Navier-Stokes simulation with constraint forces: Finite-difference method for particle-laden flows and complex geometries

Kai Höfler and Stefan Schwarzer
Institut für Computeranwendungen 1, Universität Stuttgart, 70569 Stuttgart, Germany
(Received 13 September 1999)

Building on an idea of Fogelson and Peskin [J. Comput. Phys. **79**, 50 (1988)] we describe the implementation and verification of a simulation technique for systems of non-Brownian particles in fluids at Reynolds numbers up to about 20 on the particle scale. This direct simulation technique fills a gap between simulations in the viscous regime and high-Reynolds-number modeling. It also combines sufficient computational accuracy with numerical efficiency and allows studies of several thousand, in principle arbitrarily shaped, extended and hydrodynamically interacting particles on regular work stations. We verify the algorithm in two and three dimensions for (i) single falling particles and (ii) a fluid flowing through a bed of fixed spheres. In the context of sedimentation we compute the volume fraction dependence of the mean sedimentation velocity. The results are compared with experimental and other numerical results both in the viscous and inertial regime and we find very satisfactory agreement.

PACS number(s): 02.70.−c, 47.55.Kf, 83.10.Lk

I. INTRODUCTION

Many applications in chemical engineering [1,2], fluid mechanics [3], geology [4], and biology involve systems of particles immersed in a liquid or gas flow. Examples of such systems are sedimentation processes, gas-solid or liquid-solid fluidized beds, blood, mixing processes when sediment-laden rivers enter lakes or the sea, powder transport by pneumatic conveying, the ticking of hour glasses, flocculation in suspensions, and many more. The long-ranged hydrodynamic interactions mediated by the fluid in the interstitial voids of a particulate, granular system greatly change its physical behavior as compared to the ''dry'' state without medium, which is characterized by the short-ranged, mostly viscoelastic forces that act when single grains come into contact.

Because of its great importance, the problem has been and is still attracting considerable attention on both the experimental and theoretical level. As is known from the physics of liquids on the molecular scale [5], long-time tails will arise in the correlation functions of conserved currents as, e.g., the liquid's velocity field, and the long-range character of the hydrodynamic interaction necessitates very careful studies of the system size dependence of the results. Short of analytic solutions of the problem, the challenge therefore is to find simulation techniques that are on the one hand accurate enough to allow reliable predictions, but on the other hand of sufficient numerical efficiency to permit studies on ''large'' systems in terms of particle numbers and confining geometry and ''long'' times with respect to the intrinsic velocities and length scales.

Some techniques, notably finite element or finite volume techniques [6–10], can reproduce very precisely the behavior of a small number of particles, but they are too computer intensive to simulate in three dimensions inherently collective, many-particle effects as, for example, the influence of the presence of a fluid phase on convection in granular assemblies, the bubbling in fluidized beds, or instabilities associated with gravitational overturning. Recently, however, in two dimensions successful simulations of rheological behavior [9,10] have been performed. The most time consuming part of these algorithms is the recurrent necessity to generate new, geometry adapted grids because continuous distortion of the initial grid will quickly result in very elongated elements or even overlaps with parts of the internal boundary (a particle).

Other techniques can deal with many particles, but use phenomenological expressions [11–14] for the coupling between particles and fluid that are incapable of rendering correctly single particle behavior and limit severely the predictive power of a method when new parameter ranges are explored. These include also the averaged equation techniques and the Euler-Lagrangian formulations that are popular in turbulent flow simulations, where direct simulation is prohibitively expensive [11,15,16].

Several techniques are based on the assumption of low Reynolds numbers, which turns the Navier-Stokes equation into the linear Stokes equation. Then one exploits the fundamental solution of the Stokes equation to eliminate the need to compute the solution everywhere in space, but rather uses a boundary element formulation [17] or a multipole expansion for the stress on the particle surface [18–20] possibly together with specific precautions for the divergent near-field lubrication forces [21]. Naive implementations of these methods require the storage and inversion of a full matrix. Only rather complex clustering techniques that exploit the decay of the influence of one particle on the other with increasing distance between the two can reduce this effort to be proportional to the number of particles (apart from logarithmic corrections). However, no efficient boundary integral technique is known to us for the nonlinear Navier-Stokes problem.

The most powerful techniques in terms of the ability to deal with large systems of the described kind have proved to be those that use a *fixed* nonadaptive grid to represent the fluid flow. The particle boundaries can then be represented only approximately as permitted by the regular discretiza-

tion. Such techniques have been successfully used by Ladd in conjunction with a lattice-Boltzmann flow solver [22,23] or in Ref. [24] with a more conventional finite difference Navier-Stokes solver.

The method that we propose here follows the immersed boundary technique proposed by Fogelson and Peskin [25]. Similar ideas are also employed in the fictitious domain method developed by Glowinski and co-workers [26–28] and earlier domain embedding methods [29] about which we learned only after most of this work had been completed. The basic idea of these two approaches is to use the same *constant* grid for the resolution of the fluid flow at all times and represent the particles not as boundary conditions to the flow, but by a volume force term or Lagrange multipliers in the Navier-Stokes equation. The fluid equation can then still solved by very fast specialized Fourier or multigrid techniques that exploit the regular grid structure.

After an introduction into the simulation method in Sec. II we will discuss several test cases. First, we will consider flow through static or rotating periodic arrays of cylinders and disks, where literature results for the drag and the resistance to rotation in the linear regime are available (Sec. IV). In dynamical simulations for many sedimenting spheres we reproduce the experimental findings in three dimensions for the hindered settling function (Sec. V).

II. NUMERICAL METHOD

We now describe the essentials of our technique to treat large numbers of rigid particles moving in a fluid modeled by the Navier-Stokes equations. The problem can be quite cleanly separated into three parts: (i) the fluid equations, (ii) the motion of the suspended particles, and (iii) their mutual coupling. Most of the technical details of the first two subproblems can be found in the literature and we will here give only the most important facts for completeness. The way of coupling these two phases is the heart of this paper and will be addressed in more detail.

A. Liquid

Our starting points are the Navier-Stokes equations describing the motion of a Newtonian fluid with inertia,

$$\rho\frac{\partial \boldsymbol{v}}{\partial t}+\rho(\boldsymbol{v}\cdot\boldsymbol{\nabla})\boldsymbol{v}=-\boldsymbol{\nabla}p+\eta\nabla^2\boldsymbol{v}+\boldsymbol{f}. \tag{1}$$

Here ρ and η denote the fluid density and viscosity, $\boldsymbol{v}$ and p its velocity and pressure, respectively, and $\boldsymbol{f}$ a volume force term. As usual [30], we will not consider the time independent gravity contribution to $\boldsymbol{f}$ explicitly, but cancel it against the hydrostatic pressure and omit both terms from Eq. (1). The corresponding buoyancy forces will be taken into account explicitly in the equations of motion of the suspended particles. However, we will require a fluctuating component of $\boldsymbol{f}$ as an essential ingredient of our simulation technique (see below), so that the volume force term must be left in Eq. (1). For the following, we will consider the liquid as incompressible, i.e., $\boldsymbol{\nabla}\cdot\boldsymbol{v}=0$, and choose the solution method for the fluid equation accordingly. Incompressibility is not a necessary condition for the coupling technique.

We use a staggered marker and cell (MAC) mesh as the base for a second order spatial finite-difference discretization of (1), which simplifies considerably the treatment of the pressure boundary conditions [31]. Details can be found, e.g., in Ref. [32], Chap. 6. The incompressibility constraint is satisfied via an explicit operator-splitting, fractional-time-step method, described in detail in the same reference. In that framework, one introduces an additional provisional ''velocity'' $\boldsymbol{v}^*$ *without physical meaning* in order to split the one velocity equation (1) into two,

$$\rho\frac{\boldsymbol{v}^*-\boldsymbol{v}^n}{\Delta t}=-p(\boldsymbol{v}^n\cdot\boldsymbol{\nabla})\boldsymbol{v}^n+\eta\nabla^2\boldsymbol{v}^n+\boldsymbol{f}^n, \tag{2}$$

$$\rho\frac{\boldsymbol{v}^{n+1}-\boldsymbol{v}^*}{\Delta t}=-\boldsymbol{\nabla}p^{n+1}. \tag{3}$$

Here, the superscripts n and $n+1$, respectively, denote the values at $t=n\Delta t$ and $t=(n+1)\Delta t$. This system of two equations is mathematically equivalent to the single equation that we started out with.

An equation for the pressure variable is obtained from Eq. (3) by taking the divergence and using that the divergence of the updated velocity field $\boldsymbol{v}^{n+1}$ must vanish,

$$\nabla^2 p^{n+1}=\frac{\rho}{\Delta t}\boldsymbol{\nabla}\cdot\boldsymbol{v}^*. \tag{4}$$

The steps above are often considered to be a projection of Eqs. (2),(3) onto a divergence-free subspace of the velocity vector field—the pressure equation is used to remove the ''perpendicular'' velocity components contained in $\boldsymbol{v}^*$—and thus the term ''projection method'' is often employed in this context. The exact distribution of terms to the split equations is not unique; in principle, some of the velocity terms on the right-hand side of (2) could appear in (3). Consequently, the unphysical temporary field $\boldsymbol{v}^*$ would take different values, without affecting $\boldsymbol{v}^{n+1}$.

