping schemes. However, this does involve multiple solutions of the pressure equation which can be costly. It was observed from the numerical test cases that multiple inner iterations are not needed for robustness of the algorithm. If the time-steps used for particulate flow simulations are small (most simulations are conducted with CFL $\sim$ 0.5 to obtain good temporal accuracy), a single iteration is sufficient. The semi-discretization of the governing equations with multiple inner iterations in each time-step is given below for completeness.

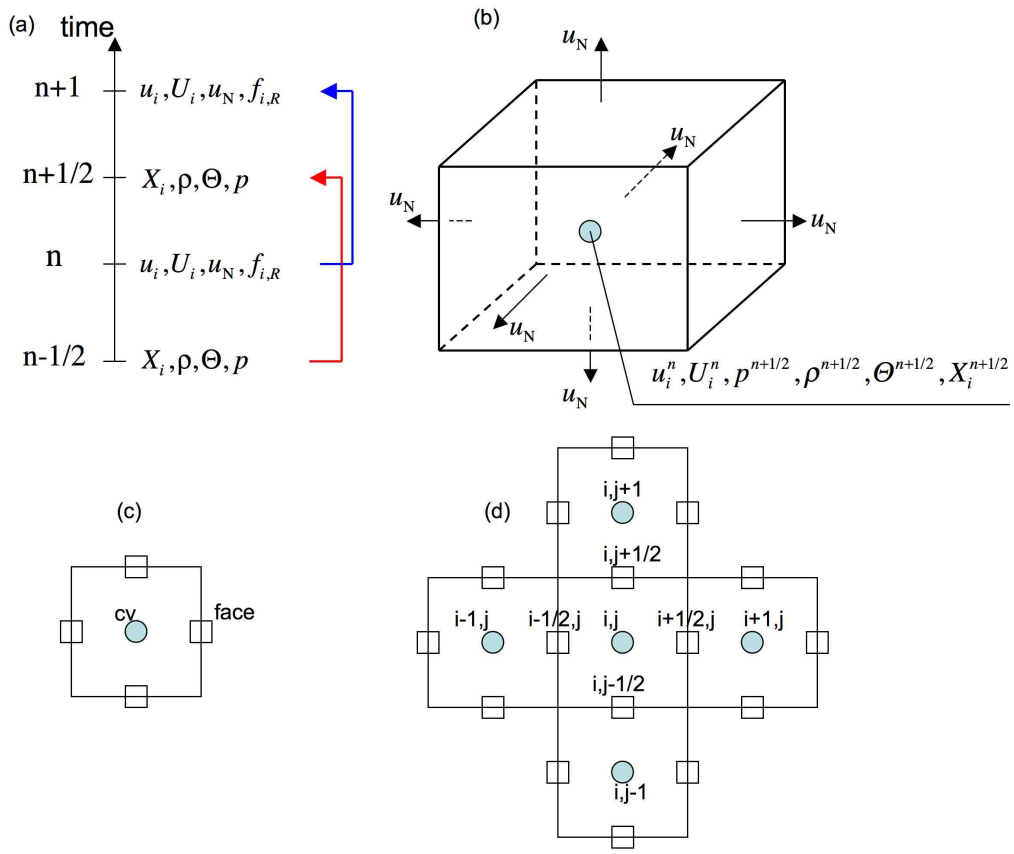


Figure 1: Schematic of the variable storage in time and space: (a) time-staggering, (b) three-dimensional variable storage, (c) cv and face notation, (d) index notation for a given k-index in the z direction. The velocity fields (u_i, u_N) are staggered in time with respect to the volume fraction (Θ), density (ρ), and particle position (X_i), the pressure field (p), and the rigid body force ($f_{i,R}$). All variables are collocated in space at the centroid of a control volume except the face-normal velocity u_N which is stored at the centroid of the faces of the control volume.

Step 1: Starting with a solution at t^n and $t^{n-1/2}$, the centroids of material volumes ($X_{i,M}$) representing immersed objects are first advanced explicitly as

$$X_{i,M}^{n+1/2} = X_{i,P}^{n-1/2} + \mathcal{R}_{ij}\left(X_{j,M}^{n-1/2} - X_{j,P}^{n-1/2}\right) + U_{i,M}^{n}\Delta t, \tag{11}$$

where $X_{i,M}$ is the position vector of the material volume center, $X_{i,P}$ is the position vector of the immersed object centroid, $U_{i,M}$ is the translational velocity, $\Omega_{i,M}$ is the angular velocity, and Δt is the time-step. Here $\mathcal{R}_{ij}$ is the rotation matrix evaluated using particle locations at $t^{n-1/2}$. The details of the particle update and the rotation matrix are similar to that presented in [25] and are given in the appendix for completeness.

Step 2: Knowing the new positions of the material volumes and particle centroid, an indicator function (color function) $\Theta^{n+1/2}$ is evaluated at the cv-center of the fixed background grid. A discrete delta-function developed by [15] is used to compute the color function. The color function is unity inside the particle region and vanishes outside with smooth variation near the boundary. This thus allows identification of the particle on the background mesh. Details of the interpolation between the material volume centers and the cv center are similar to that presented in [25]. The density and the viscosity are then calculated over the entire domain as

$$\rho_{\rm cv}^{n+1/2} = \rho_P\Theta_{\rm cv}^{n+1/2} + \rho_F\left(1-\Theta_{\rm cv}^{n+1/2}\right) \tag{12}$$

$$\mu_{\rm cv}^{n+1/2} = \mu_P\Theta_{\rm cv}^{n+1/2} + \mu_F\left(1-\Theta_{\rm cv}^{n+1/2}\right) \tag{13}$$

where ρ_P is the density of the immersed particle and ρ_F is the density of the surrounding fluid. Likewise μ_P is dynamic viscosity of the fictitious fluid inside the particle region, and μ_F is the dynamic viscosity of the surrounding fluid. For fixed particles, assigning large viscosity for the particle region has been used in the past [33] together with harmonic mean of the viscosity values near the interfaces, especially when viscosity values are obtained at the faces of the control volumes. With large viscosity within the particle region, it is easier to obtain zero velocity for fixed particles. However, in the present work, with use of a rigidity constraint force, the viscosity within the particle region is of little significance and hence is set to be equal to the fluid viscosity for fixed as well as moving particles, to avoid any jumps in fluid properties across the rigid particle boundaries.

Step 3: Choose predictors (initial guesses) for the values of the variables at the next time level. This is done to initiate solution of the momentum equations using multiple inner iterations of preset value. Predictors are chosen using second-order Adams-Bashforth extrapolation for velocity. Starting with $k=0$,

$$u_{\rm N}^{n+1,k} = 2u_{\rm N}^{n} - u_{\rm N}^{n-1} \tag{14}$$

$$u_{i,{\rm cv}}^{n+1,k} = 2u_{i,{\rm cv}}^{n} - u_{i,{\rm cv}}^{n-1} \tag{15}$$

$$U_{i,P}^{n+1,k} = 2U_{i,P}^{n} - U_{i,P}^{n-1}. \tag{16}$$

The pressure is updated using backward Euler, $p^{n+1/2,k} = p^{n-1/2}$ and the rigidity constraint force $f_{i,{\rm cv}}^{n+1,k}$ is obtained by using $u_{i,{\rm cv}}^{n+1,k}$ and $U_{i,P}^{n+1,k}$ and performing interpolations from the material points to the cv center as described later.

Step 4: Advance the momentum equations using the fractional step method. The velocity field is advanced from t^n to t^{n+1},

$$\frac{u_{i,{\rm cv}}^{*,k+1} - u_{i,{\rm cv}}^{n}}{\Delta t} + \frac{1}{V_{\rm cv}}\sum_{\text{faces of cv}} u_{i,\text{face}}^{*,n+1/2}u_{\rm N}^{n+1/2}A_{\rm face} = -\frac{1}{\rho_{\rm cv}^{n+1/2}}\frac{\delta p_{\rm cv}}{\delta x_i}^{n+1/2,k} + \tag{17}$$

$$\frac{1}{\rho_{\rm cv}^{n+1/2}V_{\rm cv}}\left(\sum_{\text{faces of cv}}\tau_{ij,\text{face}}^{*,n+1/2}{\rm N}_{j,\text{face}}A_{\rm face}\right) + g_i + f_{i,{\rm cv}}^{n+1,k} \tag{18}$$

where g_i is the gravitational acceleration, $V_{\rm cv}$ is the volume of the cv, $A_{\rm face}$ is the area of the face of a control volume, ${\rm N}_{j,\text{face}}$ is the face-normal vector and

$$u_{\rm N}^{n+1/2} = \frac{1}{2}\left(u_{\rm N}^{n} + u_{\rm N}^{n+1,k}\right);$$

$$u_{i,\text{face}}^{*,n+1/2} = \frac{1}{2}\left(u_{i,\text{face}}^{n} + u_{i,\text{face}}^{*,k+1}\right);$$

$$\tau^{*,n+1/2}_{ij,\text{face}} = \mu^{n+1/2}_{\text{cv}} \left[\frac{1}{2} \left(\frac{\partial u^{n}_{i,\text{cv}}}{\partial x_j} + \frac{\partial u^{*,k+1}_{i,\text{cv}}}{\partial x_j} \right) + \frac{1}{2} \left(\frac{\partial u^{n}_{j,\text{cv}}}{\partial x_i} + \frac{\partial u^{n+1,k}_{j,\text{cv}}}{\partial x_i} \right) \right]_{\text{face}}$$

In the above expressions, the velocities at the 'face' are obtained by using arithmetic averages of the neighboring cvs attached to the face. The face-normal velocity $u^{n+1,k}_{\text{N}}$ is estimated from previous predictor values whereas the cv-center velocities $u_{i,\text{cv}}$ are treated implicitly. For the viscous terms, the velocity gradients in the direction of the momentum component are obtained implicitly using Crank-Nicholson scheme, whereas other components are treated explicitly. A symmetric, centered discretization scheme is used for spatial gradients. . The predicted velocity fields may not satisfy the continuity or the rigidity constraints. These are enforced later. The old iteration level pressure gradient and rigidity constraint force are retained in the above solution to obtain a complete momentum equation over the entire domain. Evaluation of the pressure gradients at the cv centers is explained below.

Step 5: Remove the old pressure gradient and rigidity constraint force,

$$u^{**,k+1}_{i,\text{cv}} = u^{*,k+1}_{i,\text{cv}} + \frac{\Delta t}{\rho^{n+1/2}_{\text{cv}}} \frac{\delta p_{\text{cv}}}{\delta x_i}^{n+1/2,k} - \Delta t f^{n+1,k}_{i,\text{cv}}. \tag{19}$$

Note that this velocity field satisfies an approximate momentum equation over the entire domain but need not satisfy the incompressibility constraint.

