
Performance of various fluid-solid coupling methods for DNS of particulate flow

Markus Uhlmann and Alfredo Pinelli

Departamento de Combustibles Fósiles
CIEMAT
Avenida Complutense 22, 28040 Madrid, Spain
`markus.uhlmann@ciemat.es`

1 Introduction

In the present article we are concerned with efficient and accurate methods for the numerical simulation of the dynamics of rigid particles suspended in an incompressible fluid. We focus our attention on so-called fictitious domain methods, a framework in which the presence of suspended particles is accounted for by means of an artificial forcing term added to the Navier-Stokes equations. Thereby, simple fixed grids can be used and the additional cost of repeated adaptation of the computational mesh is avoided.

Existing fictitious domain methods fall into two main classes: those where the constraint force is explicitly formulated (“direct forcing”) and those where some kind of feedback mechanism is employed (“indirect forcing”). The latter type of formulation is usually based upon the virtual spring-and-damper technique of Goldstein et al. [1], which has the drawback of introducing two additional free parameters into the problem. Also, since the characteristic time scale of the feedback system needs to be resolved, very small time steps are sometimes required for stability reasons.

Direct forcing methods, on the other hand, are in general free from the above mentioned problems. However, it has been observed [2] that a linear interpolation procedure (as used in references [3, 4]) can lead to highly oscillatory hydrodynamic forces when a solid immersed body is in motion relative to the fixed grid.

In reference [5] the artificial force term is instead weighted by the solid fraction of the surrounding grid cell, providing some amount of smoothing. However, the resulting drag/lift variation obtained in our computations with this method still shows substantial grid-induced oscillations (see results in § 3.1).

In Peskin’s immersed boundary method [6]—which originally uses indirect forcing—quantities are transferred between arbitrary (Lagrangian) positions

and the fixed (Eulerian) grid by means of a so-called “regularized delta function” with desirable smoothness properties and compact support.

Recently, a new direct forcing scheme which makes use of Peskin’s delta function during the necessary interpolation steps was formulated in reference [7]. In the following we will discuss this method and its validation through a selection of test cases. Finally we will present preliminary results from the simulation of the sedimentation of several hundred spherical particles.

2 Numerical method

Let us write the time-discretized momentum equation in the following form

$$\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} = \mathbf{rhs}^{n+1/2} + \mathbf{f}^{n+1/2}, \quad (1)$$

where $\mathbf{rhs}^{n+1/2}$ regroups all usual forces (convective, pressure-related, viscous) and $\mathbf{f}^{n+1/2}$ is an artificial force term, both evaluated at some intermediate time level. Since Fadlun *et al.* [3] it is common to express the additional force term by simply rewriting the above equation as

$$\mathbf{f}^{n+1/2} = \frac{\mathbf{u}^{(d)} - \mathbf{u}^n}{\Delta t} - \mathbf{rhs}^{n+1/2} \quad (2)$$

where $\mathbf{u}^{(d)}$ is the desired velocity at the point where forcing is to be applied. Formula (2) is characteristic for direct forcing methods. Problems arise from the fact that in general the locations where the desired velocity is known do not coincide with the Eulerian grid. In [7] the definition of the force term was instead formulated at Lagrangian positions attached to the surface of the particles, viz.

$$\mathbf{F}^{n+1/2} = \frac{\mathbf{U}^{(d)} - \mathbf{U}^n}{\Delta t} - \mathbf{RHS}^{n+1/2}, \quad (3)$$

where upper-case letters indicate quantities evaluated at Lagrangian coordinates. Obviously, the velocity in the particle domain \mathcal{S} is simply given by the solid-body motion,

$$\mathbf{U}^{(d)}(\mathbf{X}) = \mathbf{u}_c + \boldsymbol{\omega}_c \times (\mathbf{X} - \mathbf{x}_c) \quad \mathbf{X} \in \mathcal{S}, \quad (4)$$

as a function of the translational and rotational velocities of the particle, \mathbf{u}_c , $\boldsymbol{\omega}_c$.

