Algorithm for the simulation of particle suspensions with inertia effects

W. Kalthoff, S. Schwarzer, and H. J. Herrmann
ICA 1, Universität Stuttgart, 70569 Stuttgart, Germany
(Received 13 December 1996; revised manuscript received 1 May 1997)

We present a numerical method that can deal efficiently with large numbers of particles suspended in incompressible fluids. Our approach is based on a finite-differencing multigrid Navier-Stokes solver. The interactions between particles and fluid at the no-slip interfaces are taken into account by numerically integrating the stress tensor over the particle surface using an analytical expansion for the fluid fields. We demonstrate the validity of our approach by performing numerical simulations of several simple systems in two spatial dimensions and compare our results with experimental, numerical, and theoretical results. Our simulation method quantitatively reproduces known experimental and numerical results while remaining numerically fast enough to provide dynamical simulations of particle-liquid systems with several thousand particles in the regime of particle Reynolds numbers less than about 50. [S1063-651X(97)07008-6]

PACS number(s): 02.70.Ns, 47.11.+j, 47.55.Kf

I. INTRODUCTION

The numerical simulation of the motion of particles in a fluid is a very difficult problem which until today has not been solved to entire satisfaction. Some techniques, notably finite element [1–3], finite-volume, or boundary integral techniques [4,5], can reproduce very well the behavior of a small number of particles, but they are too computer intensive to simulate inherently collective, many-particle effects such as the influence of the presence of a fluid phase on convection in granular assemblies or the bubbling in fluidized beds. Other techniques can deal with many particles, but use phenomenological expressions [6–9] 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. Some techniques are valid only for small Reynolds numbers [10,11]; others use averaged equations that apply only in turbulent regimes and use phenomenological expressions for momentum exchange [6,12,13].

We want to study collective effects of particles immersed in fluid at low and moderate Reynolds numbers (practically less than about 50 on the particle scale) in a computer effort growing linearly with the number of particles and with the number of grid points when we increase the system size. The method that we want to put forth in this paper goes beyond the Stokesian approximation and is particularly useful in the regime of intermediate Reynolds number. Its main alternative feature is the application of a stable series expansion technique to find the stress tensor on the particle surface. As the Navier-Stokes solver we employ a fast multi-grid finite-difference technique (for details see below), which we consider to be a reliable alternative to the lattice Boltzmann solvers, which in our parameter range have been employed already to study multiphase flow [14].

In this paper we will first describe a two-dimensional implementation of our simulation technique (Sec. II). We will then show in a range of quite different simulations the degree of agreement between our results and existing experimental, numerical, and theoretical data. These simulations show the dependence on the container size of the single cylinder drag coefficient (Sec. III A), the variation of the drag coefficient with the Reynolds number (Sec. III B), and the force exerted on the cylinders in a periodic square array of cylinders (Sec. III C).

II. MODEL

The model used to obtain the results presented in this article consists of three main parts presented in the following sections. (i) The liquid phase is described using a finite-difference scheme for the Navier-Stokes equation, (ii) the liquid-particle interaction is obtained by integrating the stress tensor over the particle surface as well as implementing it as a no-slip boundary into the fluid, and finally (iii) possible particle contacts are accounted for by a short-range repulsive force acting on touching particles.

A. Liquid model

The time evolution of the liquid phase is described by the Navier-Stokes equation

$$\rho_{\mathrm{l}}\left[\frac{\partial \mathbf{v}}{\partial t}+(\mathbf{v}\cdot\boldsymbol{\nabla})\mathbf{v}\right]=-\boldsymbol{\nabla}p+\eta\nabla^2\mathbf{v}+\mathbf{f}. \tag{1}$$

Here ρ_{l} denotes the liquid density and η its dynamic viscosity and $\mathbf{f}$ is the volume force density in the system, including only gravity for the results obtained here. This equation is complemented by the incompressibility constraint

$$\boldsymbol{\nabla}\cdot\mathbf{v}=0, \tag{2}$$

justified by the fact that the typical velocities of our systems are much smaller than the sound velocity. Equation (2) presents a constraint on the velocity field that must be fulfilled at all times.

We choose to spatially discretize this equation on the staggered marker and cell (MAC) mesh, i.e., the pressure and the velocity components v_x and v_y are located on three square meshes with lattice spacing Δx. With respect to the pressure grid, the grids for the x and y velocity components are shifted by $\Delta x/2$ in the x and y directions, respectively.

The spatial discretization is second order and has been described in detail, e.g., in Chap. 6 of Ref. [15]. The MAC construction has several computational advantages and is a simple means to avoid numerical instabilities due to mesh decoupling (see Refs. [15,16] for more details).

The discretization in time is explicit and of first order. Let the index n refer to values at time t_n and $n+1$ to those at $t=t_{n+1}=t_n+\Delta t$ after a time step of duration Δt. The time-discretized Navier-Stokes equation is given by

$$\rho_l \frac{\mathbf{v}_{n+1}-\mathbf{v}_n}{\Delta t} = -\rho_l(\mathbf{v}_n\cdot\boldsymbol{\nabla})\mathbf{v}_n - \boldsymbol{\nabla} p_{n+1} + \eta\nabla^2\mathbf{v}_n + \mathbf{f}_n . \tag{3}$$

A von Neumann stability analysis [15,17] of the described discretization of the Navier-Stokes equation shows that the values Δx and Δt are subject to the two stability constraints

$$\Delta t \leqslant \frac{4\eta}{\rho_l(|v_x^{\max}|+|v_y^{\max}|)^2} \tag{4}$$

and

$$\Delta t \leqslant \frac{\rho_l(\Delta x)^2}{4\eta}. \tag{5}$$

To solve the two components of Eq. (3) in conjunction with the incompressibility constraint (2), we employ an operator splitting technique. Adding and subtracting a new temporary variable $\mathbf{v}^*$ discretized in the same manner as $\mathbf{v}$, on the left-hand side of Eq. (3) yields,

$$\rho_l \frac{(\mathbf{v}_{n+1}-\mathbf{v}^*)-(\mathbf{v}_n-\mathbf{v}^*)}{\Delta t} = -\rho_l(\mathbf{v}_n\cdot\boldsymbol{\nabla})\mathbf{v}_n - \boldsymbol{\nabla} p_{n+1} + \mathbf{f}_n + \eta\nabla^2\mathbf{v}_n . \tag{6}$$