The fluid equations must be solved subject to the boundary and initial conditions implied by the confining geometry, in our case a quadrilateral volume, which is either (i) limited by fixed walls on which no-slip conditions hold or (ii) periodically repeated in space. To obtain the boundary conditions for Eq. (4), we project expression (3) onto the boundary's outward pointing normal $\boldsymbol{n}$,

$$\frac{\rho}{\Delta t}(v_\perp^{n+1}-v_\perp^*)=-(\boldsymbol{n}\cdot\boldsymbol{\nabla})p^{n+1}. \tag{5}$$

Let us now consider the spatially discretized forms of Eq. (4) and (5) in the vicinity of a boundary, as, e.g., displayed in Fig. 1,

$$\begin{aligned}\frac{1}{h}\left(\frac{p_{1,m}^{n+1}-p_{0,m}^{n+1}}{h}-\frac{p_{0,m}^{n+1}-p_{-1,m}^{n+1}}{h}\right)+\frac{1}{h}\left(\frac{p_{0,m+1}^{n+1}-p_{0,m}^{n+1}}{h}\right.\\ \left.-\frac{p_{0,m}^{n+1}-p_{0,m-1}^{n+1}}{h}\right)\\ =\frac{\rho}{\Delta t}\left(\frac{v_{x;0,m}^*-v_{x;\Gamma}^*}{h}+\frac{v_{y;0,m}^*-v_{y;0,m-1}^*}{h}\right)\end{aligned} \tag{6}$$

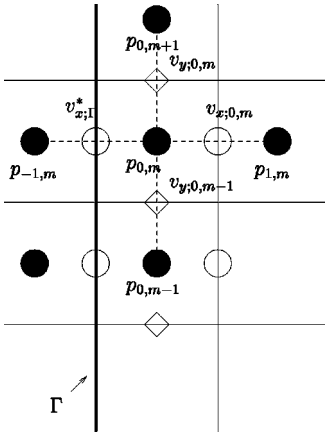

FIG. 1. Staggered marker and cell (MAC) mesh in the vicinity of a vertical boundary. The pressure $p_{0,m}$ discretization is centered in cell $(0,m)$, the velocities are shifted by half the cell size to the right $v_{x;0,m}$ and upward $v_{y;0,m}$, respectively. The evaluation of the Laplace operator applied to p involves one leg on which the pressure derivative across the boundary needs to be known. This derivative can be replaced by a term that involves both the unknown velocity v^*_Γ on the boundary and the updated velocity $v^{n+1}_{x;-1,m}$ on the boundary at the same location (not shown).

and

$$\frac{1}{h}(p^{n+1}_{0,m}-p^{n+1}_{-1,m})=-\frac{\rho}{\Delta t}(v^{n+1}_{x;\Gamma}-v^*_{x;\Gamma}). \tag{7}$$

Here, h is the lattice spacing of the grid used for the discretization and the suffix Γ refers to values on the boundary.

If we now substitute the expression $(p^{n+1}_{0,m}-p^{n+1}_{-1,m})/h$ from Eq. (7) in (6), we see that the the value of $v^*_{x;\Gamma}$ cancels from both sides of the equation. In other words, the solution of the pressure equation does not depend on the specific values of $v^*_{x;\Gamma}$ on the boundary. For computational convenience, we can thus in particular demand that the left-hand side of the equation specifying the boundary conditions shall vanish, i.e., $v^{n+1}_{x;\Gamma}=v^*_{x;\Gamma}$, corresponding to vanishing normal derivatives of the pressure across the boundary. Thus, we do not need to know these two values when we solve the pressure equation.

Also, the incompressibility guarantees that we satisfy the integrability conditions for the pressure equation with Neumann conditions:

$$0=\oint d\boldsymbol{A}\cdot\boldsymbol{\nabla}p\overset{!}{=}\frac{\rho}{\Delta t}\int d\boldsymbol{x}\boldsymbol{\nabla}\cdot\boldsymbol{v}^*=\frac{\rho}{\Delta t}\oint dA\boldsymbol{n}\cdot\boldsymbol{v}^*$$
$$=\frac{\rho}{\Delta t}\oint dA\boldsymbol{n}\cdot\boldsymbol{v}^{n+1}=0. \tag{8}$$

The actual solution of the pressure equation is effected by a fast multigrid scheme that exploits the regularity of the grid used for discretization. The general idea [33] and details for two dimensions (2D) [13,34] are given in the literature; the three-dimensional (3D) implementation is described in [35]. The hierarchical structure of a multigrid scheme ensures that the time to solve the pressure equation is asymptotically proportional to the number of grid points, a highly desirable feature for large-scale simulations. Since no transform into abstract spaces is necessary as, e.g., in Fourier techniques, multigrid is memory efficient and also well suited for parallelization by domain composition, as described, e.g., in [36].

One fluid time step hence consists of first computing the provisional velocity $\boldsymbol{v}^*$ from Eq. (2), which provides the information for the source terms of the pressure Eq. (4). Next the Poisson problem is solved with Neumann conditions $\boldsymbol{n}\cdot\boldsymbol{\nabla}p=0$ on the boundary where applicable, i.e., for the nonperiodic directions. For uniqueness, we demand that the average pressure is zero. Finally, the pressure values are substituted in Eq. (3) to calculate the updated velocities $\boldsymbol{v}^{n+1}$.

A local and linear Neumann stability analysis of the numerical scheme neglecting the coupling of the velocity equations by the pressure term leads to the condition

$$1>\frac{\eta^2}{\rho^2h^4}\left\{\left[\frac{\rho h^2}{\eta}+2\Delta t\left(d-\sum_{i=1}^{d}\cos(k_ih)\right)\right]^2+\left(\frac{\rho h}{\eta}\Delta t\sum_{i=1}^{d}v_i\sin(k_ih)\right)^2\right\} \tag{9}$$

for linear stability of a standing wave perturbation with wave number k_i; the v_i are the local fluid velocity components. The inequality (9) must be satisfied at all discretization nodes and for all wave numbers k_i. We obtain two simpler *necessary* conditions from (9) by demanding that the inequality holds for the two squares separately. In the first, we set $\cos(k_ih)=-1$ and find

$$\Delta t<\frac{1}{2d}\frac{\rho h^2}{\eta}, \tag{10}$$

which is familiar from discretizations of the diffusion equation. Now we define the maximum value of all velocities components over all space, $v_{\max}=\max|v_i(\boldsymbol{x})|$, and set $\sin(k_ih)=1$ in the second square. Then it follows that

$$\Delta t<\frac{h}{dv_{\max}}, \tag{11}$$

a term due to the presence of the convective nonlinearity of the Navier-Stokes equations.

B. Particle-fluid coupling

The most challenging part of the problem is the treatment of the no-slip boundary condition for the fluid on the particle surfaces. We here build on an idea that has been applied by Fogelson and Peskin [25] to the case of viscous flow. Instead of implementing the no-slip boundary conditions by modifying coefficients of the discretized system [37] or employing grid adaptivity as in finite-volume or finite-element techniques, we use the body-force term in the Navier-Stokes equations to implement constraints acting on the fluid such as to mimic the presence of rigid particles at appropriate regions in the flow. These regions will move as the physical particles will and they will comprise sufficiently many grid points in order to represent the geometry of the physical particles. Since the lattice points are spatially fixed, but the particles move, the association of grid points to particle representing regions will change in the course of the simulation.

In more detail, the computation of the motion of a physical particle i is decomposed into two contributions. A moving liquid volume element V_i of the same shape as the particle constitutes the first contribution. Let us for the moment assume that we knew how to move this element rigidly within the rest of the fluid, just as if it were a solid particle of fluid density ρ_f with mass $M_i^l = V_i \rho_f$ and moment of inertia $I_i^l = c M_i^l a^2$. For simplicity we assume that we deal with disks or spheres so that I is represented by a scalar, a being the particle radius and $c = 1/2$ or $2/5$ in two and three dimensions, respectively. As will become clear, our approach is not limited to these cases.

The second contribution is a particle template that also has the shape of the rigid physical particle, but it carries the mass M_i^t, and the moment of inertia I_i^t. These values complement those of the fluid contribution and sum to the values of the physical particle i, i.e., $M_i = M_i^l + M_i^t$ and $I_i = I_i^l + I_i^t$. The particle template is rigid by definition. We like to think of the template motion as representing the particle motion.

To make the general idea work, we must now describe how to achieve a rigid coupling between the template and the associated fluid element. To this end, we first introduce a number n_i of reference positions $\boldsymbol{r}_{ij}^r$, $j = 1, \ldots, n_i$ distributed over the volume of template i. The $\boldsymbol{r}_{ij}^r$ are vectors relative to the center of mass of template i. The associated spatial coordinates $\boldsymbol{x}_{ij}^r$ change only due to the movement and rotation of the rigid template,

$$\boldsymbol{x}_{ij}^r(t) = \boldsymbol{x}_i(t) + \mathbf{O}_i(t) \cdot \boldsymbol{r}_{ij}^r, \tag{12}$$

where $\mathbf{O}_i$ describes the instantaneous orientation and $\boldsymbol{x}_i$ is the position of the template. In the 2D case, we obtain $\mathbf{O}_i$ by forming the rotation matrix associated with the one angular degree of freedom of the particle, in 3D we use quaternions [38] to represent the particle orientation and to compute the rotation matrix $\mathbf{O}_i$.

Associated with each reference position is a tracer $\boldsymbol{x}_{ij}^m$, $j = 1, \ldots, n_i$ which tracks the motion of the fluid,

$$\dot{\boldsymbol{x}}_{ij}^m = \boldsymbol{v}(\boldsymbol{x}_{ij}^m). \tag{13}$$

Here and in the following we use dots to indicate the time derivative along the trajectories. Please note that the tracers are passively convected with the fluid and do not by themselves constitute new degrees of freedom.

The comparison of the location of the tracer to the position of its reference point allows us to judge whether the fluid volume V_i has changed shape or taken another trajectory as the associated template. The condition of rigid coupling translates into zero offset and zero difference velocity of the tracer and the reference point position. The density and location of the tracers should in general be chosen such that the number of degrees of freedom of the fluid that need to be controlled should equal the controlling number of tracers (please note the pertinent comments at the end of this section). That is to say that each tracer should control a fluid volume of h^d, d being the spatial dimension.

Now we obtain an explicit numerical scheme for the computation of the force density $\boldsymbol{f}$ constraining the fluid motion as follows. Whenever between tracer and reference position there occurs a nonvanishing difference in position $\boldsymbol{\xi}_{ij} = \boldsymbol{x}_{ij}^m - \boldsymbol{x}_{ij}^r$ or in velocity $\dot{\boldsymbol{\xi}}_{ij}$, we generate an additive contribution $\boldsymbol{f}_{ij}$ to the force density in the fluid that tends to drive the liquid and thus the tracer back to the reference position and to diminish their relative velocity. One possible choice is

$$\boldsymbol{f}_{ij}(\boldsymbol{x}) = (-k \boldsymbol{\xi}_{ij} - 2\gamma \dot{\boldsymbol{\xi}}_{ij}) \delta(\boldsymbol{x} - \boldsymbol{x}_{ij}^m), \tag{14}$$

where k is a ''spring'' constant, γ a damping constant, and $\delta(\boldsymbol{x})$ the Dirac distribution. In our explicit technique, k must be chosen large enough so that $|\xi_{ij}| \ll h$ holds at all times. Similarly, the dissipation introduced by the velocity-proportional friction controlled by γ must be small enough to be negligible against the external physical sources of energy dissipation. The force density $\boldsymbol{f}$ in the fluid equation (1) is the sum over all particles i and reference points j of $\boldsymbol{f}_{ij}$. A slight modification is needed in the case of periodic boundary conditions, which will be discussed separately in Sec. II D.