The final element of the method of reference [7] is the transfer of the velocity (and r.h.s. forces) from Eulerian to Lagrangian positions as well as the inverse transfer of the forcing term to the Eulerian grid positions. For this purpose we define a Cartesian grid \mathbf{x}_{ijk} with uniform mesh width h in all three directions and distribute so-called discrete Lagrangian force points \mathbf{X}_l

evenly on the particle surface. Using Peskin's [6] regularized delta function formalism, the transfer can be written as:

$$\mathbf{U}(\mathbf{X}_l) = \sum_{ijk} \mathbf{u}(\mathbf{x}_{ijk}) \delta_h(\mathbf{x}_{ijk} - \mathbf{X}_l) h^3, \quad (5a)$$

$$\mathbf{f}(\mathbf{x}_{ijk}) = \sum_l \mathbf{F}(\mathbf{X}_l) \delta_h(\mathbf{x}_{ijk} - \mathbf{X}_l) \Delta V, \quad (5b)$$

where ΔV_l designates the forcing volume assigned to the l th force point. We use the particular function δ_h given in [8] which has the properties of continuous differentiability, second order accuracy, support of three grid nodes in each direction and consistency with basic properties of the continuous delta function.

The algorithm for each time-step can then be summed up as follows:

1. compute $\tilde{\mathbf{u}} = \mathbf{u}^n + \mathbf{r}\mathbf{h}s^{n+1/2}$
2. transfer $\tilde{\mathbf{u}}$ to Lagrangian positions, using (5a)
3. compute $\mathbf{F}(\mathbf{X}_l)$ from (3)
4. transfer the force back to Eulerian positions, using (5b)
5. solve Navier-Stokes on the fixed grid with the added force term $\mathbf{f}(\mathbf{x}_{ijk})$.

The above method has been implemented in a staggered finite-difference context, involving central, second-order accurate spatial operators, an implicit treatment of the viscous terms and a three-step Runge-Kutta procedure for the non-linear part. Continuity in the entire domain is enforced by means of a projection method. The particle motion is determined by the Runge-Kutta-discretized Newton equations for rigid-body motion, which are weakly coupled to the fluid equations. In the present simulations direct particle interactions (collisions) are not considered.

3 Results

3.1 Uniform flow around an oscillating cylinder

In this first test case the particle motion is prescribed, i.e. one-way coupled. We consider the flow around a cylinder with diameter D located at the origin in a domain which measures $\Omega_1 = [-6.17, 20.5]D \times [-13.33, 13.33]D$. The uniform grid has 1024×1024 nodes, i.e. $D/h = 38.4$. The time step was set to $\Delta t = 0.003$, corresponding to a maximum CFL number of approximately 0.6. The cylinder follows a prescribed periodic motion perpendicular to the mean flow, i.e.:

$$y_c(t) = A \sin(2\pi f_f t), \quad (6)$$

with the amplitude set to $A = 0.2D$ and the frequency $f_f/f_n = 0.8$, where f_n is the natural shedding frequency obtained from the value of the Strouhal number from the literature: $St = f_n D/u_\infty = 0.195$ (for $Re_D = 185$). This

case corresponds to one of the cases simulated in reference [9]. The maximum velocity of the cylinder is $\max(|\mathbf{u}_c|)/u_\infty = 2\pi f_f A/u_\infty = 0.196$. The boundary conditions are: uniform velocity at the inflow and along the top and bottom boundaries; convective condition at the outflow.