This contrivance allows us to split Eq. (6) into two equations, which must then be solved simultaneously:

$$\rho_l \frac{-(\mathbf{v}_n-\mathbf{v}^*)}{\Delta t} = -\rho_l(\mathbf{v}_n\cdot\boldsymbol{\nabla})\mathbf{v}_n + \mathbf{f}_n + \eta\nabla^2\mathbf{v}_n , \tag{7}$$

$$\rho_l \frac{\mathbf{v}_{n+1}-\mathbf{v}^*}{\Delta t} = -\boldsymbol{\nabla} p_{n+1}. \tag{8}$$

Taking the divergence of Eq. (8), and observing $\boldsymbol{\nabla}\cdot\mathbf{v}_{n+1}=0$, we find that the pressure obeys the Poisson equation

$$\frac{\boldsymbol{\nabla}\cdot\mathbf{v}^*}{\Delta t} = \frac{\nabla^2 p_{n+1}}{\rho_l}. \tag{9}$$

Once we have calculated $\mathbf{v}^*$ from Eq. (7), we solve the Poisson equation by a fast iterative multigrid scheme [18] under the Neumann condition that the normal derivative of the pressure on the system boundary vanishes. We justify these boundary conditions below. After inserting the resulting pressure values at the new time $n+1$ into Eq. (8), we finally find the updated divergence free velocities $\mathbf{v}_{n+1}$.

An essential part of the solution are the boundary conditions on the MAC mesh for the unknown quantities $\mathbf{v}$ and p. We position container walls such that the mesh points corresponding to the perpendicular velocity components lie on the wall. That is to say that on the vertical walls one finds the x-velocity component and on the horizontal walls the y component. The only difference operator that requires parallel components located on the wall is the Laplacian of the viscous term in Eq. (7). We treat this case by employing a mirror principle, which uses the bulk expression for the operator near the system boundaries and defines a fictitious velocity value ''beyond'' the wall from linear extrapolation of the interior velocity value and the fixed known boundary value. Periodic boundary conditions do not require additional work.

One finds the boundary conditions of the Poisson equation for the pressure by multiplying Eq. (8) with the outward directed surface normal $\hat{\mathbf{n}}$,

$$-\frac{\partial p_{n+1}}{\partial \hat{\mathbf{n}}} = \rho_l \frac{\hat{\mathbf{n}}\cdot(\mathbf{v}_{n+1}-\mathbf{v}^*)}{\Delta t}. \tag{10}$$

According to Gauss's theorem, the volume integral of Eq. (9) must equal the surface integral of Eq. (10) if the pressure equation shall have a solution

$$\int d\mathbf{x}\ \boldsymbol{\nabla}\cdot\mathbf{v}^* = \oint ds\ \hat{\mathbf{n}}\cdot(\mathbf{v}_{n+1}-\mathbf{v}^*). \tag{11}$$

This requirement is fulfilled and not dependent on $\mathbf{v}^*$ because there is no flux of liquid through fixed walls and contributions from cuts between periodic images cancel pairwise.

Somewhat surprisingly it turns out that the discretization of the Poisson equation (9) does not depend on the value of $\mathbf{v}^*$ on the boundary. The reason is that the discretization of the Laplacian on the right-hand side in Eq. (9) contains a term in $\mathbf{v}^*$ on the boundary that, if substituted from Eq. (10), is canceled by a term in the discretized divergence occurring on the left-hand side. Thus one can choose *arbitrary* values for $\mathbf{v}^*$ on the boundary. In particular, we demand $\hat{\mathbf{n}}\cdot(\mathbf{v}_{n+1}-\mathbf{v}^*)=0$, corresponding to vanishing normal derivatives of the pressure on the boundary.

1. The force exerted on the particles

To obtain the force on the particle due to the liquid we integrate the stress tensor (here $\mathbf{1}$ denotes the rank-2 unit tensor)

$$\sigma = -p\mathbf{1} + \mu[\boldsymbol{\nabla}\mathbf{v} + (\boldsymbol{\nabla}\mathbf{v})^t] \tag{12}$$

over the particle surface, i.e.,

$$\mathbf{F} = \int_S \sigma\cdot\hat{\mathbf{n}}\ dA. \tag{13}$$

A straightforward approach to evaluate this integral is to consider directly the numerical liquid pressure and velocity derivatives at the particle surface, insert them into Eq. (13), and integrate numerically to find the force on the particle. However, such an approach suffers from a number of problems, namely, (i) on the scale of the particle size, the liquid grid is rather coarse. Therefore, the numerical liquid field is not sufficiently smooth to obtain accurate values of its nu-

merical derivatives. (ii) The amount of noise introduced by taking numerical derivatives seems to be so large that the system proved to be unstable even at extremely small time steps. The exact source of the problem is unclear to us, but our observation is that, within reasonable limits for the time step, the inaccuracies of the numerical force determination introduced a positive feedback via the no-slip boundary condition onto the liquid fields. We therefore follow a different approach to calculate the integral (13).

If viscous forces are sufficiently strong (or more precisely, if the particle Reynolds number rV/ν is small enough) and if the system is not too dense, then the fluid velocity and pressure in the vicinity of the sphere will not differ much from the analytically known smooth distributions for spheres or cylinders in resting or sheared liquids (see, e.g., [19]). Therefore, we are led to consider an expansion of the unknown analytical solution in a complete set of functions, leaving us with the unknown expansion coefficients. We approximate these coefficients using the numerical data that the simulation provides at each time step as described in detail further below.

The advantages of such an approach are that (i) the function system chosen can be a physically motivated one, for instance, the lowest-order approximation could be the analytical solution for vanishing Reynolds number; (ii) numerical noise can be reduced very effectively by matching the function system to a sufficiently large number of lattice points; and (iii) no numerical integration is necessary, because we now have an analytical expression for σ in Eq. (13). Thus we can express the integral (13) as a function of solely the expansion coefficients.