We must regularize the δ functions in the first part of the sum (14) by, e.g., linearly or quadratically weighted, interpolation to the nearest grid points. For example, in 2D, $\delta(\boldsymbol{x} - \boldsymbol{x}')$ is ''distributed'' linearly to the four grid points closest to $\boldsymbol{x}'$ with weights $w_{\Delta k, \Delta l}$ according to the perpendicular distances of $\boldsymbol{x}' = (x', y')$ from the discretization grid point (x_{kl}, y_{kl}) just below to the left. The indices k and l shall here denote grid indices as used in Fig. 1

$$w_{00} = \frac{1}{h^2}\left(1 - \frac{|x' - x_{kl}|}{h}\right)\left(1 - \frac{|y' - y_{kl}|}{h}\right),$$

$$w_{10} = \frac{1}{h^2}\frac{|x' - x_{kl}|}{h}\left(1 - \frac{|y' - y_{kl}|}{h}\right),$$

$$w_{01} = \frac{1}{h^2}\left(1 - \frac{|x' - x_{kl}|}{h}\right)\frac{|y' - y_{kl}|}{h},$$

$$w_{11} = \frac{1}{h^2}\frac{|x' - x_{kl}|}{h}\frac{|y' - x_{kl}|}{h}.$$

Please note that due to the use of a staggered grid, the (x_i, y_i) will in general be different for different components of the force.

The prefactor $1/h^2$ present in the expression for the weights above ensures that the spatial integral over this representation yields 1 [14]. Generalization of this formula to quadratic or nth order weights as well as general dimensions d is straightforward,

$$w^{(n)}_{\Delta l_1, \ldots, \Delta l_d} = \frac{1}{h^d}\prod_{i=1}^{d}\left[\delta_{0\Delta l_i}\left(1 - \frac{|x_i' - x_{i;l_1,\ldots,l_d}|^n}{h^n}\right) + \delta_{1\Delta l_i}\frac{|x_i' - x_{i;l_1,\ldots,l_d}|^n}{h^n}\right]. \tag{15}$$

Here, the index i refers to the vector component and the indices l_i the location of the closest grid point whose position components are all just smaller than those of $\boldsymbol{x}'$. Larger

n tend to concentrate the weight in the grid point closest to $\boldsymbol{x}'$. The application of such higher order weights has advantages when the reference points are placed up to exactly the radius of the particle.

Fogelson, Peskin [25] and Stockie [39] use smoother, longer ranged kernels, involving exponentials and trigonometric functions that (i) are numerically more expensive to evaluate and (ii) do not seem to accelerate convergence of the pressure solution even if the source terms are smoother. We consider as the advantage of these kernels their capability to obtain a grid independent limit for the force density and thus the emerging motion if the number of control points is fixed and $h \to 0$.

In order to estimate the largest possible k at a given time step Δt we consider a system of coupled masses. The value of k together with the reduced mass

$$M = \frac{M_i^{\mathrm{t}} \Delta M_{\mathrm{f}}}{M_i^{\mathrm{t}} + \Delta M_{\mathrm{f}}} \tag{16}$$

of the particle template M_i^{t} and the fluid element $\Delta M_{\mathrm{f}} = \rho_{\mathrm{f}} h^d$ at $\boldsymbol{x}_{ij}^r$ introduces a time scale of oscillation,

$$T^{\mathrm{t}} = \frac{1}{2\pi} \sqrt{\frac{M}{n_i k}}, \tag{17}$$

which must be resolved by the integration, i.e., $\Delta t \ll T^{\mathrm{t}}$. Only for these sufficiently small Δt, we can guarantee stability and numerical correctness of the particle integration.

For many computations we have adjusted k so that the above inequality is satisfied when Δt is of the order of the diffusive stability limit (10) imposed by the fluid integration, i.e.,

$$\Delta t < \frac{1}{2d} \frac{\rho h^2}{\eta}. \tag{18}$$

Thus, k can be determined from

$$k = \mathrm{const} \times \frac{M}{n_i \Delta t^2}. \tag{19}$$

In our tests that we have performed, stability was always achieved if $\Delta t < 1/20/T^{\mathrm{t}}$.

In 2D simulations we have had good experiences with γ values close to aperiodic damping of the particle template—in this case the assumption that the tracer positions are fixed yields $\gamma = \sqrt{k} M_t$. This has often eliminated small, but unphysical oscillations of the particles, in particular in the beginning of the simulations. These oscillations do not indicate instabilities of the numerical scheme, but reflect the oscillatory time scale resulting from template mass and coupling constant k.

In 3D simulations, probably due to the larger number of control points, $\gamma = 0$ seems to suffice. Our empirical experiences concerning the admissible maximum number of tracers are not conclusive. Whereas in dynamical simulations with moving particles it seems often possible to increase the number beyond that implied by the number of grid cells within one particle volume, in low-Reynolds-number computations with *static* particle arrangements, we often need to decrease the tracer density to be slightly below this limit to avoid instabilities.

Since the strongest gradients of the stress occur on the particle surface, computational efficiency suggests to reduce the tracer density in the particle interior. Moreover, in the viscous regime, the inertial effects due to the fluid in the interior of each particle domain are negligible. Interior tracers are only necessary to resolve changes in the angular velocity accurately. These will be important if significant changes in the angular velocity occur on time scales shorter than those for the diffusion of vorticity across the particle diameter, i.e., when we leave the regime of validity of the quasistatic approximation. For particle Reynolds numbers of about 1 and sufficiently small concentrations $\Phi \lesssim 0.2$, where it follows from the work of Goldman *et al.* [40] that effects from the rotational motion in viscous suspensions are generally weak, we consider the neglect of interior tracers justified.

C. Particle motion

The constraint force distribution that we have introduced to guarantee quasirigid fluid motion must be cancelled by opposite equal terms acting on the particle templates so that only external forces remain in the momentum balance of the combined system. This cancellation is quite naturally achieved by applying Newton's second law to the ''spring'' associated with each reference-point-tracer pair. That is to say that

$$\boldsymbol{F}_{ij} = k \boldsymbol{\xi}_{ij} + 2\gamma \dot{\boldsymbol{\xi}}_{ij} \tag{20}$$

is the force acting on particle template i at the location of reference point j. Similarly, the angular momentum balance will be satisfied when we take the torque to be

$$\boldsymbol{\tau}_{ij} = (\boldsymbol{x}_{ij}^r - \boldsymbol{x}_i) \times \boldsymbol{F}_{ij} \tag{21}$$

with respect to the template center of mass $\boldsymbol{x}_i$.

As further contributions single particle forces, gravity and buoyancy need to be taken into account,

$$\boldsymbol{F}_i^s = -M_i g \boldsymbol{e}_z + \rho V_i g \boldsymbol{e}_z = (\rho - \rho_p) V_i g \boldsymbol{e}_z, \tag{22}$$

where we have used ρ_p for the particle density and V_i for its volume (or area in 2D).

We now need to address the question of direct particle-particle interactions. In most non-Brownian suspensions these are negligible compared to the hydrodynamic effects at short distances between particles. For example, the approach of two particles is very strongly damped by the hydrodynamical lubrication forces between the two surfaces (see, e.g., [41]). Numerically these forces will be strongly underestimated when the distance between two particles becomes of the order of the lattice spacing. It is clear that all fixed grid techniques (including lattice Boltzmann) show this shortcoming, but even off-lattice particle methods such as dissipative particle dynamics (DPD) similarly underestimate the lubrication forces because the model intrinsic mean free path sets a length scale below which viscous stresses cannot be properly represented.

We model surface contacts between particles—and solve at the same time the numerical problems arising from possible large particle overlaps—by introducing a pairwise repulsive force that acts when the centers of two particles come closer than the sum of their radii a_i+a_k. Let the overlap between particles i and k be defined as $\boldsymbol{\zeta}_{ik}=(a_i+a_k-|\boldsymbol{x}_i-\boldsymbol{x}_k|)\boldsymbol{e}_{ik}$, with $\boldsymbol{e}_{ik}$ denoting the unit vector pointing from k to i. Then we take the force on i to be

$$\boldsymbol{F}_{ik}^{p}=-k_p\boldsymbol{\zeta}_{ik}. \tag{23}$$

In a similar manner, one may consider explicit interaction terms to restore the correct forces on the particles at short distances $\approx h$ [14,23]. However, our results indicate that in the studied concentration range the lubrication effects are recovered to a sufficient degree by our numerics. At increasing concentrations, such corrections will be of crucial importance [42].

The total force $\boldsymbol{F}_i$ on particle template i is the sum $\boldsymbol{F}_i=\boldsymbol{F}_i^s+\Sigma_k\boldsymbol{F}_{ik}^p+\Sigma_j\boldsymbol{F}_{ij}$. Apart from the fluid reaction force $\boldsymbol{F}_{ij}$, which also describes the ''unphysical'' constraint forces, (cf. Sec. II B), these terms are those that we expect for a ''physical'' particle to be present.

A Gear-predictor-corrector integrator of fourth order [34] serves to integrate the equations of motion for the translation of the template,

$$\ddot{\boldsymbol{x}}_i=\boldsymbol{F}_i/M_i. \tag{24}$$

For the quaternion formulation of the rotation, we refer the reader to Ref. [38].

To briefly summarize the above, we would like to stress again that the modeling of a rigid, heavy particle requires the (i) ''freezing'' of the region of fluid occupying the space of the particle and (ii) the coupling of this region to a particle template whose dynamical properties supplement those of the fluid in such a fashion that the coupled system behaves just as the modeled particle would. The arguments above can be put in a slightly more stringent mathematical context, for which we refer the reader to the Appendix of this paper.

D. Treatment of periodic systems and driving

In order to minimize effects from rigid walls it is often convenient to study periodically repeated cells. Periodicity in the following will always mean periodicity *in space* only. One should be aware, however, that the long-range nature of the hydrodynamic interactions might cause some artifacts in this case [43], mainly increasing autocorrelation times associated with the vertical motion.

Gravity, or an imposed driving pressure gradient single out one specific direction, say, the vertical. Whereas periodic boundary conditions pose no additional difficulties for the solution of the fluid and particle equations *perpendicular* to that direction, we need to specify more precisely what we mean by periodicity *parallel* to it. We will here consider the case of driving the system gravitationally via the density difference of particles and fluid.

For the fluid velocity field we require periodicity of the solution across the horizontal system boundary in z direction, i.e., $\boldsymbol{v}(\boldsymbol{x}+NL_z\boldsymbol{e}_z)=\boldsymbol{v}(\boldsymbol{x})$, where N is an arbitrary integer and L_z the size of the system in z direction. Similarly, the particle positions and associated velocities can be periodically extended.