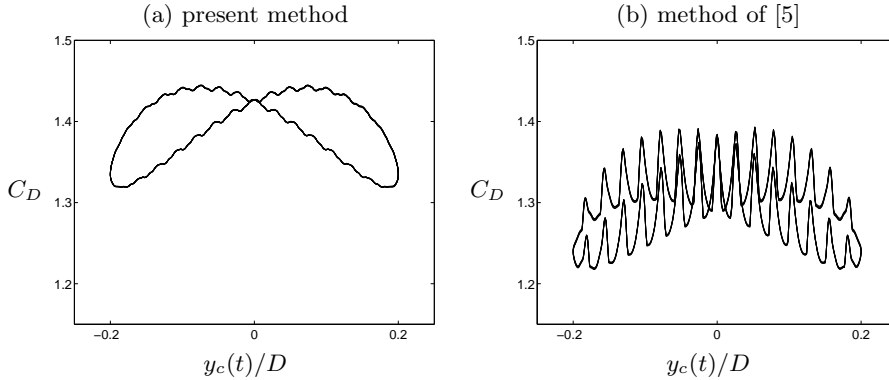


Fig. 1. Time-periodic variation of the drag coefficient in the case of a translationally oscillating cylinder in uniform cross-flow at $Re_D = 185$ with $D/h = 38.4$ and $CFL \approx 0.6$. Left graph: present method. Right graph: method of reference [5], implemented into the present solver as described in [2].

Figure 1 shows that the temporal variation of the drag follows a reasonably smooth periodic curve when using the current scheme. The same goes for the lift force which has been omitted. On the other hand, the method of Kajishima and Takiguchi [5], implemented as shown in [2], yields significant oscillations on the time-scale of the mesh-width divided by the cylinder velocity. In other words, the smoothing provided by the present method proves more efficient in hiding the influence of the fixed grid.

The mean values and fluctuations of drag and lift are given in table 1. The mean drag is over-predicted by approximately 10% with the current scheme. A similar over-prediction was noted in reference [10] where the original immersed boundary method was used for the prediction of the flow around a stationary cylinder on the same grid. In the latter reference the over-prediction was attributed to an insufficient domain size. Here we verify this argument by repeating the simulation in an enlarged domain $\Omega_2 = 1.5\Omega_1$, while maintaining the mesh width and time step. The effect is that indeed both mean drag and lift fluctuations decrease, yielding an error of approximately 8% in the larger domain.

3.2 Sedimentation of a single sphere

We consider a single sphere which is released from rest at $t=0$ in a quiescent fluid. The physical parameters of the simulation are chosen in order to match

	\bar{C}_D	C'_D	$(C_L)_{rms}$
present	1.380 ± 0.063		0.176
present, enlarged domain Ω_2	1.354 ± 0.065		0.166
Kajishima & Takiguchi's scheme [5]	1.282 ± 0.088		0.223
Lu and Dalton [9]	1.25		0.18

Table 1. Dimensionless coefficients obtained from the simulation of the flow around a cylinder at $Re_D = 185$ which oscillates near the natural shedding frequency and using $D/h = 38.4$ and $\Delta t = 0.003$. The domain Ω_1 has been used, except where otherwise stated.

cases 1,2,4 of the experiment of Mordant and Pinton [11], where the motion of spherical beads in water was investigated, while their material and diameter were varied from case to case. The experiment takes place in a large container, justifying the use of periodic conditions in the simulation. By similarity with the experiment (density ratio, Froude number, particle Reynolds number) we have selected the values for the particle diameter $D = 1/6$ and the gravitational acceleration $|\mathbf{g}| = 9.81$ alongside the parameters given in table 2. The range of Reynolds numbers spans $40 \dots 360$ and the density ratio is varied between 2.56 and 7.71. The values for the numerical parameters are: mesh width $h = 1/76.8$, i.e. $D/h = 12.8$; time step $\Delta t = 0.0025$, i.e. yielding a maximum CFL number of 0.3, 0.75, 0.5, respectively.

Figure 2 shows the vertical particle velocity as a function of the elapsed time. Gravitational scaling is used, i.e. $u_{ref} = \sqrt{|\mathbf{g}|D}$ and $t_{ref} = \sqrt{D/|\mathbf{g}|}$ are the reference velocity and time, respectively. The computational results are shown for times before the particle motion in the periodic domain is affected by the remnants of its own wake. A very good agreement with the experimental measurements can be observed. In table 2 the terminal value of the Reynolds number is reported for all three cases. It can be seen that the maximum error is below 2% (case 2).

case	$\frac{\rho_p}{\rho_f}$	$\nu \times 10^3$	Re_D exp. [11]	Re_D present
1	2.56	5.41637	41.17	41.12
2	2.56	1.04238	362.70	366.69
4	7.71	2.67626	280.42	282.45

Table 2. Parameters of the experiment of reference [11] and resulting terminal particle Reynolds number Re_D in the case of a single sedimenting sphere.