We now demonstrate how to calculate the coefficients a_i of the expansion. Let y represent one of the liquid velocity components v_x, v_y, or the pressure field p. Then we write

$$y(\mathbf{r})=\sum_j a_i f_i(\mathbf{r}), \tag{14}$$

where the f_i's form our set of functions, which does not necessarily have to be orthogonal. The coefficients a_i will depend on the values of the liquid field at the grid points surrounding the particle. In general, we will have to calculate m coefficients from n grid points, with $m<n$, thus giving us the following overdetermined system of equations,

$$y_{\mathbf{r}_i}=y(\mathbf{r}_i), \tag{15}$$

written for all grid points $\mathbf{r}_1,\ldots,\mathbf{r}_n$, $y(\mathbf{r}_i)$ being the value of the analytical expansion (14) at the point $\mathbf{r}_i$ and $y_{\mathbf{r}_i}$ the numerical field value at the corresponding grid point.

To find the best possible approximation to a solution for this set of equations, we consider the following defect function χ, the minimum of which shall define the ''best'' set of coefficients a_i:

$$\chi=\sum_{r_i}\left(\frac{y_{r_i}-y(\mathbf{r}_i)}{s_i}\right)^2. \tag{16}$$

The hereby introduced variables s_i give us the possibility to assign a statistical weight to each grid point. To find the minimum of Eq. (16), we set its derivative to zero, giving us the equation

$$\beta_i=\sum_{j=1}^{m}\alpha_{ij}a_j, \tag{17}$$

where

$$\alpha_{ij}=\sum_{k=1}^{n}\frac{f_i(\mathbf{r}_k)f_j(\mathbf{r}_k)}{s_k^2}, \tag{18}$$

$$\beta_i=\sum_{k=1}^{n}\frac{y_{\mathbf{r}_k}f_i(\mathbf{r}_k)}{s_k^2}. \tag{19}$$

The rank of this matrix equation is equal to the number of functions that we have incorporated into our expansion.

We now have three choices to make: We need (a) to choose a system of functions to expand in, (b) to decide to what degree to expand to, and finally (c) to choose the statistical weights s_i.

(a) Choice of a set of functions. When choosing a system of functions, we have to check the following criteria. Our functions should guarantee that (i) they can represent in principle arbitrary fluid fields and (ii) the representation converges as fast as possible when the order of the system is increased. We found these conditions best satisfied by orthogonal function systems and used in particular the Chebychev polynomials.

In the case of a single cylinder there exists an analytical near-field approximation [19]

$$v_x=\frac{C}{4}(r^2-a^2)\frac{\partial^2}{\partial x\partial y}\ln r=\frac{C}{2r^2}(r^2+a^2)\cos(\theta)\sin(\theta), \tag{20}$$

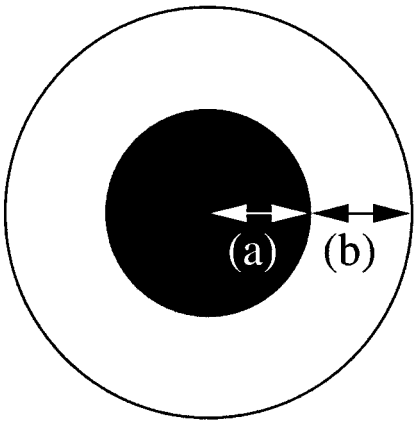

FIG. 1. Determination of the ring surrounding the particle in which we take the grid sites into consideration: (a) particle radius and (b) width of the fitting annulus, which is generally of the order of 1–2 particle radii.

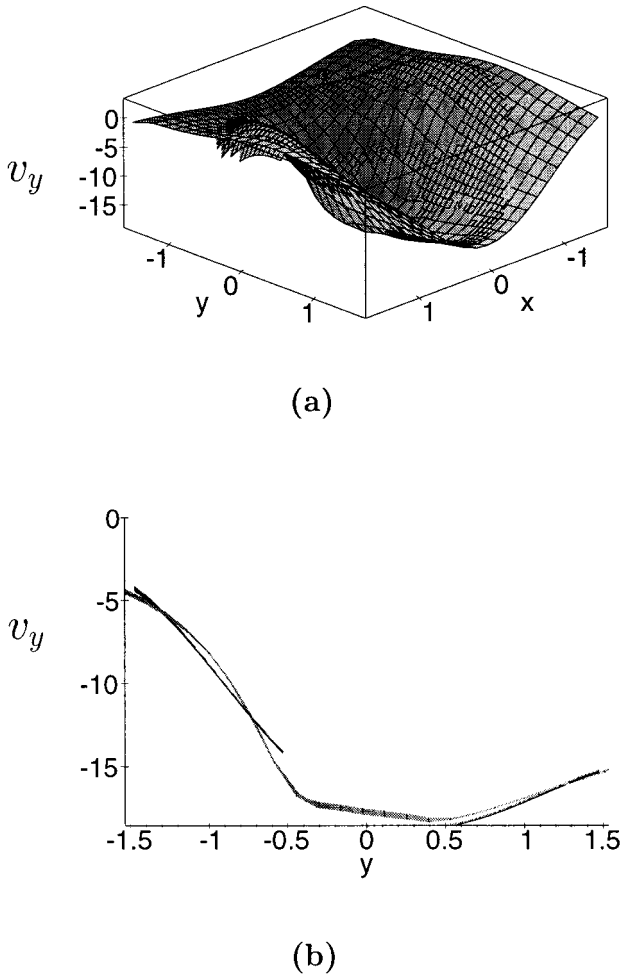

FIG. 2. (a) Numerical data (coarse grid) and the analytical approximation (fine grid) around a single cylinder of radius $a=0.5$ cm falling at terminal velocity. The values x and y are spatial coordinates relative to the cylinder axis in units of cm. The vertical axis shows the y (vertical) velocity component v_y in units of cm/s. The analytical function is only shown within the ring used to calculate the coefficients. In (b) we show a section from (a) along $x=0$.