Step 6: Solve the Poisson equation for pressure to enforce incompressibility constraint,

$$\sum_{\text{faces of cv}} \frac{1}{\rho^{n+1/2}_{\text{face}}} \frac{\delta p}{\delta \text{N}}^{n+1/2,k+1} A_{\text{face}} = \frac{1}{\Delta t} \sum_{\text{faces of cv}} u^{**,k+1}_{\text{N}} A_{\text{face}}, \tag{20}$$

where ρ_{face} is obtained using arithmetic averages of density in the neighboring *cv*s. The face-normal velocity u^{*}_{N} and the face-normal pressure gradient are obtained as:

$$\begin{aligned} u^{**,k+1}_{\text{N}} &= \frac{1}{2}(u^{**,k+1}_{i,\text{nbr}} + u^{**,k+1}_{i,\text{cv}})\text{N}_{i,\text{face}} \\ P'_{\text{face}} &= \frac{1}{\rho^{n+1/2}_{\text{face}}} \frac{\delta p}{\delta \text{N}}^{n+1/2,k+1} = \frac{1}{\rho^{n+1/2}_{\text{face}}} \frac{p^{n+1/2,k+1}_{\text{nbr}} - p^{n+1/2,k+1}_{\text{cv}}}{|\mathbf{s}_{\text{cv,nbr}}|} \end{aligned}$$

where 'nbr' represents neighboring 'cv' associated with the 'face' of the cv, and $|\mathbf{s}_{\text{cv,nbr}}|$ is the distance between the two *cv*s. Here P'_{face} represents the pressure gradient per unit density at the face of a control volume. The variable-coefficient pressure equation is solved using a Bi-Conjugate gradient algorithm [34].

Step 7: Reconstruct the pressure gradient at the cv centers using density and face-area weighting

$$P'_{,i} = \frac{1}{\rho^{n+1/2}_{\text{cv}}} \frac{\delta p}{\delta x_i}^{n+1/2,k+1} = \frac{\sum_{\text{faces of cv}} P'_{\text{face}} \cdot \vec{i} |\text{N}_{i,\text{face}} A_{\text{face}}|}{\sum_{\text{faces of cv}} |N_{i,\text{face}} A_{\text{face}}|}. \tag{21}$$

Using the density weighted reconstruction, provides the pressure gradient per unit density at the cv centers which can be directly used in correcting the velocity fields. Not using density weighted pressure gradient reconstruction was found to be less stable, especially for large density ratios.

Step 8: Update the cv-center and face-normal velocities to satisfy the incompressibility constraint:

$$\widehat{u}^{k+1}_{i,\text{cv}} = u^{**,k+1}_{i,\text{cv}} - \Delta t P'_{,i} \tag{22}$$

$$\widehat{u}^{k+1}_{\text{N}} = u^{**,k+1}_{\text{N}} - \Delta t P'_{\text{face}} \tag{23}$$

With the above correction, the face-normal velocity field $\widehat{u}^{k+1}_{\text{N}}$ will satisfy the incompressibility constraint. At this stage, the cv-based velocity may not satisfy the rigid-body constraint inside the particle region. Note that in the absence of any rigid body, $\rho = \rho_F$ throughout the domain, and the algorithm reduces to the standard fractional step scheme for single-phase, incompressible flow. The above velocity field will then be denoted as $u^{n+1}_{i,\text{cv}}$. In the presence of rigid bodies, the following steps are performed to enforce the rigidity constraint within the particle domain.

Step 9: First interpolate the velocity field $\widehat{u}_{i,\mathrm{cv}}^{k+1}$ from the grid *cv*s to the material volume centroids to obtain $\widehat{U}_{i,M}^{k+1}$ using the kernel interpolation outlined in [25] and given in Appendix. Once the interpolated velocity field at the material volume centers is obtained, solve for the translational and rotational velocity of the particle, by simple additions over all material volumes:

$$\mathcal{M}_P \mathbf{U}_P^{T,k+1} = \sum_{M=1}^{N} V_M \rho_M \widehat{\mathbf{U}}_M^{k+1} \tag{24}$$

$$\mathcal{I}_P \boldsymbol{\Omega}_P{}^{k+1} = \sum_{M=1}^{N} \rho_M V_M (\mathbf{r} \times \widehat{\mathbf{U}}_M^{k+1}), \tag{25}$$

where subscripts P and M denote the particle and the material volume centroids respectively, V_M is the volume and ρ_M the density of each material volume, $\mathcal{M}_P = \sum_{M=1}^{N} \rho_M V_M$ is the total mass of the particle, $\mathcal{I}_P$ is the moment of inertia of the particle about the coordinate axes fixed to the particle centroid, and $\mathbf{r}$ is the position vector of a point within the particle region with respect to the particle centroid.. The moment of inertia is given as

$$\mathcal{I}_P = \sum_{M=1}^{N} \rho_M V_M \left[(\mathbf{r} \cdot \mathbf{r}) \mathbf{I} - \mathbf{r} \otimes \mathbf{r} \right], \tag{26}$$

where $\mathbf{I}$ represents the identity matrix. The rigid body motion at the material volume center is then obtained as

$$\mathbf{U}_M^{RBM,k+1} = \mathbf{U}_M^{T,k+1} + \boldsymbol{\Omega}_P^{k+1} \times (\mathbf{X}_M - \mathbf{X}_P), \tag{27}$$

where $\mathbf{U}_M^{T,k+1} = \mathbf{U}_P^{T,k+1}$.

Step 10: Compute the rigid-body constraint force and correct the velocity field to satisfy this constraint within the particle region.

$$\mathrm{F}_{i,M}^{n+1,k+1} = -\frac{(\widehat{U}_{i,M}^{k+1} - U_{i,M}^{RBM,k+1})}{\Delta t}. \tag{28}$$

The force on the grid control volumes ($f_{i,\mathrm{cv}}^{n+1,k+1}$) is obtained from $\mathrm{F}_{i,M}^{n+1,k+1}$ through a consistent interpolation scheme with the same kernel as used for interpolating the *cv*-center values to the material volume centroids [25] and is given in Appendix. The velocity field inside the particle region is then modified as

$$u_{i,\mathrm{cv}}^{n+1,k+1} = \widehat{u}_{i,\mathrm{cv}}^{k+1} + \Delta t f_{i,\mathrm{cv}}^{n+1,k+1}. \tag{29}$$

Step 11: Set $k = k + 1$, check for convergence and repeat by going to Step 4 for a preset number of inner iterations. After completion of the inner iterations, $u_{i,\mathrm{cv}}^{n+1,k+1} = u_{i,\mathrm{cv}}^{n+1}$.

3.3. Density weighted reconstruction of pressure gradient

In order to understand the effectiveness of the density-weighted reconstruction of the pressure gradient at the cv centers (Step 7 in the above algorithm), a simple one-dimensional example with uniform grid spacing of Δx as shown in figure 2a is devised. As an example, the pressure values at the cv centers are assumed to be equal to 90, 100, and 110 for the W, P, and E control volumes, respectively, corresponding to a linear variation. A simple reconstruction of the pressure gradient at the cv centers from the pressure values at the cv centers is given as,

$$\frac{\delta P'}{\delta x}^{\mathrm{CD}} = \frac{1}{\rho_{\mathrm{cv}}} \frac{\delta p}{\delta x}^{\mathrm{CD}} = \frac{1}{\rho_{\mathrm{cv}}} \left(\frac{p_E - p_W}{2\Delta x} \right), \tag{30}$$

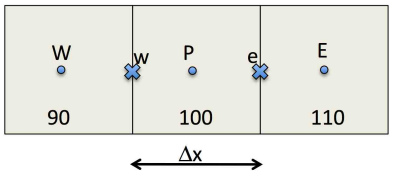
where E and W correspond to the east and west neighbors of point P, respectively. For simplicity, the superscripts associated with time-level are dropped in this example without lack of generality. This is the standard centered difference approximation for the first derivative.

A density-weighted reconstruction given by equation 21 provides the following expression for the pressure-gradient at the cv centers,

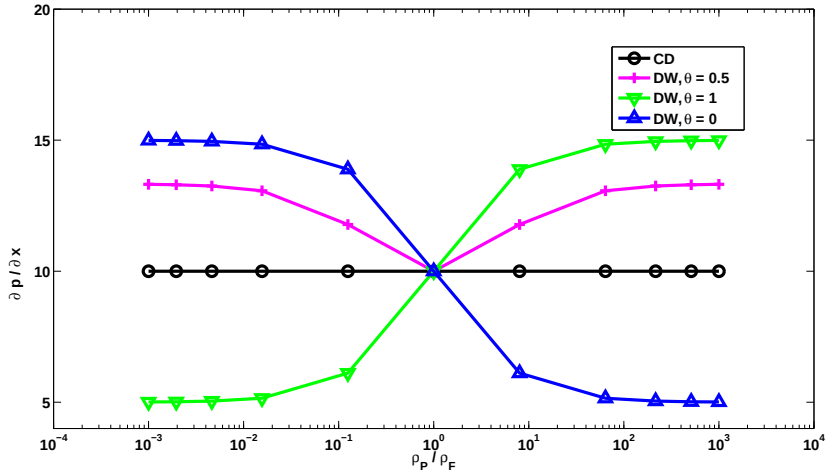
$$\frac{\delta P'}{\delta x}^{\mathrm{DW}} = \frac{1}{\rho_{\mathrm{cv}}} \frac{\delta p}{\delta x}^{\mathrm{DW}} = \frac{\frac{1}{\rho_e} \frac{p_E - p_{\mathrm{cv}}}{\Delta x} A_e - \frac{1}{\rho_w} \frac{p_W - p_{\mathrm{cv}}}{\Delta x} A_w}{A_e + A_w} \tag{31}$$

$$= \frac{\frac{1}{\rho_e} p_E + \left(\frac{1}{\rho_w} - \frac{1}{\rho_e} \right) p_{\mathrm{cv}} - \frac{1}{\rho_w} p_W}{2\Delta x}, \tag{32}$$

where DW stands for density-weighted reconstruction, e and w correspond to the east and west faces of the control volume cv, A_e and A_w are the areas of the east and west faces (assumed equal for uniform grids), respectively. Notice that for uniform density, $\rho_e = \rho_w = \rho_{\rm cv}$ the above expression is identical to the standard central differencing.



(a)



(b)

Figure 2: Comparison of the standard centered difference (CD) and density weighted (DW) pressure gradient construction: (a) grid stencil, (b) pressure gradient for wide range of density ratios with $\Theta_P = 0, 0.5, 1$ values together with $\Theta_W = 1$ and $\Theta_E = 0$. Pressure values at P, W and E are assumed as 100, 90, and 110, respectively, corresponding to a linear variation.

For the example considered, the interface between the solid and fluid is assumed to lie within the control volume P, with the west cv being pure solid $\Theta = 1$ and east cv a pure fluid $\Theta = 0$. The interface location within the cv P, can be varied by changing Θ_P values between 0 and 1. For each interface location, the density-to-fluid ratio is varied over the range of 10^{-3}-10^{3} and the pressure gradients ($\delta p/\delta x$) as obtained from the central differencing and the density weighted construction are plotted in figure 2b. The central differencing predicts a constant pressure gradient across the interface, irrespective of the interface location. However, the density-weighted reconstruction assigns different weights to the face-based pressure gradients, and appropriately alters the pressure gradient at the cv P. As is shown later through a wide range of test-cases, this pressure-gradient construction is found to be stable as well as accurate and approaches the standard gradient approximation based on central differencing in the limit of density ratios close to unity (or neutrally buoyant particles).