3.3 Many-particle sedimentation

Here we consider a similar case as the one studied in reference [5]. A large number of particles are sedimenting in a tri-periodic domain. The dynamic

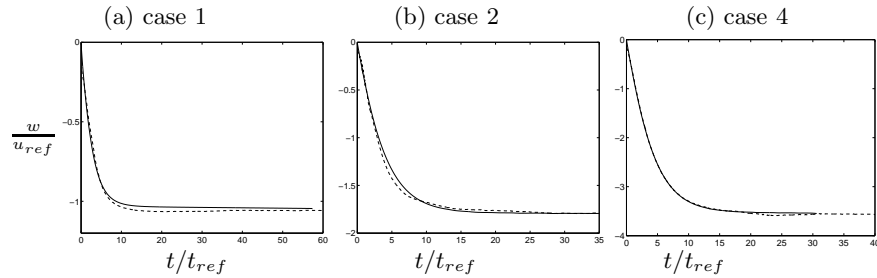


Fig. 2. Sedimentation of a single sphere corresponding to reference [11]. Vertical velocity: — present, ---- experimental data.

formation of particle agglomerations is the object of the investigation in [5]. In the present study, all particles have a density ratio $\rho_p/\rho_f = 2.56$ and a diameter $D = 1/6$; the fluid has a viscosity value of $\nu = 10^{-3}$ and the gravitational acceleration measures $|\mathbf{g}| = 9.81$. This leads to a terminal Reynolds number of approximately 400 for a single sphere, similar to case 2 in § 3.2. Our series of simulations have only been initialized recently and, therefore, the present results only show the behavior of the systems for early times.

Table 3 gives the details of the two configurations presently studied. The volume fraction is kept below 1%, meaning that the dilute regime is addressed. Figure 3 shows the initial particle positions and the configuration after 100 time units (gravitational scaling) for case A. At that time the particles have reached a seemingly disordered state with an inhomogeneous spatial distribution. Statistically, this means that the range of inter-particle distances changes. Most prominently, the global minimum of that distance rapidly approaches the limit of one particle diameter (cf. figure 4). In fact, since we do not use any explicit collision strategy, the distance can drop below this limit and cause non-physical overlap. When this occurred we have stopped the simulation. Figure 4 also shows that the average distance to the nearest particle neighbor decreases significantly from the initial homogeneous state. From figure 5 we can see that during the initial phase the average sedimentation velocity reaches a minimum (where $\bar{w}/u_{ref} \approx 2.6$) and then levels out to approximately $\bar{w}/u_{ref} = 2$. This is a manifestation of a strong wake-sheltering effect as already observed in [5]. Obviously, the minimum is not observed in the case of a single sedimenting sphere. Figure 5 also shows the r.m.s. values of the angular particle velocity. It is interesting to note that the values for rotation vectors in the horizontal plane are by a factor 6 higher than those in the vertical direction.

case	N_p	ϵ_p	L_x/D	L_y/D	L_z/D
A	512	0.4%	40	20	80
B	1000	0.8%	40	40	40

Table 3. Definitions for the two different configurations used in triply-periodic many-particle simulations in § 3.3, listing the number of particles N_p , the volume fraction of solid ϵ_p and the domain size in the three coordinate directions L_i (gravity acts in the negative z -direction).

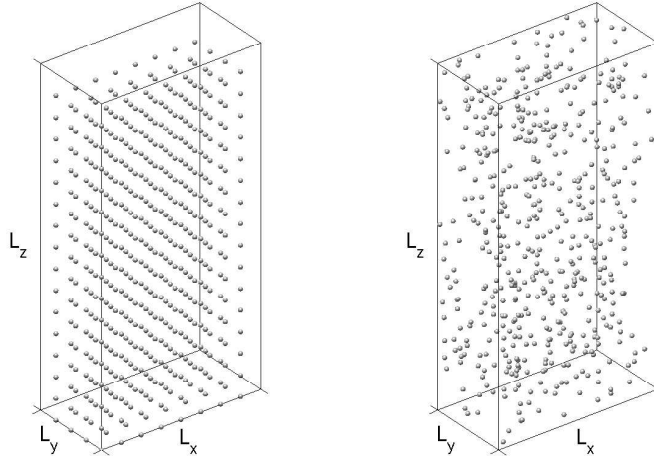


Fig. 3. Particle positions during the simulation of many-particle sedimentation, case A: $t = 0$ (left graph); $t/t_{ref} = 100$ (right graph).