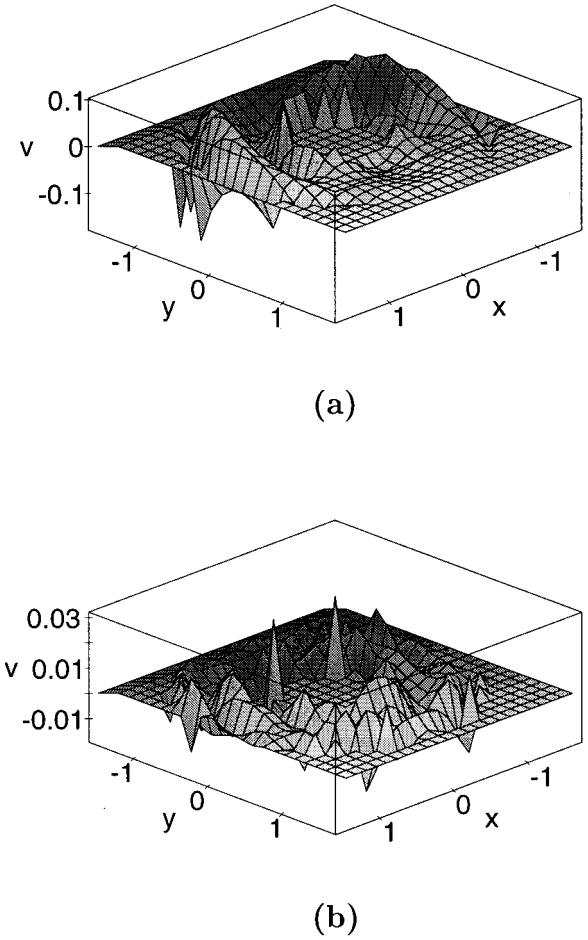

FIG. 3. (a) Difference of the velocity ''landscapes'' shown in Fig. 2(a). Here the vertical axis has been scaled by the modulus of the particle velocity, showing deviations of about 10%, the horizontal axis is the same as in Fig. 2(a). (b) Same result, but calculated with a Chebychev expansion to $o=10$ employing 66 fitted coefficients. Here the resulting deviation is less than 3%.

$$v_y=\frac{C}{2}\left(\gamma-\frac{1}{2}+\ln\frac{1}{2}\,kr+\frac{1}{2}\,(r^2-a^2)\,\frac{\partial^2}{\partial y^2}\ln r\right)$$
$$=\frac{C}{4r^2}\left(2\gamma r^2-a^2+2r^2\ln\frac{1}{2}\,kr-2(r^2+a^2)\cos^2(\theta)\right), \tag{21}$$

where

$$k=\frac{Ua}{2\nu},\quad C=\frac{2U}{\frac{1}{2}-\gamma-\ln\frac{1}{2}ka}, \tag{22}$$

and $\gamma\approx 0.577\,216$ is Euler's constant.

These functions could serve as a base for a systematic expansion leading to a system of functions consisting of spherical harmonics multiplied by increasingly complex radial functions, which is well suited to the symmetry of the problem. However, for two reasons this approach has not been taken in this paper. (i) Our main goal is the study of many-particle systems where the underlying symmetry (cylinder in parallel flow) is no longer satisfied, but, more importantly, (ii) this set of functions proved to have numerical stability problems. For comparison we show that the results of this ansatz in Sec. III C are very close to those obtained with Chebychev polynomials.

The Chebychev polynomials obey the recursive definition

$$T_0(x)=1, \tag{23}$$

$$T_1(x)=x, \tag{24}$$

$$T_{n+1}(x)=2xT_n(x)-T_{n-1}(x),\quad n\geqslant 1. \tag{25}$$

They are orthogonal over the interval $[-1,1]$ with respect to the L_2 norm and also assume values in $[-1,1]$. We can now rescale these polynomials with the variable substitution $y=(x-l)/l$, so that they are orthogonal over the interval $[-l,l]$ in which we fit our expansion to the numerical data. The value of l is given by the sum of the particle radius and the size of a fitting annulus, as described in the following paragraph. The expansions for the liquid fields to order o can now be written as

$$v_x(\mathbf{r})=\sum_{i=0}^{o-1}\sum_{j=0}^{i} c_{x_{(oi+j+1)}}T_j(x)T_{i-j}(y),$$

$$v_y(\mathbf{r})=\sum_{i=0}^{o-1}\sum_{j=0}^{i} c_{y_{(oi+j+1)}}T_j(x)T_{i-j}(y),$$

$$p(\mathbf{r})=\sum_{i=0}^{o-1}\sum_{j=0}^{i} c_{p_{(oi+j+1)}}T_j(x)T_{i-j}(y). \tag{26}$$

(b) Order of the expansion. In choosing the number of coefficients we want to incorporate into our approximation, we have to take the following considerations into account. As we have to solve a system of linear equations of the order of the number of coefficients, the computational costs increase rapidly with the number of coefficients chosen. Furthermore, we have to increase the number of grid sites used in solving the equation, to at least avoid an underdetermined system of equations. This can be achieved either by increasing the circle around the particle in which we consider the grid points, causing us to take grid sites into account that are heavily influenced by other particles or the walls, or by using a finer grid, which again increases greatly the computational cost of the algorithm. Unless otherwise stated, we have used an expansion up to polynomials of degree 5, which results in 21 coefficients. We place an annulus of width b around the particle as shown in Fig. 1 and use all grid points within this ring to set up Eqs. (26).

(c) Choosing the s_is. As to the choice of the s's, the natural choice of unity leads to the highest statistical weight of rings furthest away from the particle, as these rings contain more points. However, as the area closest to the particle is the most relevant, we have decided to set $s \propto r^2$, where r is the distance from the center of the particle, thus giving the same weights to all annuli.

The coefficients are calculated numerically by solving Eq. (17) at each time step for each particle. An example of the fits obtained for a single falling cylinder under the conditions described in Sec. III A are shown in the following figures. Figure 2 shows both the numerical and analytical data for the vertical y component of the velocity field. The analytical approximation is shown only in the ring used to fit the coefficients, i.e., not for the interior of the particle. Figure 3 shows the difference between the numerical data and the fitted function, for an expansion to order 5 as well as to order 10.