3.4. Numerical errors

As mentioned earlier, multiple inner iterations ($k > 1$), are usually employed when using large time-steps to minimize the splitting error and to improve robustness of the algorithm. From the above steps, it can be shown that the total splitting error in the above fractional step is

$$u_{i,\rm cv}^{n+1,k+1} - u_{i,\rm cv}^{*,k+1} = -\Delta t \left[\frac{1}{\rho_{cv}^{n+1/2}} \frac{\delta}{\delta x_i} \left(p_{\rm cv}^{n+1/2,k+1} - p_{\rm cv}^{n+1/2,k} \right) - \left(f_{i,\rm cv}^{n+1,k+1} - f_{i,\rm cv}^{n+1,k} \right) \right] \tag{33}$$

$$= -\Delta t^2 \left[\frac{1}{\rho_{cv}^{n+1/2}} \frac{\delta}{\delta x_i} \left(\frac{\delta p_{\rm cv}}{\delta t} \right) + \frac{\delta f_{i,\rm cv}}{\delta t} \right] + \mathcal{O}\left(\delta p_{\rm cv}^2 + \delta f_{i,\rm cv}^2 \right). \tag{34}$$

Note that $\delta p_{\rm cv}$ and $\delta f_{i,{\rm cv}}$ are defined as the differences between iteration levels (k and $k+1$) for the pressure and the rigidity constraint force, respectively. By performing multiple iterations within each time step, the splitting error can be reduced. If $k = 1$, the splitting error is dependent on the initial guess for $p_{\rm cv}^{n+1/2,k}$ and $f_{i,icv}^{n+1,k}$. With multiple iterations, the Poisson solver is solved multiple times per iteration. After the first iteration, subsequent solutions of the Poisson solver are not as computationally intensive. In the present work, the time-step used is such that the maximum CFL ≤ 1 at all times and only a single iteration is sufficient in the test cases studied in this work.

The other source of numerical error is the interpolation from the grid points to the material volumes and back from the material volumes to the grid points. In the above algorithm, interpolations to the material volumes and back to the grid points are required once per time-step. It is important to use at least second-order, conservative kernels that will conserve the inter-phase force and torque during the interpolation procedure. The interpolation operator described in the Appendix is used in this study. Recently, Kempe & Frohlich [21] looked at improving the immersed boundary method based on the approach first proposed by Uhlmann [9] in order to improve the stability of the original scheme. They found that inconsistencies in interpolations between the background grid and the Lagrangian points can result in considerable inaccuracies in enforcing the no-slip conditions at the fluid-solid boundary. In order to improve the accuracy and reduce any inconsistencies, they proposed a less costly inner iteration, wherein the iterations are done over steps 9 and 10 only in the algorithm. This avoids solution of the pressure solve multiple times, but evaluates the control volume material point interpolations in a more consistent manner. In the present approach, subgrid resolution of the material volumes is used and the rigidity constraint force is applied over the entire rigid body region as opposed to only near the interface region by Uhlmann [9] and Kempe & Frohlich [21]. The subgrid material volumes used in the present approach allows consistent interpolations between the material volumes and the background grid and multiple iterations to reduce interpolation errors are not necessary, as is shown below.

4. Numerical Test Cases

The above numerical algorithm is implemented in a parallel, finite volume framework and validated for a number of test cases: (i) flow over a fixed sphere and hydrofoil, (ii) particle subjected to constant acceleration for varying fluid-particle density ratios, and (iii) freely falling/rising particles at low and high density ratios. Finally, interactions of a buoyant sphere with a stationary Gaussian vortex at different density ratios are simulated to test the capability of the algorithm.

4.1. Flow over a Cylinder

Flow over a fixed circular cylinder is investigated following the case previously considered by Mittal *et al.* [11] for testing their sharp interface immersed boundary method. A cylinder with diameter d is centered in a $2d \times 2d$ domain. The flow is started from rest and driven by a uniform inflow velocity, U_∞, in the axial (X) direction so that $Re = U_\infty d/\nu = 100$. Similar to [11] we set U_∞ and d to unity, so that all quantities may be considered dimensionless. Periodic boundaries are applied in the vertical (Y) direction. This small test case allows thorough investigation of the spatial convergence of the scheme for moderate Reynolds number flows.

For this short domain, there is no exact solution for the problem. In order to investigate spatial discretization errors, a solution up to $t = 0.2s$ on a very fine, 800×800, grid with a small timestep, $dt = 1 \times 10^{-5}$, is computed to ensure that temporal discretization errors are small. The same flow is then computed on a sequence of coarser grids so that the spatial order of convergence may be observed. Grid resolutions used within the particle region are $d/\Delta = 25$, 50, and 100 corresponding to $\Delta = 0.04$, 0.02, and 0.01, respectively. The error on each grid is computed by interpolating the very fine grid solution to the coarse grid nodes and calculating the difference in solutions. In Figure 3, the L_1, L_2, and L_∞ norms of both the axial and vertical velocity errors are shown as a function of grid spacing. The magnitude of these errors is comparable to those obtained by Mittal *et al.* [11] on similar grid resolutions. Slightly lower convergence rate is observed for the present approach on coarser meshes, however, near second order convergence is obtained for sufficiently refined grids as shown in figure 3. Since the cylinder boundary is represented by material volumes in a stair-stepped fashion, with grid refinement, the boundary is approximated much more accurately, and all errors converge at a rate close to second-order.

Next, the uncertainty in the velocity field is evaluated using the *Grid Convergence Index* (GCI) [35]. The GCI does not rely on the existence of an exact solution or the assumption that a very fine grid solution may be taken as such (as was done above), and is robust as a general post-processing tool for error estimation in

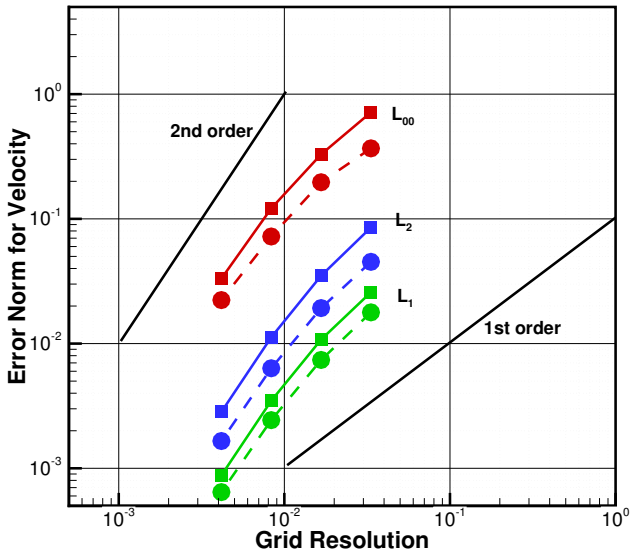


Figure 3: Error norms in axial (solid line) and vertical (dotted line) velocity components for flow over a cylinder at $Re = 100$.

CFD calculations. This approach is used to provide an error band for some later results, and this case serves to demonstrate the method in some detail. To compute the GCI uncertainty of the flow variable, ϕ, the procedure outlined by Cadafalch *et al.* [36] and Celik *et al.* [37] is followed, which requires the the solution for ϕ to be obtained on three grids with (not necessarily equal) refinement ratios $r_{21} = \Delta_2/\Delta_1$ and $r_{32} = \Delta_3/\Delta_2$. The fine and medium grid solutions, ϕ_1 and ϕ_2, are first interpolated to the coarse grid, where the variations, $\epsilon_{32}(\mathbf{x}) = \phi_3(\mathbf{x}) - \phi_2(\mathbf{x})$, and $\epsilon_{21}(\mathbf{x}) = \phi_2(\mathbf{x}) - \phi_1(\mathbf{x})$, are computed. From $\epsilon_{32}(\mathbf{x})$ and $\epsilon_{21}(\mathbf{x})$, the *local* apparent order of accuracy is calculated, using the following equation [37]

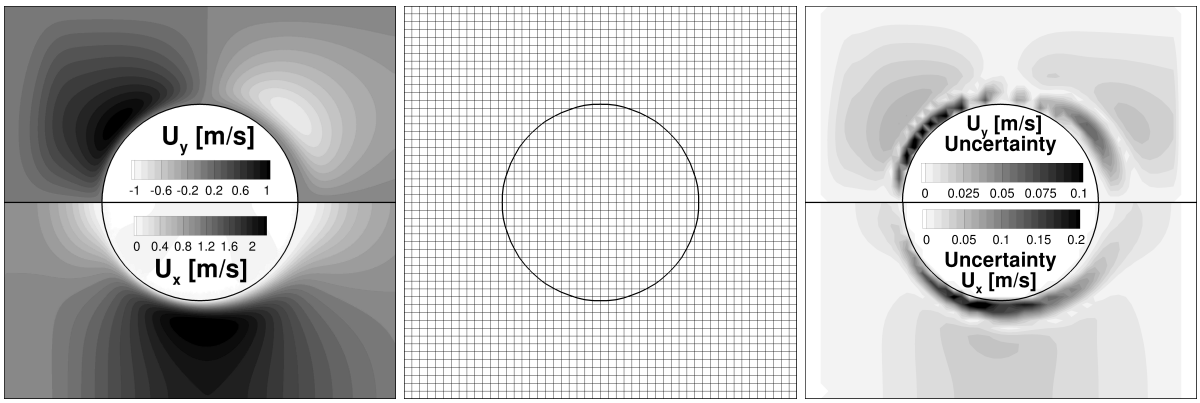
$$\mathcal{P}(\mathbf{x}) = \frac{\ln\left|\epsilon_{32}(\mathbf{x})/\epsilon_{21}(\mathbf{x})\right|}{\ln\left(r_{21}\right)} + \ln\left(\frac{r_{21}^{\mathcal{P}(\mathbf{x})} - sign(\epsilon_{32}/\epsilon_{21})}{r_{32}^{\mathcal{P}(\mathbf{x})} - sign(\epsilon_{32}/\epsilon_{21})}\right). \tag{35}$$

In the event that $r_{21} \neq r_{32}$, a straightforward Picard iteration of equation 35 can be used to determine $\mathcal{P}(\mathbf{x})$. The global order of convergence, $\mathcal{P}_G$, is then computed by averaging $\mathcal{P}(\mathbf{x})$ at nodes where asymptotic convergence is observed, indicated by $sign(\epsilon_{32}/\epsilon_{21}) > 0$. Using the global order of convergence, the GCI uncertainty of the fine grid solution is then computed as

$$GCI(\mathbf{x}) = F_s \left|\frac{\phi_1(\mathbf{x}) - \phi_2(\mathbf{x})}{1 - r^{\mathcal{P}_G}}\right|, \tag{36}$$

where $F_s = 1.25$ is generally agreed to be a reasonable factor of safety [37, 36]. Finally the globally averaged uncertainty, GCI_G, can be computed by averaging $\text{GCI}(\mathbf{x})$, again over nodes where $sign(\epsilon_{32}/\epsilon_{21}) > 0$. Figure 4 illustrates this process for the axial and vertical velocity components in the cylinder flow for $\Delta_1 = 0.01$, $\Delta_2 = 0.02$, $\Delta_3 = 0.04$. The fine and medium grid velocity components shown in figure 3(a) are interpolated to the coarse grid nodes shown in figure 4b. The local order of convergence, $\mathcal{P}(\mathbf{x})$ is computed on this grid using equation 35. This is averaged over the asymptotic nodes to obtain, $\mathcal{P}_G$, which is used to compute the local GCI, plotted as the uncertainty in figure 4c. The largest uncertainties are located near the upstream surface of the cylinder, where the accelerating flow results in a thin boundary layer that is under-resolved by the current grids.