If we consider the case of a section of a vertically positioned tube, then we can also demand the pressure to be a function periodic with period $L_z\boldsymbol{e}_z$. Due to the periodicity of $\boldsymbol{f}$ arising from the (periodic) particle motion and the periodic velocity field, the source term of the pressure equation is also periodic, the integrability conditions are satisfied and we obtain a periodic solution for the pressure which is unique up to a constant, which we choose so that the average pressure vanishes. Physically, this system will evolve towards a statistically stationary state in which on average the viscous forces originating at the walls balance the gravitational driving.

In the case of sedimenting systems, it is however more natural to imagine the simulation cell as a small part of a larger system. In this case, there are no walls that could provide balancing viscous forces to counteract the gravitation. The least constraining condition on the system that still guarantees evolution to a stationary state is to assume that there is no net acceleration at any time on the components within the considered cell. That is to say, that at any time, we must make sure that the integral over the simulation volume of the right-hand side

$$\frac{D}{Dt}\rho\boldsymbol{v}=\underbrace{-\boldsymbol{\nabla}p}_{(I)}+\underbrace{\eta\nabla^2\boldsymbol{v}}_{(II)}+\underbrace{\boldsymbol{f}}_{(III)} \tag{25}$$

of the Navier-Stokes equation (1) vanishes. Now, for the term (II) we apply Gauss's theorem and find that it depends only on the values of the velocity gradient tensor integrated over the surface of the cell. Since the velocity field is periodic, contributions from opposite faces cancel identically and this term is always zero. If we decompose the pressure into (i) a linearly varying part and (ii) a purely periodic contribution, then for similar reasons, the volume integral over the gradient of the periodic part vanishes.

Thus we are left with contributions from the integrated force density (III) and an average pressure gradient (I), which arises in addition to the well-known hydrostatic part. If we take these two time dependent terms to be equal—in formal analogy to the hydrostatic case—then they cancel from Eq. (25). The total acceleration of the fluid in the simulation volume thus vanishes.

For computational purposes, we simply subtract from the field $\boldsymbol{f}$ its spatial average and thus obtain fluid and pressure equations that allow spatially periodic solutions,

$$\boldsymbol{f}(\boldsymbol{x})=\sum_{ij}\boldsymbol{f}_{ij}(\boldsymbol{x})-\frac{1}{V}\int_V d\boldsymbol{x}\sum_{ij}\boldsymbol{f}_{ij}(\boldsymbol{x})=\frac{1}{V}\sum_{ij}\boldsymbol{F}_{ij}. \tag{26}$$

Physically, we can think of the associated constant pressure gradient $\nabla p_{\mathrm{lin}}=(1/V)\int d\boldsymbol{x}\boldsymbol{f}$, which drops from Eq. (25), as giving rise to a buoyancy force. This buoyancy force, however, is already correctly included in the ''tracer-spring'' forces appearing in the particle equations of motion (cf. Sec. II C), since we would reduce the tracer forces by their average and add the same term again, now in the form of a buoyancy contribution from the liquid.

E. Summary of the numerical procedure

We now briefly summarize the sequence of steps described above that are necessary to perform one time step in our simulation.

(1) We perform the predictor part of the Gear algorithm for both the template and marker positions. No forces need to be known at this point, because the prediction is solely based on Taylor coefficients of the trajectories that were previously recorded.

(2) Likewise, we update the quaternions and the angular velocity representing the degrees of freedom of rotation of each template. We then use their values to compute the rotation matrix $\mathbf{O}_i(t)$ to find the predicted location of the reference points [cf. Eq. (12)].

(3) From the predicted particle template location the interparticle forces are computed according to Eq. (23).

(4) Now we are in a position to compare the predicted tracer positions to the predicted reference point locations and infer the constraint forces $\boldsymbol{f}_{ij}$ necessary to impose rigid motion on the fluid [Eq. (14)]. The integration of $\boldsymbol{f}_{ij}$ over its support and summation over all tracers associated to template i yields the reaction force of the fluid onto the template. Similarly, we determine the acting torque using Eq. (21).

(5) Knowing the force distribution $\boldsymbol{f}_{ij}$ and the fluid velocity field, we know know all terms on the right-hand side of Eq. (2) and can perform a fluid update by (a) calculation of the provisional ''velocity'' field $\boldsymbol{v}^*$, from the current velocities $\boldsymbol{v}^n$ and the constraint forces $\boldsymbol{f}_{ij}$, (b) determination of the pressure from the Poisson equation (4) by a multigrid procedure, (c) and finally by advancing the fluid velocity to $\boldsymbol{v}^{n+1}$ from the knowledge of the pressure field and the ''old'' velocities $\boldsymbol{v}^n$ [Eq. (3)].

(6) From the predicted tracer positions and the new fluid velocity values we obtain the correction terms for the tracer trajectories necessary for the second (correction) part of Gear's integrator for the tracers [Eq. (24)].

(7) Similarly, a correction step is performed for the translational and the angular velocity of the templates using the previously computed torques and forces. The corrected angular velocity will then be used to correct the quaternion values that trace the orientation of the particle (Sec. II C).

At this point we have completed the time stepping and can perform measurements on a consistent set of dynamical quantities.

III. VALIDATION OF THE NAVIER-STOKES SOLVER

The Navier-Stokes solver outlined in Sec. II A has been tested on the limiting stationary flow pattern in the driven-cavity problem [44] for Reynolds numbers on the box scale of $\mathrm{Re}=UL\rho/\eta<100$, where U denotes the scale of the imposed velocities. As time dependent flows we have tested sinusoidal velocity profiles between parallel plates and verified the exactly exponential approach to rest and the associated decay constant.

Due to the explicit method and the inherent ''diffusive'' stability constraint, $\Delta t<\rho h^2/2d\eta$, very low-Reynolds-number calculations ($\eta\to\infty$) require increasingly shorter time steps and will become prohibitively expensive. However, this limitation can be overcome by an implicit time stepping, possibly exploiting that the Navier-Stokes equations, turn into the linear Stokes equations for $\mathrm{Re}\to 0$.

We do not think that it is practical to use the proposed method in many particle systems (order of 10 000) beyond Reynolds numbers *on the particle scale* larger than $\mathrm{Re}=\rho aU/\eta=10,\ldots,20$, because the flow on and below the Kolmogorov scale must be resolved. In this regime the grid refinement for methods without turbulence modeling will soon render the computational effort unacceptable.

It should also be noted that modern implicit and adaptive-grid methods are more suitable for high precision computation of time dependent pure fluid flows in *static* geometries. In the context of model building for suspension flows with *moving particles* however, fixed grid methods do not require remeshing. Moreover, an explicit technique like the one proposed above provides great flexibility to implement particle-particle interactions, i.e., additional short-range attractive forces of van der Waals type to model aggregation phenomena, or material properties of the particles.

IV. THE CASE OF ONE PARTICLE: CUBIC PERIODIC ARRAYS

A. Setup

As the first test case in which boundary conditions on the particle surface have a nontrivial influence on the flow we consider a fluid passing through a cubic periodic arrangement of fixed spheres at low Reynolds numbers. The arising flow has been analyzed in the point particle approximation by Hasimoto [45] for small volume fractions Φ. Among others, Ladd [19] as well as Sangani and Acrivos [46] have considered larger volume fractions up to the limit of $\Phi=\pi/6$ with numerical methods in the viscous regime. More recently, Koch and Ladd [47] have published results for cylinder arrays at moderate Reynolds numbers using lattice-Boltzmann techniques.

To this end, we impose at each point of the grid at first a constant acceleration $\boldsymbol{f}=\boldsymbol{e}_z\Delta P/L$ corresponding to a constant pressure drop $\Delta P/L$ over the length L of the cubic cell. Periodic boundary conditions are used on the fluctuating, nonlinearly increasing part of the pressure and the flow velocities. The particle positions and orientations are fixed to be the initial ones.

The simulations are performed in 2D and 3D as dynamical calculations starting from a fluid field at rest until a steady flow state results. The approach to stationarity is slower at lower solid volume/area fractions. In 3D, we stop when an exponential approximation indicates that the expected additional changes to the flow rate are less than a fraction of 0.02. In 2D, we have adjusted the pressure drop during the simulation to obtain constant volumetric flow rates and thus constant Reynolds numbers.

The drag force $\boldsymbol{F}_D$ on one particle is determined by summing the contributions from all the corresponding reference-point-tracer pairs according to Eq. (20). If we divide the modulus $F_D=|\boldsymbol{F}_D|$ of the drag force by the volumetric flow rate U,

$$U=\frac{1}{V}\int d\boldsymbol{x}v_z(\boldsymbol{x}) \tag{27}$$

obtained by averaging the flow velocity over the cell volume

V, and refer the value to that of an isolated sphere, then the dimensionless drag coefficient

$$\chi_D = \frac{F_D}{6\pi\eta a U} \tag{28}$$

results.

In 2D, F_D is the drag force per unit length of the cylinder. We refer it to ηU, which has the required units of force per unit length,

$$\chi_D = \frac{F_D}{\eta U}. \tag{29}$$

Both these drag coefficients are known to have corrections $O(\mathrm{Re}^2)$ in arrays that have reflection symmetry with respect to the axis of the flow [47,48].

Similarly, the rotational drag coefficients can be computed by imposing constant angular velocities on the particles and letting the flow adjust. In the viscous regime, both in 2D and 3D, the torque is proportional to the angular velocity. In 3D, we refer the resulting torque on the sphere to the theoretical value of an isolated sphere

$$\chi_R = \frac{\tau_R}{8\pi\eta a^3 \omega}. \tag{30}$$

In 2D, the theoretical value of the torque per unit length on an isolated cylinder in an infinite medium is $4\pi a^2 \eta\omega$, so that a dimensionless drag coefficient

$$\chi_R = \frac{\tau_R}{4\pi a^2 \eta\omega} \tag{31}$$

can be defined.

B. Mesh size dependence

Due to the lack of adaptivity at the particle surface, we expect the effective hydrodynamic radius of the particles to be slightly larger than the geometrical radius of the reference point arrangement, because each point controls a fluid volume of extent h^d reaching beyond its geometric location by $h/2$ in each direction. In fact, if the confining radius of the reference point placement is taken to be the geometric template radius, then we measure as a function of mesh size the drag coefficients shown in Fig. 2, here $\Phi = 0.0335$. We see that indeed the values extrapolate to asymptotic values as $h \to 0$ with leading error proportional to h.