Substituting our expansion (26) into the expression for the stress tensor (12), we can now calculate the force on the particle by integrating the stress tensor over the particle surface (13). The resulting force on each particle is an expression in the coefficients c_x, c_y, c_p. For $o=5$ it is given by

$$\begin{aligned}
F_x=&\frac{3a^4\pi\mu c_{y_{12}}}{l^4}+\frac{6a^4\pi\mu c_{x_{13}}}{l^4}-\frac{a^6\pi c_{p_{20}}}{l^5}+\frac{2a^4\pi c_{p_{20}}}{l^3}-\frac{5a^2\pi c_{p_{16}}}{l}+\frac{3a^2\pi c_{p_7}}{l}-\frac{a^6\pi c_{p_{18}}}{l^5}+\frac{a^2\pi\mu c_{y_5}}{l^2}-\frac{3a^2\pi\mu c_{y_{14}}}{l^2}\\
&-\frac{3a^2\pi\mu c_{y_{12}}}{l^2}-\frac{10a^6\pi c_{p_{16}}}{l^5}-\frac{a^4\pi c_{p_9}}{2l^3}+\frac{9a^4\pi c_{p_{18}}}{2l^3}-\frac{a^2\pi c_{p_2}}{l}-\frac{3a^2\pi c_{p_{18}}}{l}-\frac{a^2\pi c_{p_{20}}}{l}+\frac{3a^4\pi\mu c_{y_{14}}}{l^4}+\frac{8a^2\pi\mu c_{x_4}}{l^2}\\
&-\frac{3a^4\pi c_{p_7}}{l^3}+\frac{48a^4\pi\mu c_{x_{11}}}{l^4}-\frac{16a^2\pi\mu c_{x_{15}}}{l^2}-\frac{32a^2\pi\mu c_{x_{11}}}{l^2}+\frac{24a^4\pi\mu c_{x_{15}}}{l^4}\\
&+\frac{4a^2\pi\mu c_{x_6}}{l^2}-\frac{12a^2\pi\mu c_{x_{13}}}{l^2}+\frac{15a^4\pi c_{p_{16}}}{l^3}+\frac{a^2\pi c_{p_9}}{l},\\
F_y=&-\frac{a^2\pi c_{p_3}}{l}-\frac{5a^2\pi c_{p_{21}}}{l}+\frac{3a^2\pi c_{p_{10}}}{l}-\frac{10a^6\pi c_{p_{21}}}{l^5}-\frac{3a^2\pi\mu c_{x_{12}}}{l^2}-\frac{3a^2\pi\mu c_{x_{14}}}{l^2}+\frac{a^2\pi\mu c_{x_5}}{l^2}\\
&+\frac{3a^4\pi\mu c_{x_{14}}}{l^4}+\frac{3a^4\pi\mu c_{x_{12}}}{l^4}+\frac{48a^4\pi\mu c_{y_{15}}}{l^4}+\frac{6a^4\pi\mu c_{y_{13}}}{l^4}+\frac{4a^2\pi\mu c_{y_4}}{l^2}+\frac{8a^2\pi\mu c_{y_6}}{l^2}-\frac{a^2\pi c_{p_{17}}}{l}\\
&+\frac{24a^4\pi\mu c_{y_{11}}}{l^4}-\frac{3a^2\pi c_{p_{19}}}{l}-\frac{a^6\pi c_{p_{17}}}{l^5}-\frac{3a^4\pi c_{p_{10}}}{l^3}-\frac{a^6\pi c_{p_{19}}}{l^5}+\frac{2a^4\pi c_{p_{17}}}{l^3}+\frac{9a^4\pi c_{p_{19}}}{2l^3}-\frac{16a^2\pi\mu c_{y_{11}}}{l^2}\\
&-\frac{a^4\pi c_{p_8}}{2l^3}+\frac{15a^4\pi c_{p_{21}}}{l^3}-\frac{32a^2\pi\mu c_{y_{15}}}{l^2}-\frac{12a^2\pi\mu c_{y_{13}}}{l^2}+\frac{a^2\pi c_{p_8}}{l},
\end{aligned} \tag{27}$$

where a is the radius, l the ring size, and μ the viscosity.

2. Implementation of the no-slip boundary condition

To obtain the proper particle-liquid interaction, we need to implement the no-slip liquid boundary condition on each particle surface. This seemingly easy task poses quite a few difficulties, especially in the case of high volume fraction, where we have closely packed moving boundaries throughout the system. In order to preserve numerical efficiency, an approximative approach has been chosen to implement these boundary conditions.

Let us first consider the simplest case. All liquid grid velocities located ''inside'' a particle are forced to equal the particle velocity before we enter the evaluation of $\mathbf{v}^*$ in Eq. (7), while all other points retain the old values. The ''interior'' points will then be updated after the pressure determination in the same manner as all other points, in order to assure incompressibility. This procedure obviously leads to a slight change in the velocity of the liquid inside the particle, so that after the iteration, it results in velocity values between the velocity of the particle and that of the surrounding liquid.

We could more closely approximate the no-slip boundary condition by requiring that the liquid velocity for the grid points inside the particle v^l be equal to the particle velocity v^p *after* the liquid iteration. To this end, we set the liquid velocity inside a particle equal to some factor α times the particle velocity v^p, $v_0^l=\alpha v^p$. The parameter α is now chosen so that $v_{\Delta t}^l=v^p$. The multiplier α is set to unity at $t=0$ and is then adjusted according to $d\alpha/dt=\dot{\alpha}(v_p-v_l)$ at every time step until $v_{\Delta t}^l=v^p$. Unfortunately, this procedure would introduce a purely numerical parameter $\dot{\alpha}$ so that we have decided to use rather the parameter-free approach as described before. We note that our method generally leads to a slight systematic underestimation of the forces on the particle corresponding to an apparent radius smaller than the real one.

B. Short-range particle-particle interaction

Lubrication theory predicts that for low Reynolds numbers there are diverging forces between particle pairs that in principle suffice to avoid contact between them. However, the finite grid spacing in the above described algorithm cannot exactly resolve this behavior when particle surface distance become close. Moreover, since lubrication theory breaks down at finite Reynolds numbers and since real particles are not perfectly smooth on the surface, one will always have to deal with the case of particle contacts.