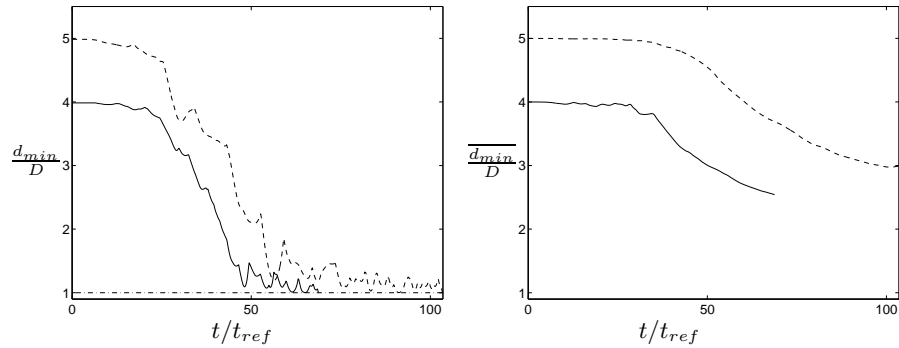


Fig. 4. Statistical particle-related quantities during the simulation of many-particle sedimentation. The left graph shows the minimum inter-particle distance for case A (----), case B (—). The graph on the right shows the average distance to the nearest neighbor for the two cases.

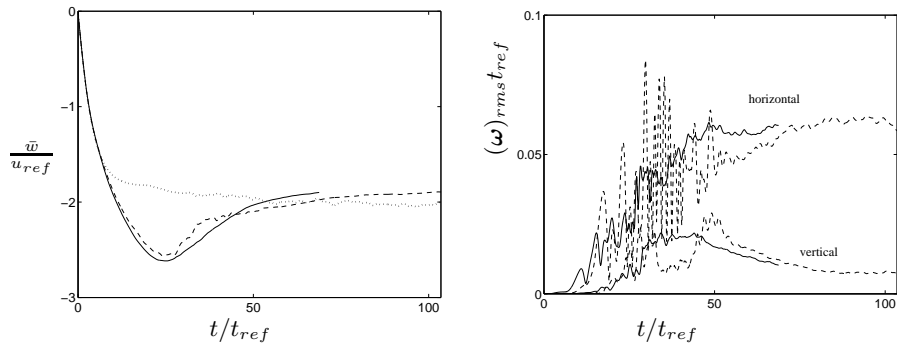


Fig. 5. Statistical particle-related quantities during the simulation of many-particle sedimentation. The left graph shows the mean sedimentation velocity for case A (-----), case B (————) and a single sphere (······). The graph on the right shows the root-mean-square angular particle velocities in the horizontal plane and the vertical direction for the former two cases. In both graphs gravitational scaling is used.

3.4 Efficiency of the method

Operation count.

The following numbers refer to the operations carried out during one Runge-Kutta sub-step, of which there are three per full time step. The main work in the pure fluid part of the code is done while solving the Helmholtz problems during the prediction step and when solving the Poisson problem of the projection step. Using a multi-grid method, the number of operations scales as $\mathcal{O}(N_x N_y N_z)$. On the other hand, the particle-related work scales as $\mathcal{O}\left(N_p \cdot \left(\frac{D}{h}\right)^2\right)$, i.e. linear with the number of particles (since we neglect collisions) and with the square of the number of grid points per diameter (since we only force the surface of the particle).

Time step.

It was observed that the present method does not have a noticeable influence upon the theoretical temporal stability limit, $CFL < \sqrt{3}$.