We can thus improve the accuracy of the simulation by taking this effect explicitly into account [23] by modifying the placement of the reference points. Let us write for the effective hydrodynamic radius a_{eff}, assuming that Δa is linear in h,

$$a_{\mathrm{eff}}(h) = a + \Delta a(h) = a + mh. \tag{32}$$

The dependence of the friction coefficient on the volume fraction can be Taylor expanded around the volume fraction corresponding to a particle with radius a,

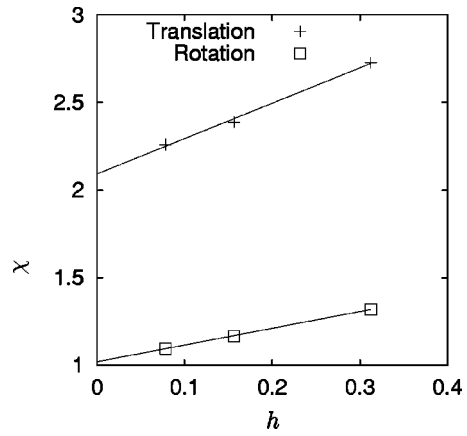

FIG. 2. Dependence on mesh size h of the dimensionless translational $\chi_D/6\pi\eta a$ and rotational friction coefficient $\chi_R/8\pi\eta a^2$ in three dimensions, denoted by symbols + and ⊡, respectively. The values are for a volume fraction of $\Phi = 0.0335$ at $\mathrm{Re} \to 0$. It can be seen that the necessary correction to the drag coefficient is linear in h.

$$\chi[\Phi(a_{\mathrm{eff}})] = \chi[\Phi(a)] + \chi'[\Phi(a)]\Phi'(a)\Delta a(h) = \chi[\Phi(a)] + \chi'[\Phi(a)]\Phi'(a)mh. \tag{33}$$

Comparing the slope in Fig. 2 to the prefactor of h, and using the derivative $\chi'[\Phi(a)]$ from the literature [19], we find $m \approx 0.3$. This value holds for 3D and quadratic $n=2$ interpolation scheme for the δ functions in the force density [cf. Eq. (15)]. For the dynamical simulations in the next section, we therefore retract the tracers by an amount of $0.3h$ from the geometric surface; in 2D we use $m = 0.5h$ with $n=1$ interpolation.

C. Translational drag and rotational friction coefficients

In Fig. 3 we show the results of the extrapolation to $h = 0$ of the translational drag for 2D and 3D. In 3D we compare with Ladd's [19] numerical solutions of the Stokes equation with the same boundary conditions. In 2D, we compare to the results of Sangani [46].

In the 3D simulations the pressure drop is adjusted during the simulation to achieve the same volumetric flow rate (and thus Reynolds number) independent of volume or area fraction of the obstacles. In 2D, the Reynolds number based on the flow rate and the radius of the obstacle does not exceed 0.1.

In Fig. 4 the corresponding results for the rotational friction coefficients χ_R are shown as functions of volume and area fraction. In two dimensions different symbols denote different arrangements of the cylinders with respect to the discretization. In one case the particle is located at position (0,0) in the cell of size $L \times L$, in the other the unit cell was chosen to include two particles, one at (0,0), the other at $(L/2, L/2)$. The symmetries of the array imply that in both cases the same scalar friction coefficient must result.

In both 2D and 3D, we have computed the drag coefficients by linear interpolation to $h = 0$ of the h-dependent results. For the densest packings in 2D, the gap between the particles is about 13 grid cells wide for the finest grid used. The relative difference of the friction coefficients computed

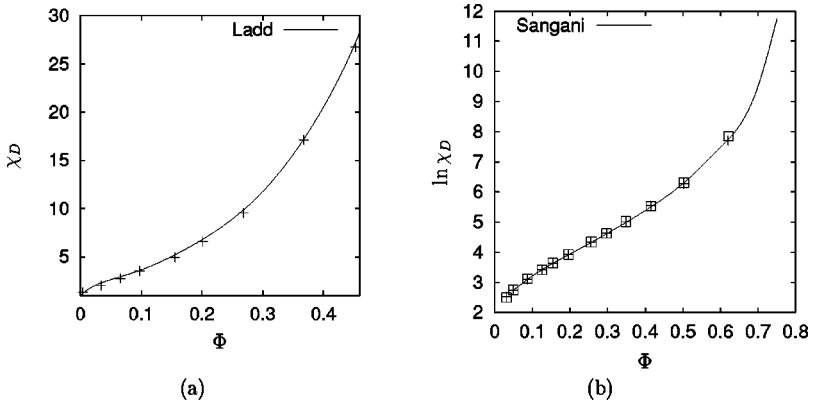

FIG. 3. (a) Drag coefficient χ_D in a periodic, 3D simple cubic arrangement of spheres. (b) Natural logarithm $\ln \chi_D$ of the drag coefficient in an array of cylinders for two different unit cells that are expected to yield the same resistance to flow. The solid line (b) is a cubic spline through the logarithm of the semianalytical results for the friction coefficients given in Ref. [46].

for the two finest grids (13 grid cells and 6 grid cells across the gap) is about 0.04. Table I summarizes the numerical results for the rotational drag in the cylinder array for which we did not find values in the literature available to us.

V. SEDIMENTATION VELOCITY AS A FUNCTION OF VOLUME FRACTION

The measurement of the sedimentation velocity of an ensemble of many spheres as a function of the volume fraction constitutes a more realistic test than the two above. As in the case of the fixed sphere arrangement in the two preceding tests we choose periodic boundary conditions on our sample cell that now, however, contains many spheres whose position and orientation can evolve dynamically without artificial kinematic constraints. Gravitation acts as the driving force via the density difference of particles and fluid. It was chosen such that the Reynolds number on the particle scale of a single sedimenting particle, in 3D,

$$\mathrm{Re}=\frac{2}{9}\frac{a^3}{\eta^2}\rho(\rho_p-\rho)g \tag{34}$$

is about 0.1. In 2D, we assured by test simulations on single falling discs that we are in the same range of Reynolds numbers.

As motivated in Sec. II B, we use only one shell of tracers such that their hydrodynamic radius equals the geometric radius of the template. Their number is taken to be $4\pi a^2/h^2$, i.e., approximately equal to the number of fluid volume elements on the surface of each particle.

The mean sedimentation velocity v_s is measured in experiments either as the velocity with which the upper front of the particle rich phase settles or, when single particle velocities can be measured, as the mean velocity of the particle phase. We will here adopt the latter method because we cannot observe concentration fronts in a periodic simulation cell,

$$v_s=\frac{1}{N}\sum_{i=1}^{N}\boldsymbol{v}_i\cdot\boldsymbol{e}_z. \tag{35}$$

To obtain a dimensionless quantity, we refer the value of v_s to the velocity v_0 of a single falling particle in the same cell. This procedure takes some corrections due to finite cell size and Reynolds number into account.

Figure 5 shows the results for the sedimentation velocity as a function of volume fraction for 2D and 3D. In 3D we compare to the empirical law of Richardson and Zaki [49]

$$v_s/v_0=(1-\Phi)^n, \tag{36}$$

with $n=5.0$. We see good agreement to the experimental findings in the viscous regime, which finds values of $n=5,\ldots,6$ [50].

In 2D, the computation of a hindered settling function is a somewhat academic exercise, which we have done to juxtapose the results to the 3D ones. We have fixed the Reynolds number for the single cylinder experiment to 0.1 on the particle scale. We vary the area fraction, keeping all other parameters constant. The mean settling velocity as a function of area fraction is then computed and normalized by the single particle value.

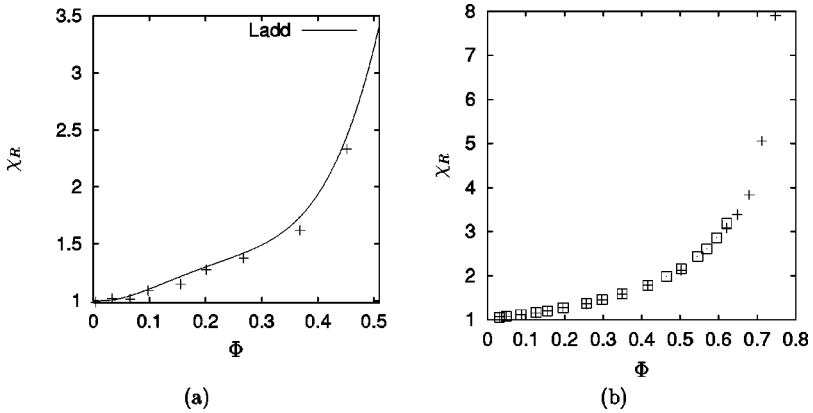

FIG. 4. Rotational friction coefficient χ_R in a periodic arrangement of (a) 3D spheres, (b) 2D cylinders. In 3D, the solid line is a cubic spline through the numerical results of Ladd [19].

TABLE I. Dimensionless resistance to rotation in an array of cylinders at $\mathrm{Re}=a^2\omega\rho/\eta=0.1$.

Φ	$\chi_R=\tau_R/4\pi a^2\eta\omega$
0.0314	1.07
0.049	1.08
0.0872	1.11
0.125	1.15
0.155	1.2
0.196	1.27
0.256	1.36
0.297	1.45
0.349	1.58
0.415	1.78
0.502	2.12
0.62	3.08
0.649	3.38
0.679	3.83
0.712	5.05
0.747	7.9

In 2D, the best fit of a Richardson-Zaki-like law yields an exponent of $n\approx 3.8$, substantially smaller than in 3D. The effect of a lower n in 2D has been seen before using point-force approximations for the suspended particles [51]. However, the fit to such a power law is only motivated by the analogy to 3D and is in fact not convincing. Most of the difference at larger volume fractions might be related to the area fraction of the random loose disk packing in 2D (≈ 0.7) as compared to the smaller value of the random loose sphere packing in 3D (≈ 0.6), where we expect the sedimentation velocity to drop to 0 [52].

VI. SOME NUMERICAL QUESTIONS

A. Performance Data

As described in Sec. II our method requires the time stepping of the Navier-Stokes equation, the computation of the tracer movement with two interpolation steps for the fluid velocity and force density, and the integration of the ordinary differential equations that describe the particle motion. For this latter part, we use a linked-cell technique to reduce the determination of the particle contact forces to an $O(N)$ problem in the number of particles. The time spent in this part of the algorithm is only a few percent of the total computation time and is negligible.

Because the exact ratio of time spent in the fluid solver and in the coupling between the fluid and the particles depends on various parameters, as, e.g., the mesh size h, the volume fraction Φ, and the number of tracers per particle, we just want to give an example for the computational cost of a typical simulation done to calculate the sedimentation velocity of a suspension. In this particular case we used $65^3\approx 2.7\times 10^5$ fluid grid points and $h=0.375$. Thus the cell was of a cube with edges of approximately 24 particle radii length. A volume fraction of $\Phi=0.1$ corresponds therefore to 331 particles with 93 tracers each, representing a total of 30 783 tracers.