Thus, to avoid unphysical overlap between particles that otherwise could result in these situations, we have incorporated an elastic, central, and repulsive particle-particle interaction that acts when two particles i and j touch each other [20–23]. That is, the forces between two particles are zero unless the distance between their two centers is less than the sum of the radii r_i+r_j. Then the acting force on i is assumed to be

$$\mathbf{F}_i^{\mathrm{el}}=-k_n[r_i+r_j-(\mathbf{x}_j-\mathbf{x}_i)\cdot\hat{\mathbf{n}}_{ji}]\hat{\mathbf{n}}_{ji}. \tag{28}$$

Here $\mathbf{x}_i=(x_i,y_i)$ is the position vector of the center of particle i and $\hat{\mathbf{n}}_{ji}$ is the unit vector pointing from the center of particle i to the center of j. The spring constant k_n can be chosen to reflect real material parameters leading, e.g., to contact durations between particles in the microsecond range as typical for metal beads of mass 10^{-3} kg. However, since the liquid properties are dominating the physical behavior of the system, we have selected $k_n\approx 2\times 10^{-2}m/(\Delta t)^2$, so that the duration of a contact is about 20 times the time step Δt of the simulation. In fact, we have observed that such close encounters are sufficiently rare so that the simulation results are not significantly altered by the exact functional form of $\mathbf{F}_i^{\mathrm{el}}$ or the value of k_n [9].

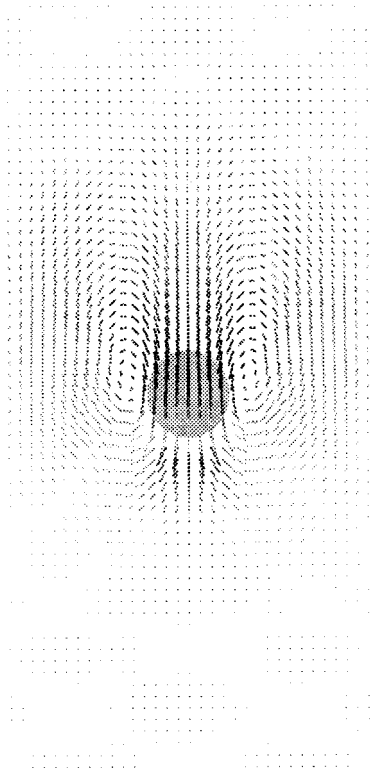

FIG. 4. Example of the flow field around a falling cylinder. There is one arrow per liquid grid site indicating the direction and magnitude of the liquid velocity. The system shown here is one-quarter of the linear size of the systems used to obtain the results.

III. MODEL VALIDATION

The comparison of this model with experimental and theoretical data proves to be a difficult task as both experimental and theoretical work are easier in three dimensions and thus little literature treating the two-dimensional case is found. Nonetheless, three relevant results in two dimensions, two experimental and one numerical, can be compared with the present method. The first two results concern falling cylinders and a typical setup can be seen in Fig. 4.

A. Drag coefficients as a function of the system size

One experimentally and numerically well-studied result is the relationship between the drag coefficient on a cylinder as a function of the width of the container in which it moves [24,25]. In simulating this situation, we have taken a closed container of fixed height twice its width with the simulation

TABLE I. List of fixed simulation parameters. Unless otherwise specified, simulations runs have been performed using the parameters in this table. These parameters where chosen to give reasonable Reynolds numbers. All given results are in nondimensional form.

Symbol	Quantity	Value	Unit
L_x	system width	10	cm
L_y	system height	20	cm
a	particle radius	0.5	cm
η	liquid shear viscosity	2.0	g/cm s
ρ_l	liquid density	1.0	g/cm^3
g	gravitational acceleration	980	cm/s^2
	grid spacing	0.16	cm

parameters shown in Table I. A single cylinder of varying diameter is placed two-thirds of the way up the container and allowed to fall under the influence of gravity. When the cylinder velocity reaches the steady state, the terminal velocity U is measured, permitting us to calculate the drag coefficient given by the equation

$$C_D = \frac{D}{\frac{1}{2}\rho_l U^2 2r}, \tag{29}$$

where D is the drag force exerted on the cylinder and its modulus is equal to that of the gravitational force mg.

In order to achieve similar terminal velocities and thus container Reynolds numbers for particles of all sizes, we vary the particle density with the diameter in the range from 1.025–1.4 g/cm^3. The resulting Reynolds number based on the system size varies in a range between 12 and 154, while the particle Reynolds number lies between 9.6 and 30.8.

The diameter of the cylinder is varied between 0.05 and 0.8 times the width of the container, thus giving the different values of λ reported in Fig. 5. For comparison, we have also plotted the experimental results obtained in [24].

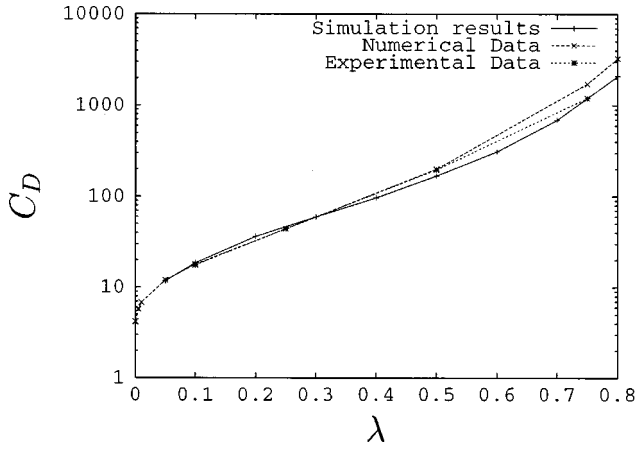

FIG. 5. Drag coefficient C_D (dimensionless) of a cylinder as a function of the relative particle size λ (dimensionless) given by the cylinder diameter divided by the box width (+). The other numerical (×) and experimental data (⋆) are taken from [24].

TABLE II. Parameters deviating from those in Table I used to obtain the single-particle drag coefficient C_D as a function of Re in Sec. III B.