Key results of the GCI analysis are also summarized in table 1. Effect of inner iterations on the uncertainty is also evaluated. As described in the algorithm, an inner iteration over the entire solution procedure (Steps 4 through 11) improves robustness and accuracy of the approach by minimizing the splitting error in this fractional-step approach. This, termed as 'Solution Iterations' in table 1, involves multiple solutions to the pressure equation. Recently, Kempe & Frohlich [21] have proposed several modifications to the widely used immersed boundary



(a) Velocity field at $t = 0.2s$ (b) Post-processing mesh ($\Delta_3 = 0.04$) (c) GCI uncertainty for ($\Delta_1 = 0.01$)

Figure 4: Post processing of the velocity field to determine uncertainty in velocity components.

Table 1: Uncertainty quantified using the GCI for the cylinder case with $\Delta_1 = 0.01,\ \Delta_2 = 0.02,\ \Delta_3 = 0.04$.

ϕ	Solution Iterations	Forcing Loops	$\mathcal{P}_G$	GCI_G [m/s]	MAX($GCI(\mathbf{x})$) [m/s]
U_x	1	1	1.49	0.0234	0.2776
U_x	1	3	1.41	0.0254	0.3041
U_x	1	5	1.39	0.0263	0.3152
U_x	3	1	1.44	0.0237	0.2804
U_x	5	1	1.44	0.0237	0.2804
U_y	1	1	1.54	0.0130	0.1200
U_y	1	3	1.51	0.0138	0.1322
U_y	1	5	1.54	0.0136	0.1322
U_y	3	1	1.54	0.0130	0.1200
U_y	5	1	1.54	0.0130	0.1200

scheme of Uhlmann [9]. With their scheme, which utilizes forcing points with roughly the same spacing as the background grid, they demonstrated improvement in the imposition of no-slip particle boundaries by improving inconsistencies in interpolations between the forcing points (or material points) and the background grid and vice-versa. They proposed performing multiple iterations only on the interpolation steps, that is looping over steps 9 and 10 which enforce the rigidity constraint force in our scheme. This approach does not involve iterations over the entire solution and the pressure Poisson equation is solved only once per time-step. This is referenced as iterations over 'Forcing Loops' in table 1.

In the present approach, subgrid resolution of the material volumes is used and the rigidity constraint force is applied over the entire rigid body region as opposed to only near the interface region by Uhlmann [9] and Kempe & Frohlich [21]. It is observed that the subgrid material volumes make multiple forcing loops of lower importance. On a relatively coarse grid with $d/\Delta = 25$ the maximum velocity interpolated to our cylinder boundary is 3% of U_∞ at $t = 0.2$ s. Table 1 summarizes the GCI results for the cylinder case performed with multiple forcing loops, as well as multiple inner iterations, where the Poisson equation is solved. The extra interpolations and iterations were found to have no measurable effect on the convergence behavior of the solution in the present approach. As noted earlier, the inner solution iterations only improve the splitting error and since a small time-step is used in the present work, multiple inner iterations do not show effect on convergence. Multiple iterations on the forcing loop (interpolation steps) also do not alter the convergence, indicating that the grid to material volume interpolation and vice-versa is done consistently in the present scheme. The global order of convergence remains fixed at roughly 1.5 for all cases. This is slightly lower than the sharp interface approach by Mittal *et al.* [11] and is attributed to the fact that in the present approach, the rigidity constraint force is regularized around the interface. With refined grids, however, it was found to be closer to 2 similar to figure 3 as the interface gets represented more accurately with refined grids. The global and max uncertainties are nearly identical in all cases.

4.2. Flow over a Sphere

To evaluate the accuracy of the algorithm for three-dimensional configurations, flow over a fixed sphere at different Reynolds numbers is evaluated and compared with published data. A sphere of diameter $d = 1.1$ mm is placed in a domain of $15d \times 15d \times 15d$. The sphere is located at $x = 5d$ and $y = z = 7.5d$. The grid used is $128 \times 128 \times 128$. The grid is uniform and refined around the sphere forming a patch of $1.5d \times 1.5d \times 1.5d$. There are approximately 26 grid points along the diameter of the sphere. For comparison Mittal *et al.* [11] used a domain of $16d \times 15d \times 15d$ and a grid of $192 \times 120 \times 120$ for their highest Reynolds number of 350 and Marella *et al.* [38] employed a $130 \times 110 \times 110$ mesh on a $16d \times 15d \times 15d$ domain. The fluid properties are $\rho = 1$ kg/m^3 and the viscosity $\mu = 10^{-5}$ kg/m.s. The x direction is slightly moved towards the inlet in order to increase the size of the domain in the wake. Also the density of grid-points is increased in the wake of the sphere in order to properly resolve the flow. Table 2 compares the predicted drag coefficients with other published data showing very good agreement.

Table 2: Drag coefficient C_d for flow over a sphere at different Reynolds number

Study	Re_p					
	20	*50*	*100*	*150*	*300*	*350*
Current	**2.633**	**1.550**	**1.101**	**0.907**	**0.686**	**0.649**
Mittal [39]	-	1.57	1.09	-	-	0.62
Mittal *et al.* [11]	-	-	1.08	0.88	0.68	0.63
Clift *et al.* [40]	2.61	1.57	1.09	0.89	0.684	0.644
Johnson & Patel [41]	-	1.57	1.08	0.9	0.629	-
Marella *et al.* [38]	-	1.56	1.06	0.85	0.621	-
Kim *et al.* [19]	-	-	1.087	-	0.657	-

Figure 5 shows the vortical structure represented by the eigenvalues of the velocity gradient tensor λ in the xz plane for $Re_P = 350$. Qualitatively the plots show very similar structures as shown by Mittal [39]. This snapshot shows the vortex ring in the near wake of the sphere. Another important feature is the symmetry axis shown in the xz plane which has been observed experimentally.

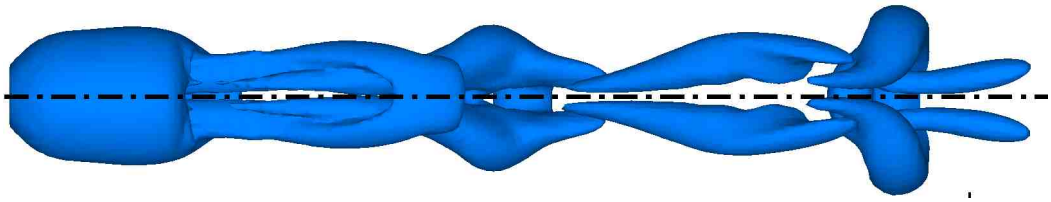


Figure 5: Iso-surface of $\lambda = 0.008$ for flow over a stationary sphere at $Re_P = 350$ in the xz plane. The dash-dotted line shows the symmetry axis of the structure in this plane.

4.3. Rising/falling particle with small density ratio

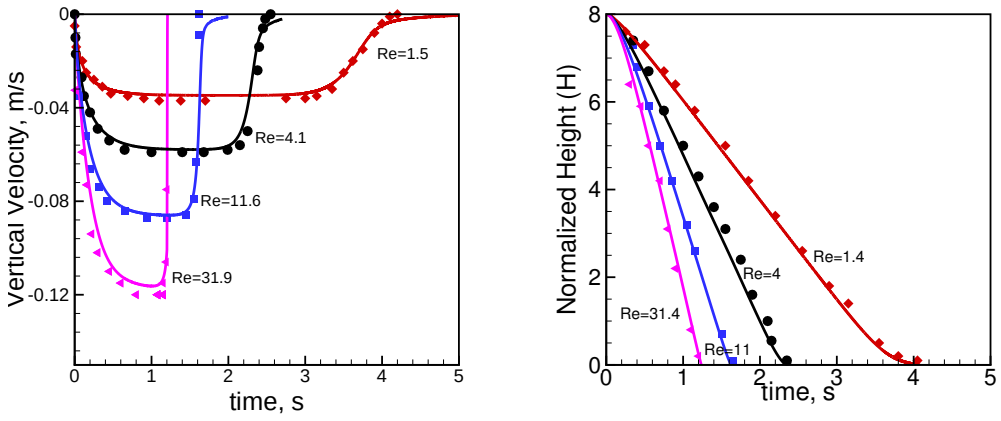
First, the numerical algorithm is tested on simple cases of freely falling (heavier-than fluid) and rising (lighter-than-fluid) particle under gravity. The particle density is on the order of the fluid density (density ratio 0.97-1.2). The computational results are compared with available experimental data. In addition, a systematic grid refinement study is also conducted to study the convergence of the predicted solutions.

Freely falling particle with low particle-to-fluid density ratio

The problem of a single sphere falling under gravity in a closed container is considered. The particle density is ($\rho_P = 1120$ kg/m^3) and the diameter is (15 mm). The sphere is settling in a box of dimensions $10 \times 10 \times 16$ cm^3. The particle is released at a height of $h = 12$ cm from the bottom of the box. The boundaries of the box are treated as no-slip walls. The fluid properties are varied to obtain different Reynolds numbers based on the terminal velocity of the particle. The simulation conditions correspond to the experimental study by ten Cate *et al* [42]. Table 3 provides detailed information about the parameters used in this test problem. The above cases

Table 3: Parameters for freely falling sphere.

	ρ_F (kg/m^3)	ρ_P/ρ_F	μ_F (10^{-3}Ns/m^2)	Re
#1	970	1.154	373	1.5
#2	965	1.1606	212	4.1
#3	962	1.1642	113	11.6
#4	960	1.166	58	31.9



(a) Fall Velocity

(b) Normalized Height

Figure 6: Comparison with the experimental data of the sphere fall velocity and the normalized height from the bottom wall for different Reynolds numbers: (Symbols: experiment by [42], lines: present simulation) Here $H = \frac{h-0.5d}{d}$ where h is the height of the sphere center from the bottom wall and d is the particle diameter.

are simulated on a uniform grid of $100 \times 100 \times 160$ points with a grid resolution of $\Delta = 1$ mm. This provides

around 15 grid points inside the particle domain. The material volumes are cubical with $\Delta/\Delta_M = 2$, where Δ_M is the size of the material volume. A uniform time-step ($\Delta t = 0.5$ ms) is used for all cases.