One timestep of this system takes on average 2.6 s on a Digital Personal Workstation 433au. The fluid solver needs about 54% and the coupling code 35% of the time. The rest was spent in miscellaneous statistical calculations, file input/output, etc. The computation of particle interactions and the trajectory integration requires less than 1% of the total CPU time.

In other words, per timestep we spent about 5.1×10^{-6} s on each grid point, about 29×10^{-6} s on each tracer and about 50×10^{-6} s on each suspended particle. The memory requirements sum up to 80 bytes per tracer and 120 bytes per fluid grid point, including the data structures necessary for the implicit pressure update.

B. Comparison to other techniques

1. Particle methods

As we have seen, the major challenge of direct suspension simulation is the proper representation of moving boundaries in conjunction with numerical efficiency. Since particles are naturally described in a Lagrangian picture by co-moving dynamical variables, one could justifiably ask why not go the whole way and also use Lagrangian techniques for the fluid. This is even more true as the Navier-Stokes equations result as the continuum limit of the average motion of particle systems under quite general circumstances [53]. The following issues should be considered.

(1) The implementation of a pure Lagrangian technique is cleaner and more straightforward from a technical point. At the same time, even very simple interparticle potentials (as in the dissipative particle dynamics method (DPD) or in direct simulations of, e.g., Lennard-Jones systems, require a lot more computational effort than the treatment of a grid node in a Navier-Stokes solver. Our experience points to factors of $3-5$ for structureless particles with linear spring interactions

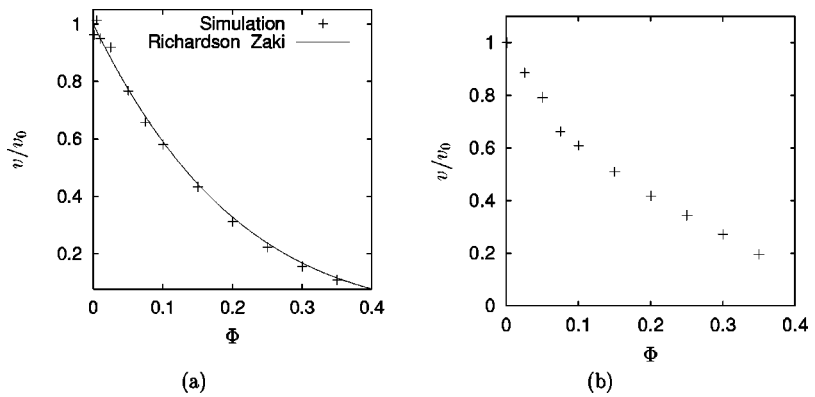

FIG. 5. Sedimentation velocity in a periodically repeated cell: (a) 3D, (b) 2D. The action of gravity is counteracted by a constant pressure gradient in the opposite direction, such that at all times the volume flux across a horizontal cut through the cell vanishes. The size of the 3D cell is $12a\times 12a\times 12a$, that of the 2D cell $64a\times 64a$ ($h=0.25$). Both simulations are performed at a Reynolds number of approximately 0.1 on the particle scale (for single particles falling).

as compared to one Navier-Stokes node in 3D.

(2) In general, it is not simple to predict the average material properties of the particle phase from the properties of the interaction. One strong point of DPD with respect to other computational methods is the theoretical knowledge that has been collected in the recent past about how the microscopic simulation parameters control the macroscopic properties of the DPD liquid. However, as the discussion about modeling heat conduction within the DPD framework has shown, such extensions require quite considerable contrivances and yet often material parameters of the pure suspending phase have to be determined *a posteriori* from the simulation before the properties of the multiphase system can be determined.

(3) In particle codes, one often takes the route to model larger or different particles by gluing together several (constituent) particles of the type constituting the solvent phase [54]. Such an approach is convenient and efficient because it normally does not require special measures to track the orientation or the translation of the composite particle, because these can be reconstructed from the position of the glued particles. Also, the interactions between the solvent and the constituents of the composite complex particles are often of the same nature as those between the solvent particles and can thus be computed in the same fashion without changing the structure of the simulation program.

However, this convenience comes at the price of computing *pair* interactions between the constituents to fix the shape of the composite particles. In contrast, the marker-template coupling is a *single* particle interaction.

Of course, it is also quite possible to connect the constituents of a solute particle to a shape preserving (possibly massless) rigid template as has been suggested here for the fluid-particle coupling. This approach seems to have been taken in the DPD framework by a technique used in Ref. [55].

(4) In general one must be aware of proper scale separation. For example, the discrete nature of the particle phase introduces a ''noise'' term, which causes fluctuating motion of the suspended particles. This noise can be considered a feature if simulations at low Peclèt numbers shall be performed (as the Lennard-Jones simulations by Rapaport [56]) but must be eliminated in one way or the other in the limit of large Peclèt numbers. Similarly, the discrete nature of the solvent particles will cause breakdown of lubrication forces when the solute particles approach surface-surface distances closer than the mean free path (Knudsen length) of the solvent, as we have discussed in the context of lubrication force modeling in Sec. II C for fixed grid methods.

(5) Particle codes with short-ranged interactions, similar to explicit lattice-Boltzmann techniques, display a built-in compressibility of the solvent phase. The speed of sound in these simulations is often very small because the particle interactions are taken to be rather soft in order to allow (apparently) large time steps. In contrast, in continuum techniques one has efficient methods to impose incompressibility (solving a Laplace equation for the pressure equation in our case) or to resort to penalty techniques [32] that effectively replace the long-range Laplace solver by introducing an artificially compressible material law.

2. Fixed lattice based methods

If instead of using a particle technique one attempts to solve a continuum equation for the fluid phase, then one has considerable freedom in the choice of solution methodology and in the choice of the phenomenological parameters entering the equation. Every numerical solution of a continuum equation requires a discretization of the computational domain. If a high precision of the solution is required, the discretization is often taken on an unstructured grid and locally refined in the course of the solution process. The computational drawbacks are the complex data structures required and the entrained computational cost.

In addition, since particle-fluid systems are characterized by moving boundary conditions, a remeshing of the computational grid must occur when it is distorted to such an extent that the requirements of the, e.g., finite-element or finite-volume technique cannot longer be met.

As we have outlined in the introduction of this paper, it is thus advantageous for the simulation of many-particle systems and if the demands on the precision of the solution are not too high, to pick a nonadaptive, regular grid to discretize the continuum fluid equation. On such a lattice we can, for example, use the lattice-Boltzmann technique, a finite-difference compressible or incompressible Navier-Stokes or Stokes solver, certain lattice gas automata, etc.

As in particle methods, certain issues should be kept in mind.

(1) Perhaps the most problematic point is the implementation of the no-slip conditions on the particle surfaces and the computation of the stresses. Concerning the matching of the particles to the grid, considerable freedom exists and includes (a) (first order) discretization of the particle surface on appropriate grid nodes or links of the respective technique [22,36,57] (first order); (b) smoothly varying ''interpolation'' coefficients to obtain a better, possibly second order accurate estimate of the local influence of the boundary; for lattice Boltzmann, cf. Ref. [58], for finite differences Ref. [37].

Similarly, boundary stresses must be evaluated. For lattice Boltzmann, typically the knowledge of the change of the velocity population associated with a boundary link or node is required, whereas the stress evaluation for finite differences involves direct evaluation of the stress tensor using local pressure and velocity. Please note that in our method none of the above steps is necessary, as the penalty forces on the fluid include the stresses that are determined already in the course of the fluid time step.

(2) Depending on the type of discretization used, stabilization measures need to be taken. The explicit lattice Boltzmann described in [22] requires time averaging (over two time steps) of the torque to avoid instabilities. Nonstaggered finite difference grids may show grid decoupling instabilities. Also the general stability constraints from linear analysis of all explicit methods have to be considered, most notably the Neumann diffusive criterion linking time step and square of the grid size.

(3) In most modern (explicit or implicit) methods the computational effort to perform one time step is proportional to the number of discretization ''cells'' that represent often a set of variables describing the local configuration of the continuum. For example, in our method, the explicit part of the

Navier-Stokes time step and the implicit solution of the pressure equation both have this property. Likewise, lattice-Boltzmann methods require only a reweighting of locally available information to update the local velocity histogram. So, the computational effort differs due to the constant of proportionality. An explicit lattice-Boltzmann technique can most directly be compared to a penalty technique for incompressible flows [32], which are of similar computational complexity.

(4) The inclusion of energy conservation into a lattice-Boltzmann scheme, which is necessary to compute heat flow, is to our knowledge still an open question [59,60]. There is no comparable difficulty when using directly discretized flux-conservation equations as, e.g., the Navier-Stokes equation for the momentum flux.

To summarize the above, in suspension physics there is not one single simulation method that addresses the entire spectrum of possible conditions, with or without fluid inertia, with or without particle inertia, low or high Peclèt numbers, constitutive properties of the suspending fluid, constitutive properties of the suspended phase (surface tension, elasticity, additional interactions), particle geometry, etc. with flexibility and efficiency. As we have stated in the Introduction, the niche for the fixed grid method presented here is computation at moderate or zero Reynolds number with likewise good to moderate accuracy depending on the price in processing time that one is willing to pay. In that respect it is very similar to the lattice-Boltzmann methods [19,22,23].

3. Brownian motion

The Navier-Stokes describes the evolution of average velocity and pressure and is thus naturally suitable for systems at high Peclèt numbers where no information about the discrete nature of the fluid and the associated fluctuations is required.

On the other extreme, molecular simulation captures all fluctuating quantities as well as their spatial and temporal correlations. In between these two extremes we find Brownian dynamics ideas [18,38]—incorporating thermal fluctuations into the particle equations of motions—and fluctuating hydrodynamics (cf. [30], Ch. XVII); both neglect spatial and temporal correlations of the thermal driving forces.

In fluctuating hydrodynamics stress fluctuations $\sigma'_{ij}(\boldsymbol{x})$ are built into the fluid equations and their statistical properties are found by considering ideas from nonequilibrium thermodynamics [30],

$$\langle \sigma'_{ik}(\boldsymbol{x}_1,t_1)\sigma'_{lm}(\boldsymbol{x}_2,t_2)\rangle = 2k_BT\eta(\delta_{il}\delta_{km}+\delta_{im}\delta_{kl}) \times \delta(\boldsymbol{x}_1-\boldsymbol{x}_2)\,\delta(t_1-t_2), \quad (37)$$

where k_B is Boltzmann's constant and T the temperature. From this expression, we can find for any given discretization a finite stress vector to be included on the right-hand side of the Navier-Stokes equation (1). We thus see that the description of fluid thermal fluctuations can quite naturally be included in our numerical framework and might be the subject of a future study. It should be mentioned here that in the context of lattice-Boltzmann methods similar ideas have been used to include thermal fluctuations [22,61,62].