Parallelization.

Standard domain decomposition over a three-dimensional Cartesian processor grid was used for the fluid solver. The particle-related operations are performed by a master processor who is responsible for all particles contained in its private sub-volume of the domain. Particles overlapping more than one sub-volume are handled by one or more slave processors.

Timing.

Table 4 shows some execution times per full time step for the present scheme. Good scaling with the number of processors can be observed. Also, it becomes clear that the particle-related work makes up only a small fraction of the total execution time for the problems under consideration.

$N_x \times N_y \times N_z$	N_p	nproc	$t_{exec}[s]$
$512 \times 512 \times 512$	1000	64	115.0
$512 \times 512 \times 1024$	1000	128	144.9
$512 \times 512 \times 1024$	2000	128	147.4

Table 4. Execution times on an IBM Cluster with Power 4 processors at 1.1GHz (64bit arithmetic), using different grid sizes $N_x \times N_y \times N_z$, numbers of particles N_p and numbers of processors **nproc**. The resolution of the spherical particles was set to $D/h = 12.8$ in all cases.

4 Conclusions

We have discussed the problems associated with fictitious domain methods of the direct and indirect type, presenting a recently proposed variant which uses the regularized delta function of Peskin and co-workers [6, 10, 8] for the association between arbitrary Lagrangian and discrete Eulerian positions. Thereby, the hydrodynamic forces acting upon the solid domains, which are at the same time driving the particle motion, are free from significant oscillations. This effect was demonstrated for the flow around an oscillating cylinder.

The comparison of the new scheme with well-established experimental results for the sedimentation of a spherical particle shows its accuracy over a significant range of Reynolds numbers while using a very realistic resolution of only 13 grid points per diameter.

Taking into account the fast execution speed of the current method we can conclude that it is indeed very competitive. Further work—which is currently underway—will include the performance of simulations with $\mathcal{O}(10000)$ particles in larger domains. The evaluation of numerical collision strategies will be an important issue which needs to be addressed in detail.

Acknowledgements

This work was supported by the Spanish Ministry of Education and Science under the Ramón y Cajal program (contract DPI-2002-040550-C07-04) and through grant DPI-2002-1314-C07-04.

References

1. D. Goldstein, R. Handler, and L. Sirovich. Modeling a no-slip boundary with an external force field. *J. Comput. Phys.*, 105:354–366, 1993.
2. M. Uhlmann. First experiments with the simulation of particulate flows. Technical Report No. 1020, CIEMAT, Madrid, Spain, 2003. ISSN 1135-9420.
3. E.A. Fadlun, R. Verzicco, P. Orlandi, and J. Mohd-Yusof. Combined immersed-boundary finite-difference methods for three-dimensional complex flow simulations. *J. Comput. Phys.*, 161:35–60, 2000.
4. J. Kim, D. Kim, and H. Choi. An immersed-boundary finite-volume method for simulations of flow in complex geometries. *J. Comput. Phys.*, 171:132–150, 2001.
5. T. Kajishima and S. Takiguchi. Interaction between particle clusters and particle-induced turbulence. *Int. J. Heat Fluid Flow*, 23:639–646, 2002.
6. C.S. Peskin. The immersed boundary method. *Acta Numerica*, 11:1–39, 2002.
7. M. Uhlmann. An immersed boundary method with direct forcing for the simulation of particulate flows, 2004. (*accepted for publication in J. Comput. Physics*).
8. A.M. Roma, C.S. Peskin, and M.J. Berger. An adaptive version of the immersed boundary method. *J. Comput. Phys.*, 153:509–534, 1999.
9. X.Y. Lu and C. Dalton. Calculation of the timing of vortex formation from an oscillating cylinder. *J. Fluids Structures*, 10(527-541), 1996.
10. M.-C. Lai and C.S. Peskin. An immersed boundary method with formal second-order accuracy and reduced numerical viscosity. *J. Comput. Phys.*, 160:705–719, 2000.
11. N. Mordant and J.-F. Pinton. Velocity measurement of a settling sphere. *Eur. Phys. J. B*, 18:343–352, 2000.