Figures 6a-b show the comparison of the time evolution of particle settling velocity and position at different times obtained from the numerical simulations with the experimental data by [42]. The simulation predictions for both the particle velocity and the particle position show good agreement with the experimental data. The slowing of the particle towards the end of the simulation is to due to the presence of the bottom wall.

In order to understand the convergence rate and uncertainty in numerical prediction, a systematic grid refinement study was conducted for the $Re = 31.9$ case. Three grids with uniform resolution of $100 \times 100 \times 160$, $150 \times 150 \times 240$, and $200 \times 200 \times 320$ were used for this analysis. The time-step for the coarse grid was same as above, whereas for refined grids, it was lowered in order to keep the CFL the same for all simulations. The GCI uncertainty of the fine grid solution is computed as described in section 4.1, and is plotted as error bars in figure 7. The error bars show a relative uncertainty of 5.8% in the fine grid solution at $t = 1$, when the particle is close to its terminal velocity. Overall the uncertainty in prediction is found to be small, and indicates convergence to the experimental data.

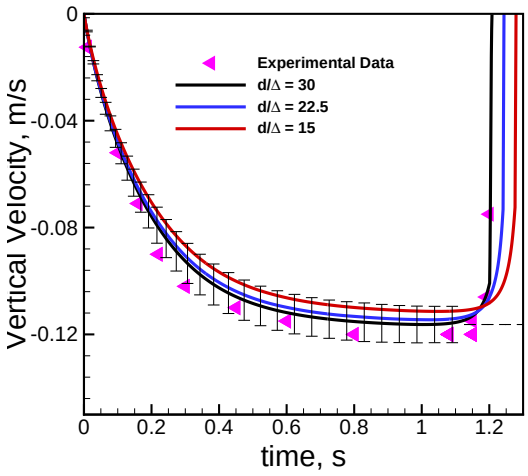


Figure 7: Grid convergence study and uncertainty analysis for falling sphere, $Re = 31.9$ showing error bars around the solution predicted by the fine resolution. Also shown is experimental data [42] for reference.

Lighter than fluid sphere rising in an inclined channel

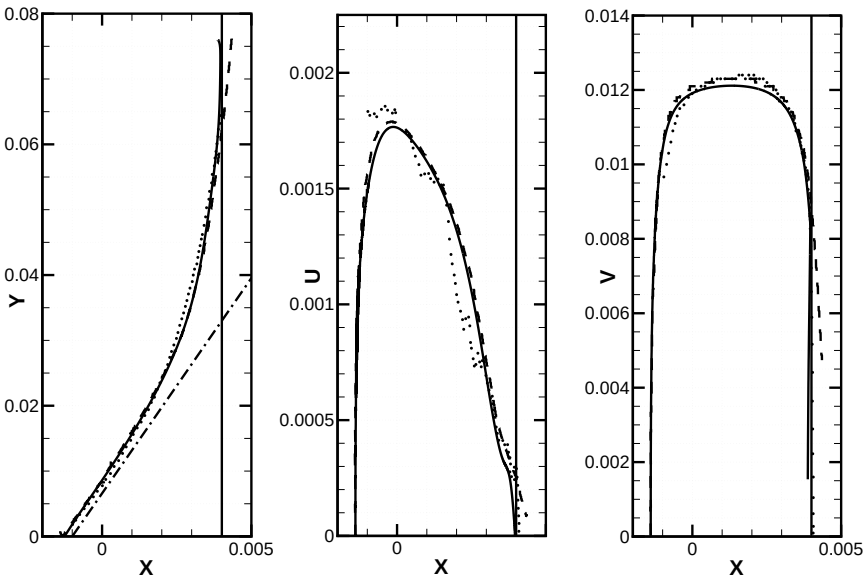
A lighter-than-fluid particle rising in an inclined channel is considered. This is an important test case corresponding to the experiment by Lomholt *et al.* [43]. The particle rises in this inclined channel owing to gravity. However, it slows down as it approaches a wall, it forms a lubrication layer between the particle and the wall. It does not touch or collide with the wall. Instead it glides along the wall without touching it. This is a critical test to see if the numerical approach can capture the hydrodynamics of the lubrication layer properly.

The simulation is conducted with a fluid density of $\rho_F = 1115$ kg/m^3, a particle density of $\rho_P = 1081$ kg/m^3, and a fluid viscosity of $\nu = 3.125$ mm^2/s. The Reynolds number Re_P^{Stokes} based on the Stokes settling velocity W is defined as :

$$Re_P^{Stokes} = \frac{2dW}{\nu} = \frac{4d^3}{9\nu^2}\left|\frac{\rho_P}{\rho_F} - 1\right|g \tag{37}$$

where $g = 9.82$ m/s^2 is the gravitational acceleration, and $d = 2$ mm is the diameter of the particle. The channel is inclined at an angle of 8.23° with the vertical. This is simulated by adding components of gravitational forces in the horizontal and vertical directions.

The computational domain consists of a rectangular box with dimensions 10 mm in the x direction, 80 mm in the y direction and 40 mm in the z direction. The grid is Cartesian and uniform over the domain with $40 \times 320 \times 160$ grid points, respectively in the x, y and z directions so that $\Delta = 0.25$ mm. The particle is injected at $x = -1.4$ mm, $y = -1.0$ mm and $z = 20.0$ mm. Simulation results for a Reynolds number of $Re_P^{Stokes} = 13.6$ are



(a) (b) (c)

Figure 8: Comparison of predicted results by the present scheme — with the experimental • • • and numerical results − − − by Lomholt *et al.* [43]. Figure 8(a) shows the particle trajectory inside the domain (the −.− line shows the initial trajectory due only to the effect of gravity), 8(b) the velocity of the particle in the lateral direction and 8(c) the velocity on the vertical direction. The particle position is expressed in [m] and the velocities are expressed in [m/s]. A solid vertical line at X = 0.004 represents the wall.

compared with experimental and numerical data from Lomholt *et al.* [43]. As illustrated in figure 8, the numerical simulation exhibits excellent agreement with both experimental and numerical results. Buoyancy forces cause the particle to rise and travel alongside the right wall of the domain. Ultimately, the particle follows the right wall without touching it, keeping a very thin lubrication layer between the particle and the wall.

4.4. Sphere subjected to uniform acceleration

To test the stability and robustness of the numerical algorithm for high density ratio between the particle and the fluid and at high Reynolds numbers, we consider motion of a spherical particle subjected to uniform acceleration ($\mathbf{a}$) in a closed box. If the particle is in vacuum (i.e. viscous effects and fluid density are negligible), this case has a simple analytical solution of $\mathbf{u} = \mathbf{u}_0 + \mathbf{a}t$; where $\mathbf{u}$ and $\mathbf{u}_0$ are the instantaneous and initial velocity fields, respectively and t is time. Initially, a particle of diameter 5 mm is placed at the center of a cubic box of 3 cm. Uniform cubic grids of $100 \times 100 \times 100$ grid points is used. The fluid density is set to 1 kg/m^3 and the viscosity is set to zero. Gravitational acceleration is also neglected. Although the fluid has finite density (as opposed to perfect vacuum), the predicted solution is comparable to the analytical solution at early times. The particle starts from rest and is subjected to uniform acceleration of $(-40, -40, -40)$ m/s^2. The particle density is varied over several orders of magnitude, 10^3, 10^4 and 10^6, and the distance traveled by the particle is compared to the analytical solution of $S = |\mathbf{u}_0|t + 1/2|\mathbf{a}|t^2$. Figure 9 shows the temporal evolution of relative error in the distance traveled by the particle compared to the exact solution ($|\frac{S_{\text{num}} - S_{\text{exact}}}{S_{\text{exact,t=0.01}}}|$) over 1000 iterations with fixed time step of 10 μs. It is observed that the error remains small for high density ratios. The numerical algorithm was found to be very stable for wide range of density ratios and even in the limit of zero viscosity.

4.5. Freely falling/rising sphere for a broad range of density ratios

To test the accuracy of the numerical algorithm for high density ratio between the particle and the fluid, the same problem of freely falling/rising sphere is considered, except the density of the particle is varied and the fluid density is kept fixed at $\rho_F = 1$ kg/m^3. The particle density (ρ_P) is varied over a wide range 0.002, 0.001, 0.1, 10, 100, and 500. To keep the terminal velocity of the particle small the viscosity of the fluid is set at 0.06 kg/m.s. The diameter of the particle, the domain size, computational grid and release point are same as that of the falling sphere calculation at low-density ratio presented earlier. For lighter than fluid particles, the particle is placed near the bottom wall instead of the top wall and it rises up owing to buoyancy.

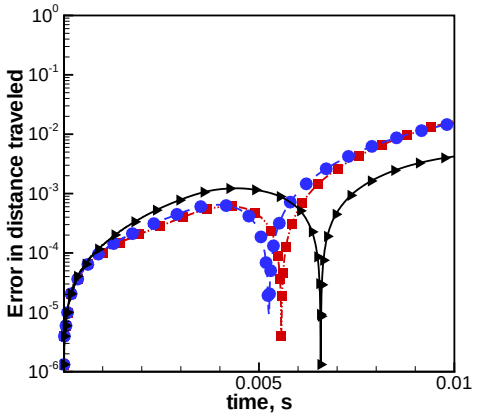


Figure 9: Temporal evolution of the relative error in distance traveled by the sphere under uniform acceleration: ▲ $\rho_P/\rho_F = 10^3$, ● $\rho_P/\rho_F = 10^4$, □ $\rho_P/\rho_F = 10^6$.

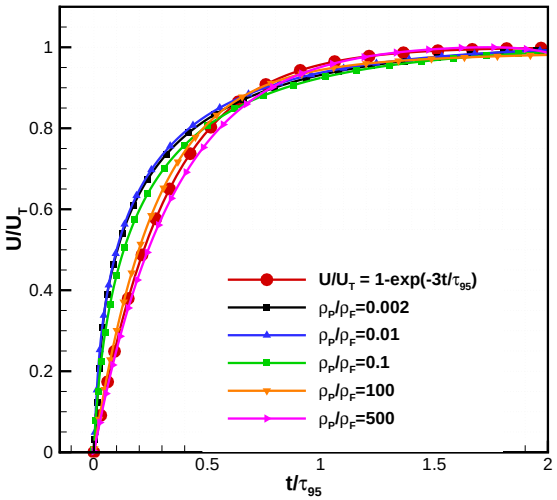


Figure 10: Temporal evolution of a spherical particle velocity falling/rising from rest and normalized by the terminal velocity U_T. τ_{95} is the time at which the particle reaches 95% of its terminal velocity. The data for different density ratios is compared with experimentally obtained correlation by Mordant and Pinton [44].