C. Scope for improving the numerical scheme

There are some unresolved numerical questions that we would like to address before we conclude.

In order to eliminate the purely numerical constants k and γ, we have tested an implicit scheme that does not require tracers, but instead associates in a quite similar fashion a force directly with each reference point. Then we make tentative fluid integration steps using these forces and compare the resulting difference velocities of fluid and template at the reference points. The forces are now modified by an additive term proportional to the velocity difference, but with the opposite direction. We thus arrive at successively ''better'' force distributions resulting in decreasing difference velocities. We have observed such a procedure to converge in simulations with $\mathrm{Re}\approx 1$, but did not use it extensively due to the additional computational effort.

Due to the small slip allowed in our explicit ''penalty'' coupling technique there occurs a small residual fluid motion *inside* the particle regions. As a result, a certain amount of energy is dissipated even if the coupling of fluid and tracers is not dissipative. In 3D, we find that the fraction of energy dissipated inside particle regions ranges from about 0.01 in suspensions with $\Phi=0.05$ up to about 0.1 in for $\Phi=0.3$. Correspondingly, one can observe changes in the sedimentation speed, which, due to these additional dissipation channels, turn out to be smaller than expected. For more precise measurements, this purely numerical dissipation must be reduced by stiffer couplings and correspondingly smaller time steps. More elaborate implicit techniques will also improve the situation, because the relative motion inside particle domains can be suppressed entirely.

VII. CONCLUSION

We have presented a comparatively simple and physically appealing method to simulate particle suspensions. The capabilities of the resulting algorithm have been assessed in 2D and 3D by comparisons to single particle behavior and the collective motion in suspensions. Two major approximations have been made that influence the precision of the algorithm: (i) the Navier-Stokes equation has been solved on a regular grid, thus trading computational speed for the possibility to accurately model geometrical details; lubrication forces will not be resolved by the fluid simulation, when particles come closer than the mesh spacing; (ii) the no-slip boundary conditions on the particle surfaces are realized by constraint forces that control the fluid motion at the positions of the particles. This representation implies that the particle surfaces acquire a certain degree of ''fuzziness,'' but allows a straightforward computation of stresses and a continuous representation of the particle motion.

The method is sufficiently accurate to reproduce translational and rotational drag in cylinder and sphere arrays with errors below 1% and to make *quantitative* predictions of suspension behavior in a wide range of volume fractions for relevant system sizes. In general, the limitations, strengths, and performance data (where known) are very similar to those of other fixed grid methods, as, e.g., the lattice-Boltzmann simulations of suspensions [22,23]. Our approach therefore presents another independent way to test analytical results for viscous suspension behavior and to extend these

results into the regime where both particle and liquid inertia become important. It is possible to incorporate more complex particle shapes, thermal fluctuations or to use the machinery of implicit numerical techniques to overcome some of the apparent limitations, in particular the need for very short time steps as Re approaches 0. As we work with discretized continuum equations, the method easily accommodates different descriptions of the fluid phase, e.g., by the Stokes equation [63] or more exotic rheological fluid properties (e.g., [64]).

ACKNOWLEDGMENTS

We thank Georg Barthelmes, George Bossis, Willi Brandstädter, Hans W. Buggisch, François Feuillebois, Roland Glowinski, Elizabeth Guazzelli, Hans Herrmann, John Hinch, Esa Kuusela, Stefan Luding, Matthias Müller, Christian Manwart, Gerald Ristow, Bernd Wachmann, and Christian Wrobel for several motivating and enlightening discussions and for contributions to the simulation code. K.H. thanks the Deutsche Forschungsgemeinschaft, SFB 404, for financial support. We have profited from generous allotments of computer time and access to the computer centers in Jülich and Stuttgart.

APPENDIX: SOME FURTHER REMARKS

1. On a continuum picture

The arguments in Sec. II B can be made more precise if we try to understand the procedure described above as the discretized version of a continuum problem. Let us state the momentum equation of the fluid (1) in the form

$$\frac{D}{Dt}\rho \boldsymbol{v} = \boldsymbol{\nabla}\cdot\mathbf{T} + \boldsymbol{f}, \tag{A1}$$

where $\mathbf{T}$ denotes the stress tensor of the fluid. Its divergence is the expression $-\boldsymbol{\nabla}p + \eta\nabla^2 v$ for Newtonian fluids used above. For the rigid particle template, we have

$$M_i^{\mathrm{t}}\dot{\boldsymbol{v}}_i = \boldsymbol{F}_i^{lp} + \boldsymbol{F}_i^{p} - (M_i - \rho V_i)g\boldsymbol{e}_z, \tag{A2}$$

with $\boldsymbol{F}_i^{lp}$ as the force acting from the fluid on the template (determined by summation of the constraint forces), $\boldsymbol{F}_i^{p}$ the force due to the presence of other particles and the term proportional to g represents weight and buoyancy.

Let the region of fluid covered by template i at time $t=0$ be denoted by $\Omega_i(0)$. Continuous time evolution according to Eq. (A1) will deform this region into $\Omega_i(t)$. The positions of the tracers track this deformation and thus the maximum $|\boldsymbol{\xi}_{ij}|$ measures the deviation of $\Omega_i(t)$ from the original particle shape. It might be intuitive to think of the marker elongations $\boldsymbol{\xi}_{ij}$ as representative of discrete amplitudes of a continuous displacement field $\boldsymbol{u}_i(\boldsymbol{x})$ defined on template i, related to the position $\boldsymbol{x}_{ij}^m$ by $\boldsymbol{x}_{ij}^m = \boldsymbol{x}_{ij}^r + \boldsymbol{u}_i(\boldsymbol{x}_{ij}^r)$. The constraint fluid force density $\boldsymbol{f}_i$ for particle i is related to $\boldsymbol{u}_i$ as implied by Eq. (14), i.e.,

$$\boldsymbol{f}_i(\boldsymbol{x}_{ij}^m) = -k\boldsymbol{u}_i(\boldsymbol{x}_{ij}^r) - 2\gamma(d/dt)\boldsymbol{u}_i(\boldsymbol{x}_{ij}^r), \tag{A3}$$

where the time derivative must take the time dependence of $\boldsymbol{x}_{ij}^r$ into account. If the external stresses remain bounded, this form guarantees that $\boldsymbol{u}$ and the rate of change of $\boldsymbol{u}$ on $\partial\Omega_i(t)$ remain bounded and approach zero as k and γ increase. By construction, $\Omega_i(t)$ is the union of $\boldsymbol{x}+\boldsymbol{u}_i(\boldsymbol{x})$ with $\boldsymbol{x}$ being one of the points constituting the template at time t. The incompressibility of the flow guarantees that the volume of $\Omega_i(t)$ is time invariant and equals V_i, the volume of the template.

2. Conserved quantities

The last argument shows that the sum of the mass of fluid in $\Omega_i(t)$ and the template mass M_i^t is constant and equal to the mass of the physical particle.

The total change of momentum of the coupled system is obtained by integrating (A1) over $\Omega_i(t)$, which results in

$$\dot{\boldsymbol{P}}_i^l = \oint_{\partial\Omega_i(t)} \mathbf{T}\cdot d\boldsymbol{A} + \boldsymbol{F}_i^l, \tag{A4}$$

where we have used Gauss's theorem to convert the integral over the divergence of the stress tensor into a surface integral. The vectors $\boldsymbol{P}_i^l$ and $\boldsymbol{F}_i^l$ denote the total fluid momentum in $\Omega_i(t)$ and the sum of the constraint forces, respectively. By construction, $\boldsymbol{F}_i^l + \boldsymbol{F}_i^{lp} = \boldsymbol{0}$. Up to a degree of accuracy determined by $\sim 1/k$ and $\sim 1/\gamma$ the center of mass velocity of the fluid in $\Omega_i(t)$ coincides with the template centers $\dot{x}_i$. Thus, for the sum of Eqs. (A4) and (A2) we recover the equation of motion for the center of mass of a rigid particle P of mass $M_i = M_i^t + \rho V_i$ in the flow,

$$M_i\ddot{\boldsymbol{x}}_i = \oint_{\partial P} \mathbf{T}\cdot d\boldsymbol{A} - (M_i - \rho V_t)g\boldsymbol{e}_z. \tag{A5}$$

To see what happens for the angular momentum balance, we form the vector products of Eq. (A1) with a vector pointing from the template center of mass $\boldsymbol{x}_i$ to $\boldsymbol{x}$ before we perform the integration over $\Omega_i(t)$, i.e.,

$$\int_{\Omega_i(t)} d\boldsymbol{x}(\boldsymbol{x}-\boldsymbol{x}_i)\times\frac{D}{Dt}\rho\boldsymbol{v} = \int_{\Omega_i(t)} d\boldsymbol{x}(\boldsymbol{x}-\boldsymbol{x}_i)\times\boldsymbol{\nabla}\cdot\mathbf{T} + \int_{\Omega_i(t)} d\boldsymbol{x}(\boldsymbol{x}-\boldsymbol{x}_i)\times\boldsymbol{f}. \tag{A6}$$

The left-hand side is the total change of angular momentum of the fluid $\dot{\boldsymbol{L}}_i^l$ and we can employ the symmetry of the stress tensor to convert the first term on the right-hand side to a surface integral,

$$\dot{\boldsymbol{L}}_i^l = \oint_{\partial\Omega_i(t)} (\boldsymbol{x}-\boldsymbol{x}_i)\times d\boldsymbol{A}\cdot\mathbf{T} + \boldsymbol{\tau}_i^l. \tag{A7}$$

As in the case of linear momentum, we also consider the corresponding equation for the template. The contribution to the torque of the gravitational forces vanishes, and the contribution of the constraint forces cancels $\boldsymbol{\tau}_i^l$ by construction up to an accuracy determined by the maximum deviation of a tracer from a reference point. To the same accuracy, the change of angular momentum of the fluid can be written as the moment of inertia times the change of angular velocity, which equals that of the template, again to the degree per-

mitted by the tracer force law. We thus recover approximately the equation of angular motion of a rigid particle P in a fluid in the absence of external torques,

$$I_i\dot{\omega}=\oint_{\partial P}(\boldsymbol{x}-\boldsymbol{x}_i)\times d\boldsymbol{A}\cdot\mathbf{T}. \tag{A8}$$

3. Uniqueness

The distribution of the constraint forces in the particle/template region is not uniquely determined by demanding that the fluid motion should match the rigid body motion of the particle. Also, the motion of the system as a whole does not uniquely fix $\boldsymbol{f}$. The prescription given in the text selects one of the possible distributions, but leaves some ''gauge freedom,'' which is very similar to the indeterminate of forces in static networks of rigid elements. Also here, the rigid template can ''absorb'' inner stress consequences on the motion.