Symbol	Quantity	Value	Unit
L_x	system width	18	cm
L_y	system height	26	cm
	grid spacing	0.14	cm

B. Drag coefficient as a function of Reynolds number

Due to the Stokes paradox, it is not possible to calculate the liquid field in the creeping flow limit of zero Reynolds number for a cylinder. Thus all results for the drag coefficient depend on the Reynolds number of the system.

We have studied this dependence by letting a cylinder fall inside a closed container. The different terminal velocities result in different Reynolds numbers and were achieved by varying the cylinder's density. All other parameters where held constant and are given in Table II. The results are presented in Fig. 6. These results compare well with experimental data in this regime described by Tritton's empiric relation [26].

C. Drag force on an array of cylinders

As a final validation of our model, we wish to see whether the many-particle interactions are correctly modeled. We therefore take a system of size 8×4 cm^2 consisting of an array of eight cylinders placed on a 128×64 grid as shown in Fig. 7. In the vertical direction we use periodic boundary conditions. In the horizontal direction we place four cylinders in each row and drive the system by a horizontal fluid flow imposed on the left and right boundaries. We now measure both the force on the central four cylinders and the average horizontal velocity down the center of the system.

To obtain results for different volume fractions, we vary the size of the cylinders, keeping all other parameters fixed. The results are presented in Fig. 8, together with numerical data from Sangani and Acrivos [27]. As can be seen, there is good agreement in the region of low to intermediate volume fraction. Towards high volume fractions, our method results

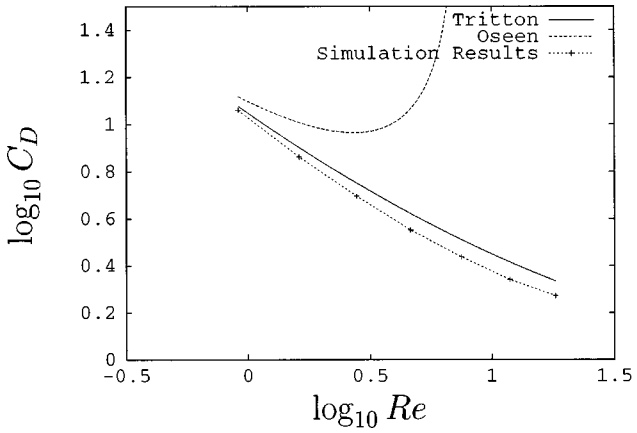

FIG. 6. Drag coefficient of a cylinder as a function of the Reynolds number. For comparison, we have also plotted Oseen's equation, valid for low Re, and the Tritton empiric relation [26].

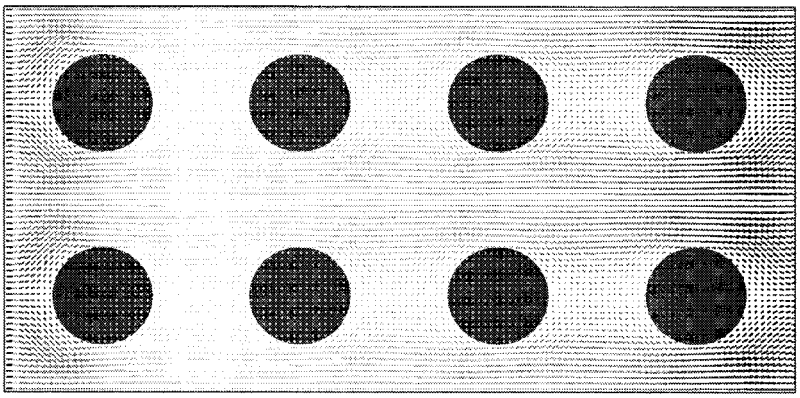

FIG. 7. Example of the flow field through an array of cylinders at a volume fraction of 20%. The horizontal velocity is fixed at both ends and the average velocity is measured down the middle of the system. The boundary is periodic in vertical direction.

in a systematically lower force. This is due to our implementation of the no-slip boundary condition (Sec. II A 2), which results in an effective particle radius slightly smaller than the real one. This effect is more pronounced for the dense systems, as the resulting error in the size of the gap through which fluid can flow is much larger. This effect can be systematically eliminated by taking a finer mesh, also shown in Fig. 8.

IV. COMPUTATIONAL EFFORT

The computational effort to obtain the results presented in this paper has been comparatively moderate. Asymptotically, both computation time and memory requirements scale *linearly* in the number of grid points and the number of particles. The current version of our algorithm requires approximately 15 μs of update time and 150 bytes of memory per grid point on a SUN ULTRA workstation as measured in the simulations for the regular cylinder array (Fig. 7) and in *dynamical* simulations with systems of about 4000 disks. By far the largest fraction of CPU time is used by the Navier-Stokes code and is thus determined by the number of grid points. The usage per particle is an additional 300 μs per update and 100 bytes of memory.

The code permits us to study systems of more than 1000 particles on average workstations with satisfactory accuracy. The results of these simulations together with three-dimensional data are planned to be reported in a forthcoming paper.

V. CONCLUSION AND OUTLOOK

We have presented a fast simulation method for particle-liquid systems. The results obtained with our method show a high degree of quantitative agreement with experimental and other numerical results in the cases presented in this paper. This is an important improvement over previous work on this subject [8], as we have been able to retain the high efficiency of the algorithm while improving the results to quantitatively correct levels.

Although the regular mesh has the computational advantage of requiring only simple data structures and small update times per point, it should be noted that it cannot provide the adaptive refinements necessary to resolve in detail the flow in the narrow gap between two particles as unstructured-grid methods do. At high particle concentrations, where lubrication or other surface-surface viscous effects become important, one therefore needs additional modeling assumptions and it is not yet fully clear how and whether these can be integrated with the proposed mesh flow equations (cf., e.g., [9]).

Another difficulty arises in the context of sheared flows, where the edges of a rigid regular grid cannot be easily matched together without losing accuracy by the introduction of some interpolation scheme providing the required boundary velocity values at non-grid-site locations. In contrast, the introduction of periodic boundary conditions along one or two directions is rather straightforward. Although the pressure values determined by the multigrid code lack the linear component, that in the presence of material walls otherwise opposes the acceleration of the system as a whole; this deficiency can be remedied by a compensating volume force term in Eq. (1).