Mordant and Pinton [44] performed experiments on freely falling spherical particles in a large water tank for various density ratios (maximum density ratio considered was $\rho_P/\rho_F = 14.6$). They showed that for small particles falling in a large tank (that is, for small values of the ratio of particle diameter to tank width $d/L \sim 0.005$), the temporal evolution of the particle velocity can be well predicted by the the curve:

$$U^* = U/U_T = 1 - \exp\left(-\frac{3t}{\tau_{95}}\right), \tag{38}$$

where U^* is the velocity of the particle (U) normalized by its terminal velocity (U_T), τ_{95} is the time it takes for the sphere to reach 95% of its terminal velocity, and t is time. The temporal evolution of the rising/falling spherical particle is compared to this curve in figure 10 showing that the particle velocity is qualitatively similar to the exponential curve. The domain size in simulations is small ($d/L = 0.15$) and thus wall effects become important. This test case confirms the stability and accuracy of the numerical solver when applied to wide range of density ratios in fluid-particle systems. All cases showed relatively close match to the experimental correlation. The heavier than fluid particle trajectories were much closer to the correlation as compared to the lighter-than-fluid cases. For the lighter than fluid particles, the initial acceleration was found to be larger, which is probably because of their small Stokes number. It should be noted that the correlation was developed from experimental data on heavier than fluid particles falling under gravity in a large tank filled with water. The density ratios considered in the experiments by Mordant and Pinton [44] were 2.56 (glass), $7.67 - 7.85$ (steel), and 14.8 (tungsten). The correlation developed was based on data for steel particles. They also showed variations in the correlation for different density particles. Differences in the predicted particle velocities and the experimental data may be attributed to the fact that the density ratios consider in the numerical test cases were varied over a much larger range and by orders of magnitude.

A systematic grid refinement and convergence study was conducted for the case of $\rho_P/\rho_F = 500$. Three grids with uniform resolution of $100 \times 100 \times 160$, $150 \times 150 \times 240$, and $200 \times 200 \times 320$ were used for this analysis. Using these three results, the GCI uncertainty is computed for the particle velocity, and is plotted as error bars on the fine grid data points in figure 11. Near its terminal velocity at $t = 0.2$ s, the GCI indicates a relative uncertainty of 5.3% in the fine grid solution. Similar uncertainties were obtained for lighter-than-fluid particles.

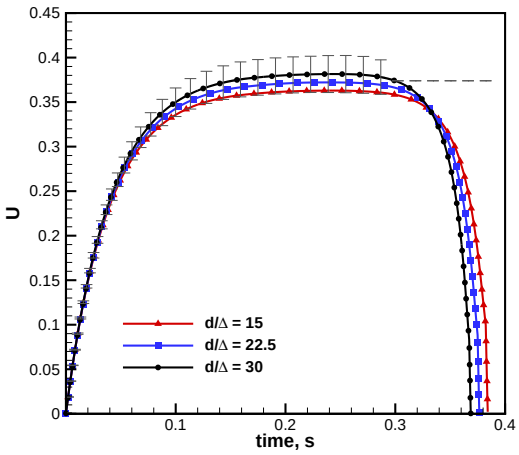


Figure 11: Grid convergence study and uncertainty analysis for falling/rising sphere ($\rho_P/\rho_F = 500$) showing error bars around the solution predicted by the fine resolution.

4.6. Buoyant sphere in a Gaussian vortex

The entrainment of a single buoyant sphere in a stationary Gaussian vortex as shown in figure 12(a) is considered as a final test case. The buoyant sphere is released in the vicinity of the vortex with core radius, r_c, and initial circulation, Γ_0. With sufficient vortex strength, the sphere gets entrained into the vortex after circling around several times and reaching a settling location with relative coordinates (r_s, θ_s) measured from the

vortex center. For the Gaussian vortex, there is no radial velocity component, and the the tangential velocity is expressed as

$$u_\theta(r) = \frac{\Gamma_0}{2\pi r}\left(1 - e^{-\eta_1 (r/r_c)^2}\right). \tag{39}$$

The vorticity and maximum tangential velocity (occurs at $r = r_c$) are given by

$$\omega(r) = \frac{\Gamma_0 \eta_1}{\pi r_c^2} e^{-\eta_1 (r/r_c)^2}; \quad u_c = \eta_2 \frac{\Gamma_0}{2\pi r_c}, \tag{40}$$

where η_1 and η_2 are constants. This flow has been used previously by Oweis *et al.* [45], as a model for wingtip

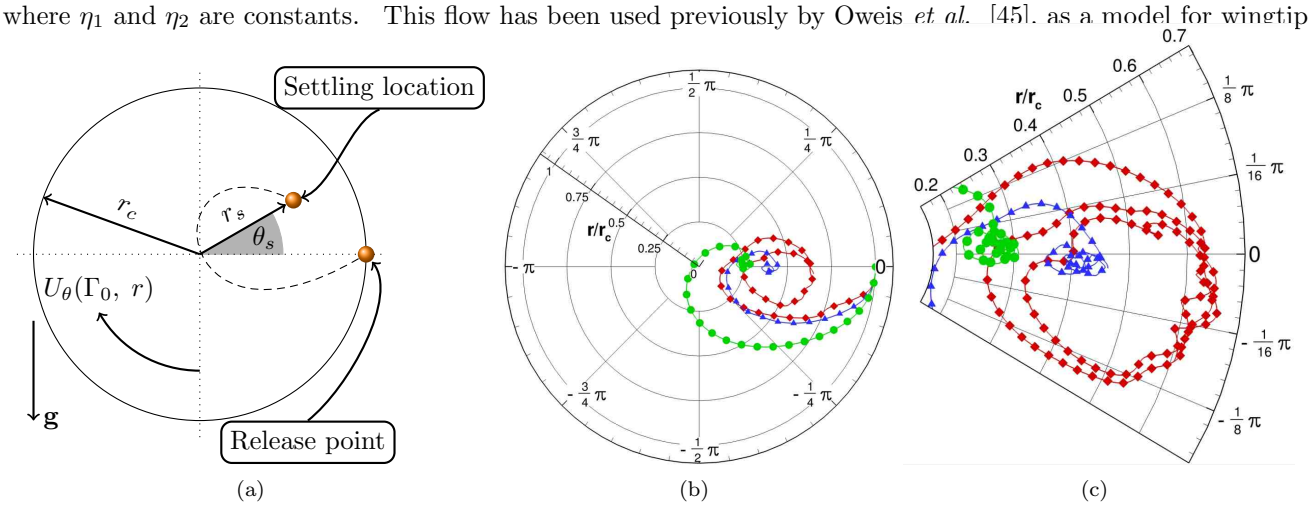
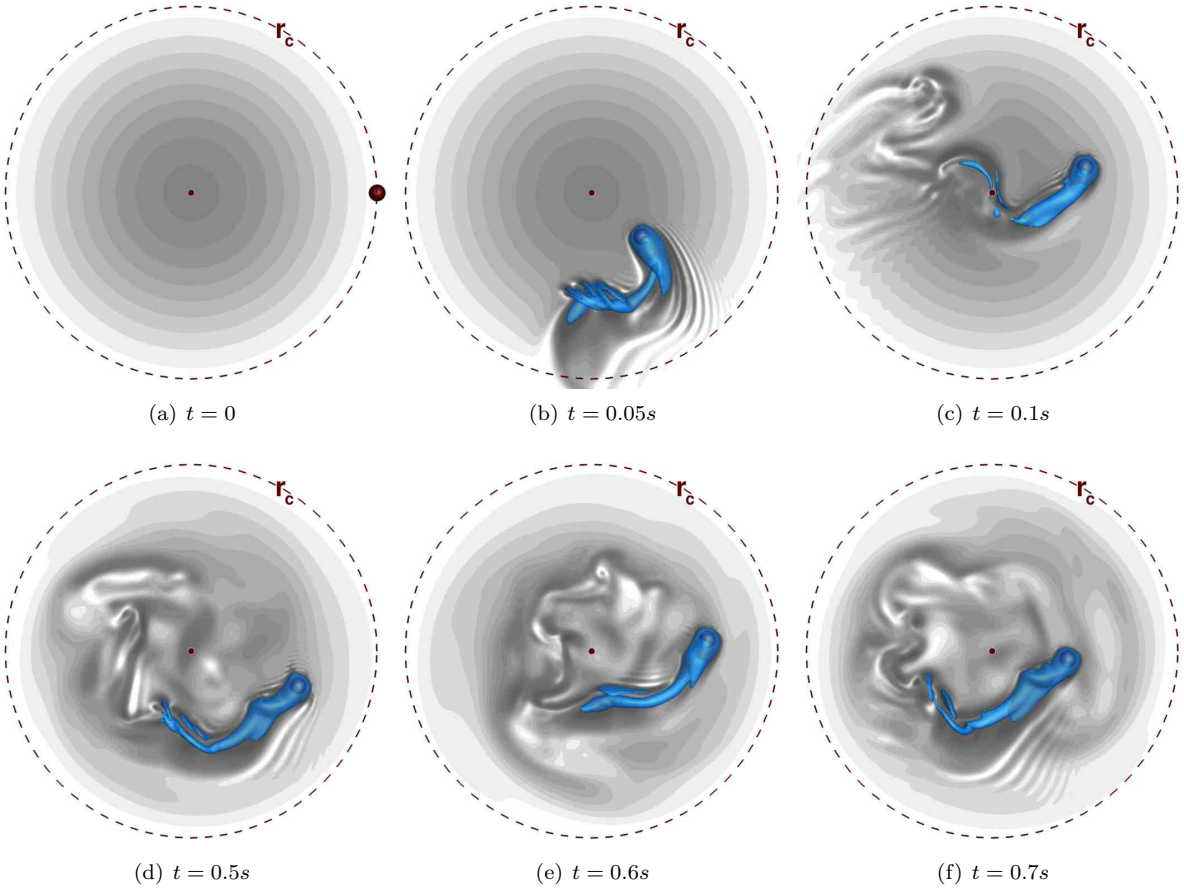


Figure 12: Entrainment of a buoyant sphere by a Gaussian vortex of core radius r_c: (a) Problem schematic, showing the definition of the settling coordinates (r_s, θ_s), (b) Entrainment trajectories for the three cases simulated, (c) The motion of the sphere at the settling location. ● $\rho_F/\rho_P = 100$, ▲ $\rho_F/\rho_P = 500$, ♦ $\rho_F/\rho_P = 1000$.

vortices in their study of bubble capture and cavitation inception. Variables relevant for the setup of this test case are summarized in table 4. The domain size is 7 r_c × 7 r_c × 0.4 r_c. A slip condition is imposed at boundaries in the X and Y directions, and the domain is periodic in the Z direction. The Cartesian grid is refined in the area of the vortex core with a cubic spacing of $\Delta_{core} = 0.1$ mm and has 450 × 450 × 50 grid points in the X, Y and Z directions. The velocity field of equation 39 is applied as an initial condition everywhere in the domain, creating a clockwise vortex with initial strength $\Gamma_0 = 0.04$ m^{2}s^{-1}, which gives the vortex Reynolds number of $Re_{vx} = \rho_F \Gamma_0/\mu_F = 40000$. During the first timestep, a single, 1.1 mm diameter spherical particle is released at $r = r_c$, $\theta = 0$. We assume typical properties of water for the fluid phase and simulate three different particle densities corresponding to $\rho_F/\rho_P = 100$, 500, and 1000. In each case, the simulation is advanced using a timestep of $\Delta t = 50\ \mu s$ up to a time of 1 s. Figure 12(b) shows the entrainment trajectories of the spheres for each

Table 4: Parameters for the Gaussian vortex case.

r_c [mm]	11.45
η_1	1.27
η_2	0.715
$L_x,\ L_y\ L_z$ [mm]	80,80,5
Δ_{core} [mm]	0.1
N_{grid}	425 × 425 × 50
Γ_0 [m^2/s]	0.04
d [mm]	1.1
ρ_F [kg/m^3]	1000
ρ_P [kg/m^3]	1, 2, 10
μ_F[kg/m.s]	0.001



(a) $t = 0$ (b) $t = 0.05s$ (c) $t = 0.1s$

(d) $t = 0.5s$ (e) $t = 0.6s$ (f) $t = 0.7s$

Figure 13: Snapshots of fluid vorticity magnitude in the vortex core for the high density ratio case ($\rho_F/\rho_P = 1000$). Contours are from 20 to 250 m^2/s. Isosurface of 350 m^2/s is also shown.