If we write the Navier-Stokes equations in the form,

$$\boldsymbol{f}-\boldsymbol{\nabla}p=\boldsymbol{G}(\boldsymbol{v}), \tag{A9}$$

where $\boldsymbol{G}$ collects the terms depending on spatial and temporal derivatives of $\boldsymbol{v}$, then from taking curl and divergence, we find

$$\boldsymbol{\nabla}\times\boldsymbol{f}=\boldsymbol{\nabla}\times\boldsymbol{G}, \tag{A10}$$

$$\nabla^2 p=\boldsymbol{\nabla}\cdot\boldsymbol{f}-\boldsymbol{\nabla}\cdot\boldsymbol{G}. \tag{A11}$$

The first equation implies that the velocity distribution (and its rate of change) only determines $\boldsymbol{f}$ up to the gradient $\boldsymbol{\nabla}\phi$ of a scalar. In general, such contributions influence the pressure distribution (A11) and thus the motion. If, however, ϕ fulfills additional conditions, i.e., $\phi=0$ on $\partial\Omega_i$, it will not have consequences for the motion or in the exterior of Ω_i. If we extend ϕ over the whole domain such that $\boldsymbol{\nabla}\phi$ exists everywhere and $\phi=0$ in the exterior of all Ω_i, then $p+\phi$ solves (A11) for $\boldsymbol{f}\rightarrow\boldsymbol{f}+\boldsymbol{\nabla}\phi$. The scalar ϕ does not contribute to the momentum flux through $\partial\Omega_i$, if its surface integral $\oint_{\partial\Omega_i}d\boldsymbol{A}\phi$ vanishes. Likewise, the contribution to the torque, here with respect to the origin, but similarly for any reference point, is

$$\int_{\Omega_i}d\boldsymbol{x}\boldsymbol{x}\times(\boldsymbol{\nabla}\phi)=\int_{\Omega_i}d\boldsymbol{x}\phi\boldsymbol{\nabla}\times\boldsymbol{x}-\int_{\Omega_i}d\boldsymbol{x}\boldsymbol{\nabla}\times(\boldsymbol{x}\phi)$$

$$=-\oint_{\partial\Omega_i}d\boldsymbol{A}\times(\boldsymbol{x}\phi) \tag{A12}$$

and vanishes if the surface integral vanishes. This is the case in particular, if $\phi=0$ on $\partial\Omega_i$ as stated above.

Since the $\boldsymbol{f}$ acting on the fluid has a reaction force of opposite sign on the particle template, these conditions guarantee also vanishing force and torque contributions on the template as a whole.

[1] S. L. Soo, *Particles and Continuum: Multiphase Fluid Dynamics* (Hemisphere, New York, 1989).

[2] D. Gidaspow, *Multiphase Flow and Fluidization* (Academic Press, San Diego, 1994).

[3] *Mobile Particulate Systems*, edited by E. Guazzelli and L. Oger (Kluwer Academic, Dordrecht, 1995).

[4] K. Pye and H. Tsoar, *Aeolian Sand and Sand Dunes* (Unwin Hyman, London, 1990).

[5] J. P. Hansen and I. R. McDonald, *Theory of Simple Liquids*, 2nd ed. (Academic Press, London, 1986).

[6] J. Feng, H. H. Hu and D. D. Joseph, J. Fluid Mech. **261**, 95 (1994).

[7] H. H. Hu, Int. J. Multiphase Flow **22**, 335 (1996).

[8] A. A. Johnson and T. E. Tezduyar, Comput. Methods Appl. Mech. Eng. **134**, 351 (1996).

[9] P. S. J. J. Haan, J. Rheol. **42**, 891 (1998).

[10] B. Maury, J. Comput. Phys. **156**, 325 (1999).

[11] Y. Tsuji, T. Tanaka, and T. Ishida, Powder Technol. **71**, 239 (1992).

[12] S. Yonemura, T. Tanaka, and Y. Tsuji, ASME/FED **166**, 303 (1993).

[13] W. Kalthoff, S. Schwarzer, and H. Herrmann, Phys. Rev. E **56**, 2234 (1997).

[14] S. Schwarzer, Phys. Rev. E **52**, 6461 (1995).

[15] M. Sommerfeld and H.-H. Qiu, Int. J. Heat Fluid Flow **12**, 20 (1991).

[16] T. Tanaka, T. Kawaguchi, S. Nishi, and Y. Tsuji, ASME/FED **166**, 17 (1993).

[17] W. L. Wendland and J. Zhu, Math. Comput. Modelling **15**, 19 (1991).

[18] J. F. Brady and G. Bossis, Annu. Rev. Fluid Mech. **20**, 111 (1988).

[19] A. J. C. Ladd, J. Chem. Phys. **88**, 5051 (1988).

[20] G. Bossis and J. F. Brady, J. Chem. Phys. **91**, 1866 (1989).

[21] A. Sangani and G. Mo, Phys. Fluids **6**, 1653 (1994).

[22] A. Ladd, J. Fluid Mech. **271**, 285 (1994).

[23] A. Ladd, J. Fluid Mech. **271**, 311 (1994).

[24] B. Wachmann, W. Kalthoff, S. Schwarzer, and H. Herrmann, Granular Matter **1**(2), 75 (1998).

[25] A. L. Fogelson and C. S. Peskin, J. Comput. Phys. **79**, 50 (1988).

[26] R. Glowinski, T.-W. Pan, and J. Périaux, Comput. Methods Appl. Mech. Eng. **112**, 133 (1994).

[27] R. Glowinski, T. W. Pan, T. I. Hesla, and D. D. Joseph, Int. J. Multiphase Flow **25**, 755 (1999).

[28] R. Glowinski *et al.*, Int. J. Numer. Methods Fluids **30**, 1043 (1999).

[29] B. L. Buzbee, F. W. Dorr, J. A. George, and G. H. Golub, SIAM (Soc. Ind. Appl. Math.) J. Numer. Anal. **8**, 722 (1971).

[30] L. D. Landau and E. M. Lifschitz, *Hydrodynamik*, Vol. 6 of Lehrbuch der Theoretischen Physik, 5th ed. (Akademie-Verlag, Berlin, 1991).

[31] M. Fortin, R. Peyret, and R. Temam, J. Mec. **10**, 357 (1971).

[32] R. Peyret and T. D. Taylor, *Computational Methods for Fluid Flow*, Springer Series in Computational Physics (Springer, New York, 1983).

[33] W. Hackbusch, *Multi-Grid Methods and Applications* (Springer, Heidelberg, 1985).

[34] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, *Numerical Recipes in C*, 2nd ed. (Cambridge University Press, Cambridge, 1992).

[35] K. Höfler, Master's thesis, Universität Stuttgart, 1997.
[36] B. Wachmann and S. Schwarzer, Int. J. Mod. Phys. C **9**, 759 (1998).
[37] H. R. Schwarz, *Numerische Mathematik* (Teubner, Stuttgart, Germany, 1986).
[38] M. P. Allen and D. J. Tildesley, *Computer Simulation of Liquids* (Clarendon Press, Oxford, 1987).
[39] J. Stockie, Ph.D. thesis, University of British Columbia, 1997.
[40] A. J. Goldman, R. G. Cox, and H. Brenner, Chem. Eng. Sci. **22**, 637 (1967).
[41] R. H. Davis, in *Mobile Particulate Systems*, Proceedings of the Summer School on Mobile Particulate Systems, Cargese, Corsica, 1994 edited by E. Guazzelli and L. Oger (Kluwer Academic, Dordrecht, 1995).
[42] J. R. Melrose, J. H. van Vliet, and R. C. Ball, Phys. Rev. Lett. **77**, 4660 (1996).
[43] D. L. Koch and E. Shaqfeh, J. Fluid Mech. **224**, 275 (1991).
[44] J. D. Bozeman and C. Dalton, J. Comput. Phys. **12**, 348 (1973).
[45] H. Hasimoto, J. Fluid Mech. **5**, 317 (1959).
[46] A. Sangani and A. Acrivos, Int. J. Multiphase Flow **8**, 193 (1982).
[47] D. L. Koch and A. J. C. Ladd, J. Fluid Mech. **349**, 31 (1997).
[48] C. C. Mei and J.-L. Auriault, J. Fluid Mech. **222**, 647 (1991).
[49] J. F. Richardson and W. N. Zaki, Trans. Inst. Chem. Eng. **32**, 35 (1954).
[50] H. Nicolai *et al.*, Phys. Fluids **7**, 12 (1995).
[51] W. Kalthoff, S. Schwarzer, G. Ristow, and H. Herrmann, Int. J. Mod. Phys. C **7**, 543 (1996).
[52] P. Mills and P. Snabre, Europhys. Lett. **25**, 651 (1994).
[53] S. Harris, *An Introduction to the Theory of the Boltzmann Equation* (Holt, Rinehart, and Winston, New York, 1971).
[54] T. Pöschel and V. Buchholtz, Phys. Rev. Lett. **71**, 3963 (1993).
[55] E. Boek, P. Coveney, H. Lekkerkerker, and P. van der Schoot, Phys. Rev. E **55**, 3124 (1997).
[56] D. Hirshfeld and D. Rapaport, Phys. Rev. Lett. **80**, 5337 (1998).
[57] G. H. Ristow, Phys. Rev. E **55**, 2808 (1996).
[58] O. Filippova and D. Hänel, J. Comput. Phys. **147**, 219 (1998).
[59] H. Puhl, Ph.D. thesis, University of Stuttgart, 1997.
[60] F. J. Alexander, S. Chen, and J. D. Sterling, Phys. Rev. E **47**, R2249 (1993).
[61] A. J. C. Ladd, Phys. Rev. Lett. **70**, 1339 (1993).
[62] E. G. Flekkøy and D. H. Rothman, Phys. Rev. E **53**, 1622 (1966).
[63] W. M. Brandstädter, Master's thesis, Universität Karlsruhe, Institut für Mechanische Verfahrenstechnik und Mechanik, 1999.
[64] J. Feng, D. D. Joseph, R. Glowinski, and T. W. Pan, J. Fluid Mech. **283**, 1 (1995).