The rather small deviations of our simulations from known results found in this paper do not improve much by increasing the grid resolution or the order of the equations. Apparently we resolve the important structures of the fluid flow with our grid resolution and also our analytical approximation closely fits numerical data for $o=5$. The deviations are due rather to the approximate implementation of the particle surface in the fluid. This is where the main potential for

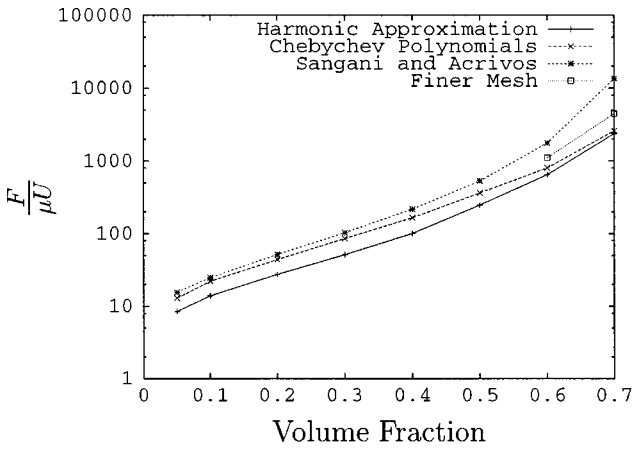

FIG. 8. Nondimensional drag force as a function of the volume fraction (dimensionless) of cylinders. These calculations have been performed for harmonic functions (+) and Chebychev polynomials ($\times$) as described in Sec. II A 1. The ''finer mesh'' system ($\square$) has a fluid grid twice as fine as the mesh in the normal Chebychev simulation. We also show numerical results from Sangani and Acrivos [27] ($\star$).

improvement of the described algorithm lies. Possible improvements of our method consist in a modification of the current treatment of boundary conditions along the lines described in Sec. II A 2. As we found, this modification achieves quantitatively better results, but not to a degree that would justify the introduction of an additional purely numerical model parameter. Another possibility, however, suffering from similar deficiencies is to increase the effective particle radius by setting liquid velocity points close to the particle to a value determined as a function of the velocities of the particle and the liquid in the previous time step. Furthermore, one could think of implementing *no-slip* boundary conditions in a similar fashion as the systems boundaries, either to first-order accuracy as in [25] or even higher order, at the price of a significant loss in performance due to the added complexity of the algorithm.

Based on the verification presented in this paper, we can now proceed to larger systems studying many-particle effects. A further and rather straightforward development is the extension of the algorithm to three dimensions, giving us further possibilities to compare with known results.

ACKNOWLEDGMENTS

The numerical calculations have been performed on the SP 2 computers of the ZAM, Forschungszentrum Jülich and of the GMD in Bonn as well as the RS/6000 cluster of the RUS, Universität Stuttgart. W.K. would like to thank the HLRZ, Forschungszentrum Jülich for funding and the DAAD for a research grant. S.S. gratefully acknowledges financial support by NATO. We are indebted for many useful discussions and help to G. Brady, F. Feuillebois, J. Hinch, S. Roux, and the group of G. Wittum at the ICA 3, Universität Stuttgart.

[1] J. Feng, H. H. Hu, and D. D. Joseph, J. Fluid Mech. **261**, 95 (1994); **277**, 271 (1994).
[2] H. H. Hu, Int. J. Multiphase Flow **22**, 335 (1996); H. H. Hu, D. D. Joseph, and M. J. Crochet, Theor. Comput. Fluid Dyn. **3**, 285 (1992).
[3] A. A. Johnson and T. E. Tezduyar, Comput. Methods Appl. Mech. Eng. **134**, 351 (1996).
[4] W. L. Wendland and J. Zhu, Math. Comput. Model. **15**, 19 (1991).
[5] A. Sangani and G. Mo, Phys. Fluids **6**, 1653 (1994).
[6] Y. Tsuji, T. Tanaka, and T. Ishida, Powder Technol. **71**, 239 (1992).
[7] S. Yonemura, T. Tanaka, and Y. Tsuji, ASME FED **166**, 303 (1993).
[8] W. Kalthoff, S. Schwarzer, G. Ristow, and H. Herrmann, Int. J. Mod. Phys. C **7**, 543 (1996).
[9] S. Schwarzer, Phys. Rev. E **52**, 6461 (1995).
[10] J. F. Brady and G. Bossis, Annu. Rev. Fluid Mech. **20**, 111 (1988).
[11] G. Bossis and J. F. Brady, J. Chem. Phys. **91**, 1866 (1989).
[12] M. Sommerfeld, ASME FED **121**, 213 (1991).
[13] T. Tanaka, T. Kawaguchi, S. Nishi, and Y. Tsuji, ASME FED **166**, 17 (1993).
[14] A. Ladd, J. Fluid Mech. **271**, 285 (1994).
[15] R. Peyret and T. D. Taylor, *Computational Methods for Fluid Flow* (Springer-Verlag, New York, 1983).
[16] A. J. Chorin, J. Comput. Phys. **2**, 12 (1967).
[17] H. Kopetsch, *20. IFF Ferienkurs: Computersimulationen in der Physik* (KFA Jülich, Jülich, 1989).
[18] W. Hackbusch, *Multi-Grid Methods and Applications* (Springer-Verlag, New York, 1985).
[19] H. Lamb, *Hydrodynamics*, 6th ed. (Cambridge University Press, Cambridge, 1932).
[20] Y. Tsuji, T. Kawaguchi, and T. Tanaka, Powder Technol. **77**, 79 (1993).
[21] P. A. Cundall and O. D. Strack, Geotechnique **29**, 47 (1979).
[22] G. H. Ristow, Int. J. Mod. Phys. C **3**, 1281 (1992).
[23] O. R. Walton and R. L. Braun, J. Rheol. **30**, 949 (1986).
[24] R. Bouard and M. Coutaceau, J. Appl. Math. Phys. **37**, 673 (1986).
[25] G. H. Ristow, Comput. Phys. Commun. **99**, 43 (1996).
[26] D. Tritton, J. Fluid Mech. **6**, 547 (1959).
[27] A. Sangani and A. Acrivos, Int. J. Multiphase Flow **8**, 193 (1982).