Table 5: Average settling coordinates of the sphere in the vortex core as shown in figure 12(a).

ρ_F/ρ_P	r_s/r_c	θ_s [rad]
100	0.30	0.005
500	0.45	-0.065
1000	0.57	-0.106

density ratio considered. Because of the high density ratios, the spheres do not follow the fluid streamlines and instead are attracted toward the upper right hand side of the inner core. This is due to the lift and added mass effects effects similarly observed by Mazzitelli *et al.* [46] and Climent *et al.* [47] for bubbles entrained in vortical structures. As density ratio and relative buoyancy force are increased, a more direct path is taken to the settling location by the sphere. The settling coordinates are tabulated in table 5 for each case. With increased density ratio, the spheres reach equilibrium at a larger radius and more negative angle from the horizontal. For ρ_F/ρ_P equal to 100 and 500, there is little relative motion of the sphere once it becomes entrained. At $\rho_F/\rho_P = 1000$, a much different dynamic exists because the sphere does not stay in one location. Rather it circles a point centered at $(r_s/r_c,\ \theta_s) = (0.57,\ -0.106)$. This motion causes a strong and highly unsteady wake to develop behind the sphere as is shown in figure 13. Strong vortical structures are periodically shed from the sphere, and advected clockwise around the vortex core back to the sphere.

By increasing the particle density, ρ_P, from 1 kg/m^3 to just 2 kg/m^3 and holding all other parameters constant, the system transitions from a vortex which is mostly undisturbed by the presence of the entrained particle to one for which the vortex core is significantly distorted by the particle's presence and motion. In density stratified flows the Atwood number, $\frac{\rho_F - \rho_P}{\rho_F + \rho_P}$, is the typical predictor of hydrodynamic instability. However, by increasing ρ_P from 1 to 2, the corresponding change to the Atwood number is just 0.2% in this system. To understand why such a small change in density ratio may result in transition to unsteady flow, a simple analytic model for the settling coordinates of a small passive (no influence on the flow), buoyant particle as a function of ρ_P/ρ_F, d, $u_\theta(r_s)$, $\omega(r_s)$, and the lift, drag, and added mass coefficients of the particle C_L, C_D, and C_{AM} was derived for this same flow by Finn *et al* [48].

$$\cos(\theta_s) = \frac{3C_D u_\theta(r_s)^2}{4dg\left[\frac{\rho_P}{\rho_F} - 1\right]} \tag{41}$$

$$r_s = \frac{(1 + C_{AM})u_\theta^2}{\left[\frac{\rho_P}{\rho_F} - 1\right] g\sin(\theta_s) + C_L u_\theta \omega(r_s)}, \tag{42}$$

where g is the gravitational acceleration. While this model uses estimated drag, lift, and added mass coefficients, it serves to show the general dependence of r_s on ρ_P/ρ_F. For the cases studied, θ_s is small. With small increase in ρ_P/ρ_F, the particle settles further away from the vortex center. A direct consequence of a small increase in r_s is a corresponding increase in the particle Reynolds number. From equation 39, and the settling coordinates in table 4, the particle Reynolds number (assuming a fixed particle) is estimated to increase from roughly 300 to 360. This leads to stronger, more influential vortex shedding and unsteady wake behind the sphere not seen at lower density ratios. The problem of lighter-than-fluid particle interacting with a vortex is of critical significance to several practical applications including drag reduction by microbubbles, and will be part of future studies. The numerical approach developed here facilitates such fundamental studies for a wide range of density ratios.

Conclusion

A numerical formulation for fully resolved simulations of freely moving rigid particles for a wide range of fluid-particle density ratios is developed based on a fictitious domain method. In this approach, the entire computational domain is first treated as a fluid of varying density corresponding to the fluid or particle densities in their respective regions. The incompressibility and rigidity constraints are applied to the fluid and particle regions, respectively, by using a fractional step algorithm. The momentum equations are recast to avoid computations of density variations across the fluid-particle interface. The resultant equations are discretized using symmetric central differences in space and time. A density and area weighted reconstruction of pressure gradients is developed to obtain stable and accurate results for wide range of fluid-particle density ratios. Use of consistent interpolations between the particle material volumes and the background grid and parallel implementation of the algorithm facilitates accurate and efficient simulations of large number of particles. Implementation of this approach in finite-volume, co-located grid based numerical solver is presented. The numerical approach is applied to simulate flow over fixed immersed objects (cylinder and sphere) at different Reynolds numbers. Flow over cylinder case is used to perform a detailed convergence analysis to show good accuracy of the solver. The numerical approach is also applied to freely falling/rising spheres for a wide range of density ratios and compared with published experimental or numerical results to show good agreement. Finally, the approach was applied to simulate freely

moving buoyant spheres in a stationary Gaussian vortex for a range of density ratios to investigate the ability of the solver to capture lift and added-mass effects on lighter-than fluid particles in vortical flows. For large vortex strengths, the lighter-than fluid particles are attracted towards the core of the vortex and settle further away from the center in the presence of gravity as expected.

Acknowledgements

Financial supports from National Science Foundation under the grants NSF-CBET #1133363 and NSF-CBET #0933857 as well as DoE's National Energy Technology Laboratory (URS Contract Number 41817M4077) are highly appreciatied. Simulations were performed on the high performance computing cluster at Oregon State University as well as at TACC's Lonestar.

5. Appendix

The details of particle position update and interpolations between the material volume centroids and the *cv* centers are provided below.

5.1. Particle Position Update

The rigid body motion (RBM) of a particle can be decomposed into translational ($\mathbf{U}^T$) and rotational ($\mathbf{U}^R$) components The total velocity field at each point within the particle is given as

$$\mathbf{U}^{RBM} = \mathbf{U}^T + \mathbf{\Omega} \times \mathbf{r} \tag{43}$$

where $\mathbf{U}^T$ is the translational velocity, $\mathbf{\Omega}$ the angular velocity, and $\mathbf{r}$ the position vector of the material volume centroid with respect to the particle centroid. All the material volumes have the same translational velocity as the particle centroid ($\mathbf{U}^T = \mathbf{U}_P$).

Given a velocity field and the positions ($\mathbf{X}_M^0$) of the material volume centroids and the particle centroid ($\mathbf{X}_P$) at $t = t_0$, the new positions ($\mathbf{X}_M^t$) at $t = t_0 + \Delta t$ are obtained by linear superposition of the rotational and translational components of the velocity. The axis of rotation passing through the rigid body centroid $\mathbf{X}_P$ is given as $\hat{\sigma} = \mathbf{\Omega}/\left|\mathbf{\Omega}\right|$. The new coordinates due to rotation around $\hat{\sigma}$ are given as

$$\mathbf{X}' = \mathcal{R}(\mathbf{X}_M^0 - \mathbf{X}_P) + \mathbf{X}_P \tag{44}$$

where the rotation matrix is

$$\mathcal{R} = \begin{bmatrix} t\hat{\sigma}_x\hat{\sigma}_x + c & t\hat{\sigma}_x\hat{\sigma}_y - s\hat{\sigma}_z & t\hat{\sigma}_x\hat{\sigma}_z + s\hat{\sigma}_y \\ t\hat{\sigma}_x\hat{\sigma}_y + s\hat{\sigma}_z & t\hat{\sigma}_y\hat{\sigma}_y + c & t\hat{\sigma}_y\hat{\sigma}_z - s\hat{\sigma}_x \\ t\hat{\sigma}_x\hat{\sigma}_z - s\hat{\sigma}_y & t\hat{\sigma}_y\hat{\sigma}_z + s\hat{\sigma}_x & t\hat{\sigma}_z\hat{\sigma}_z + c \end{bmatrix}. \tag{45}$$

Here $c = \cos(\alpha)$, $s = \sin(\alpha)$, $t = 1 - \cos(\alpha)$, and $\alpha = |\mathbf{\Omega}|\Delta t$. The material volume centroids are all uniformly translated to give the final positions,

$$\mathbf{X}_M^t = \mathbf{X}' + \mathbf{U}^T \Delta t. \tag{46}$$

5.2. Interpolation Between the Two Phases

Any property defined at the material volumes within the particle can be projected onto the background grid by using interpolation functions. Use of simple linear interpolations may give rise to unphysical values within the particle domain (e.g. volume fractions greater than unity) [1] and may give rise to numerical oscillations in the particle velocity. In order to overcome this, a smooth approximation of the quantity can be constructed from the material volumes using better interpolation kernels:

$$\Phi^{\Delta}(\mathbf{x}) = \int \Phi(\mathbf{y})\xi^{\Delta}(\mathbf{x} - \mathbf{y})d\mathbf{y} \tag{47}$$

where Δ denotes grid resolution. The interpolation operator can be discretized using the material volume centroids as the quadrature points to give

$$\Phi^{\Delta}(\mathbf{x}) = \sum_{M=1}^{N} V_M \Phi(\mathbf{X}_M)\xi^{\Delta}(\mathbf{x} - \mathbf{X}_M) \tag{48}$$

where $\mathbf{X}_M$ and V_M denote the coordinates and volume of the material volumes respectively and the summation is over all material volumes for a particle. For example, in order to compute particle volume fraction, $\Phi(\mathbf{X}_M)$ will be unity at all material points. This gives unity volume fraction within the particle domain and zero outside the particle. In order to conserve the total volume of the particle as well as the total force/torque exerted by the particle on the fluid, the interpolation kernel should at least satisfy

$$\sum_{M=1}^{N} V_M \xi^{\Delta}(\mathbf{x} - \mathbf{X}_M) = 1 \tag{49}$$

$$\sum_{M=1}^{N} V_M (\mathbf{x} - \mathbf{X}_M) \xi^{\Delta}(\mathbf{x} - \mathbf{X}_M) = 0 \tag{50}$$

Several kernels with second-order accuracy include Gaussian, quartic splines etc. A kernel with compact support requiring only the immediate neighbors of a control volume has been designed and used in immersed boundary methods [15]. For uniform meshes with resolution Δ it utilizes only three points in one dimension and gives the sharpest representation of the particle onto the background mesh:

$$\xi^{\Delta}(\mathbf{x} - \mathbf{X}_M) = \frac{1}{\Delta^3} \delta\left(\frac{x - X_M}{\Delta}\right) \delta\left(\frac{y - Y_M}{\Delta}\right) \delta\left(\frac{z - Z_M}{\Delta}\right), \tag{51}$$

where

$$\delta(r) = \begin{cases} \frac{1}{6}(5 - 3|r| - \sqrt{-3(1-|r|)^2 + 1}, & 0.5 \le |r| \le 1.5, r = \frac{(x - x_0)}{\Delta} \\ \frac{1}{3}(1 + \sqrt{-3r^2 + 1}, & |r| \le 0.5 \\ 0, & \text{otherwise.} \end{cases} \tag{52}$$

Accordingly, in section 3 the velocity field at the *cv* centers is interpolated to the material volume centers and is given by,

$$\widehat{U}_{i,M}(\mathbf{X}_M) = \sum_{\text{nbr cvs of M}} V_{cv} \widehat{u}_{i,\text{cv}}(\mathbf{x}_{\text{cv}}) \xi^{\Delta}(\mathbf{x}_{\text{cv}} - \mathbf{X}_M), \tag{53}$$

where M stands for the material volume, the summation is over all *cv*s in the neighborhood of the control volume that contains the material volume centroid (that is total of 27 cvs in three-dimensions), V_{cv} is the volume of the cv, and ξ^{Δ} is the interpolation kernel given by equation 51. Likewise, the same interpolation kernel can be used to interpolate an Eulerian quantity defined at the grid centroids to the material volume centroids. The force on the grid control volumes ($f_{i,\text{cv}}$) is obtained from $\text{F}_{i,M}$ through a consistent interpolation scheme with the same kernel as used for interpolating the *cv*-center values to the material volume centroids and is given as,

$$f_{i,\text{cv}}(\mathbf{x}_{\text{cv}}) = \sum_{\text{all}M} V_M \text{F}_{i,M}(\mathbf{X}_M) \xi^{\Delta}(\mathbf{x}_{\text{cv}} - \mathbf{X}_M), \tag{54}$$

where the summation is over all material volumes within a rigid body that are in the neighborhood of $\mathbf{x}_{cv}$ and V_M is the volume of each material point.

References

[1] N. Sharma, N. Patankar, A fast computation technique for the direct numerical simulation of rigid particulate flows, Journal of Computational Physics 205 (2) (2005) 439–457.

[2] G. Muldowney, J. Higdon, A spectral boundary element approach to three-dimensional stokes flow, Journal of Fluid Mechanics 298 (1995) 167–192.

[3] H. Hu, N. Patankar, M. Zhu, Direct numerical simulations of fluid-solid systems using the arbitrary Lagrangian-Eulerian fechnique, Journal of Computational Physics 169 (2) (2001) 427–462.

[4] A. Johnson, T. Tezduyar, Simulation of multiple spheres falling in a liquid-filled tube, Computer Methods in Applied Mechanics and Engineering 134 (3-4) (1996) 351–373.

[5] J. Monaghan, Smoothed particle hydrodynamics, Reports on Progress in Physics 68 (8) (2005) 1703–1759.

[6] R. Glowinski, T. Pan, T. Hesla, D. Joseph, J. Periaux, A fictitious domain approach to the direct numerical simulation of incompressible viscous flow past moving rigid bodies- application to particulate flow, Journal of Computational Physics 169 (2) (2001) 363–426.

[7] C. Peskin, The immersed boundary method, Acta Numerica 11 (2003) 479–517.

[8] R. Mittal, G. Iaccarino, Immersed boundary methods, Annual Review of Fluid Mechanics 37 (1) (2005) 239–261.

[9] M. Uhlmann, An immersed boundary method with direct forcing for the simulation of particulate flows, Journal of Computational Physics 209 (2) (2005) 448–476.

[10] D. Kim, H. Choi, Immersed boundary method for flow around an arbitrarily moving body, Journal of Computational Physics 212 (2) (2006) 662–680.

[11] R. Mittal, H. Dong, M. Bozkurttas, F. Najjar, A. Vargas, A. Von Loebbecke, A versatile sharp interface immersed boundary method for incompressible flows with complex boundaries, Journal of computational physics 227 (10) (2008) 4825–4852.

[12] A. Ladd, R. Verberg, Lattice-Boltzmann simulations of particle-fluid suspensions, J. Statist. Phys. 104 (2001) 1191–1251.

[13] Z. Feng, E. Michaelides, Proteus: a direct forcing method in the simulations of particulate flows, Journal of Computational Physics 202 (1) (2005) 20–51.

[14] Z. Zhang, A. Prosperetti, A second-order method for three-dimensional particle simulation, Journal of Computational Physics 210 (1) (2005) 292–324.

[15] A. Roma, C. Peskin, M. Berger, An Adaptive Version of the Immersed Boundary Method, Journal of Computational Physics 153 (2) (1999) 509–534.

[16] D. Goldstein, R. Handler, L. Sirovich, Modeling a no-slip flow boundary with an external force field, Journal of Computational Physics 105 (2) (1993) 354–366.

[17] H. Udaykumar, R. Mittal, P. Rampunggoon, A. Khanna, A sharp interface cartesian grid method for simulating flows with complex moving boundaries, Journal of Computational Physics 174 (1) (2001) 345–380.

[18] F. Gibou, R. Fedkiw, L. Cheng, M. Kang, A second-order-accurate symmetric discretization of the poisson equation on irregular domains, Journal of Computational Physics 176 (1) (2002) 205–227.

[19] J. Kim, D. Kim, H. Choi, An immersed-boundary finite-volume method for simulations of flow in complex geometries, Journal of Computational Physics 171 (1) (2001) 132–150.

[20] E. Fadlun, R. Verzicco, P. Orlandi, J. Mohd-Yusof, Combined immersed-boundary finite-difference methods for three-dimensional complex flow simulations, Journal of Computational Physics 161 (1) (2000) 35–60.

[21] T. Kempe, J. Fröhlich, An improved immersed boundary method with direct forcing for the simulation of particle laden flows, Journal of Computational Physics 231 (9) (2012) 3663 – 3684.

[22] K. Taira, T. Colonius, The immersed boundary method: A projection approach, Journal of Computational Physics 225 (2) (2007) 2118–2137.

[23] N. Patankar, P. Singh, D. Joseph, R. Glowinski, T. Pan, A new formulation of the distributed Lagrange multiplier/fictitious domain method for particulate flows, International Journal of Multiphase Flow 26 (9) (2000) 1509–1524.

[24] C. Veeramani, P. Minev, K. Nandakumar, A fictitious domain formulation for flows with rigid particles: A non-Lagrange multiplier version, Journal of Computational Physics 224 (2) (2007) 867–879.

[25] S. Apte, M. Martin, N. Patankar, A numerical method for fully resolved simulation (FRS) of rigid particle-flow interactions in complex flows, Journal of Computational Physics 228 (8) (2009) 2712–2738.

[26] M. Rudman, A volume-tracking method for incompressible multifluid flows with large density variations, International Journal for Numerical Methods in Fluids 28 (2) (1998) 357–378.

[27] M. Bussmann, D. Kothe, J. Sicilian, Modeling high density ratio incompressible interfacial flows, in: Proc. FEDSM, Vol. 2, 2002, pp. 2002–31125.

[28] M. Raessi, A level set based method for calculating flux densities in two-phase flows, Annual Research Briefs (2008) 467–478.

[29] S. Osher, R. Fedkiw, Level set methods and dynamic implicit surfaces, Springer Verlag, 2003.

[30] S. Osher, J. Sethian, Fronts propagating with curvature-dependent speed: algorithms based on Hamilton-Jacobi formulations, Journal of computational physics 79 (1) (1988) 12–49.

[31] D. Kim, H. Choi, A second-order time-accurate finite volume method for unsteady incompressible flow on hybrid unstructured grids, Journal of Computational Physics 162 (2) (2000) 411–428.

[32] K. Mahesh, G. Constantinescu, S. Apte, G. Iaccarino, F. Ham, P. Moin, Large-Eddy Simulation of Reacting Turbulent Flows in Complex Geometries, Journal of Applied Mechanics 73 (2006) 374.

[33] S. Patankar, Numerical heat transfer and fluid flow, Taylor & Francis, 1980.

[34] H. van der Vorst, Iterative Krylov Methods for Large Linear Systems, Cambridge University Press, 2003.

[35] P. Roache, Perspective: a method for uniform reporting of grid refinement studies, Journal of fluids engineering 116 (3) (1994) 405–413.

[36] J. Cadafalch, C. Pérez-Segarra, R. Consul, A. Oliva, Verification of finite volume computations on steady-state fluid flow and heat transfer, Journal of Fluids Engineering 124 (2002) 11.

[37] I. Celik, U. Ghia, P. Roache, et al., Procedure for estimation and reporting of uncertainty due to discretization in {CFD} applications, Journal of fluids Engineering-Transactions of the ASME 130 (7).

[38] S. Marella, S. Krishnan, H. Liu, H. Udaykumar, Sharp interface cartesian grid method i: An easily implemented technique for 3d moving boundary computations, Journal of Computational Physics 210 (1) (2005) 1–31.

[39] R. Mittal, A fourier-chebyshev spectral collocation method for simulating flow past spheres and spheroids, International Journal for Numerical Methods in Fluids 30 (7) (1999) 921–937.

[40] R. Clift, J. Grace, M. Weber, Bubbles, drops, and particles, Academic Press, New York, 1978.

[41] T. Johnson, V. Patel, Flow past a sphere up to a reynolds number of 300, Journal of Fluid Mechanics 378 (1999) 19–70.

[42] A. ten Cate, C. Nieuwstad, J. Derksen, H. Van den Akker, Particle imaging velocimetry experiments and lattice-Boltzmann simulations on a single sphere settling under gravity, Physics of Fluids 14 (2002) 4012.

[43] S. Lomholt, B. Stenum, M. Maxey, Experimental verification of the force coupling method for particulate flows, International Journal of Multiphase Flow 28 (2) (2002) 225–246.

[44] N. Mordant, J. Pinton, Velocity measurement of a settling sphere, The European Physical Journal B 18 (2) (2000) 343–352.

[45] G. Oweis, I. van der Hout, C. Iyer, G. Tryggvason, S. Ceccio, Capture and inception of bubbles near line vortices, Physics of Fluids 17 (2005) 022105.

[46] I. Mazzitelli, D. Lohse, F. Toschi, On the relevance of the lift force in bubbly turbulence, Journal of Fluid Mechanics 488 (2003) 283–313.

[47] E. Climent, M. Simonnet, J. Magnaudet, Preferential accumulation of bubbles in Couette-Taylor flow patterns, Physics of Fluids 19 (2007) 083301.

[48] J. Finn, E. Shams, S. Apte, Modeling and simulation of multiple bubble entrainment and interactions with two dimensional vortical flows, Physics of Fluids 23 (2) (2011) 023301–023301.