Available online at www.sciencedirect.com
SCIENCE DIRECT®
Journal of Computational Physics 187 (2003) 371–390
JOURNAL OF COMPUTATIONAL PHYSICS
www.elsevier.com/locate/jcp

PHYSALIS: a new method for particle simulation Part II: two-dimensional Navier–Stokes flow around cylinders

S. Takagi [a], H.N. Oğuz [b], Z. Zhang [b], A. Prosperetti [b,c,*]

[a] *Department of Mechanical Engineering, The University of Tokyo, Hongo, Bunkyo-ku, Tokyo 113, Japan*
[b] *Department of Mechanical Engineering, The Johns Hopkins University, Baltimore, MD 21218, USA*
[c] *Faculty of Applied Physics and Twente Institute of Mechanics, University of Twente, AE 7500 Enschede, The Netherlands, Burgerscentrum, The Netherlands*

Received 27 September 2001; received in revised form 16 January 2003; accepted 29 January 2003

Abstract

This paper presents a new approach to the direct numerical simulation of particle flows. The basic idea is to use a local analytic representation valid near the particle to "transfer" the no-slip condition from the particle surface to the adjacent grid nodes. In this way the geometric complexity arising from the irregular relation between the particle boundary and the underlying mesh is avoided and fast solvers can be used. The results suggest that the computational effort increases very slowly with the number of particles so that the method is efficient for large-scale simulations. The focus here is on the two-dimensional case (cylindrical particles), but the same procedure, to be developed in forthcoming papers, applies to three dimensions (spherical particles). Several extensions are briefly discussed.

Keywords: Flow around cylinders; Particle flow; Multiphase flow

1. Introduction

In a recent paper [1] we have presented a new approach, termed *PHYSALIS*, for the numerical computation of potential flows around spheres. Here, that basic approach is extended to the Navier–Stokes problem for two-dimensional flow past aligned cylinders, a situation for which some preliminary results were recently presented in [2]. It will be apparent from the following that the same idea can be further extended to the three-dimensional viscous flow past spheres, and work in this direction is currently under way; some early results for the three-dimensional case were presented in [3].

At the root of the *PHYSALIS* approach is a new way to deal with the geometrical complexity that the particles introduce in the calculation: because of the boundary conditions on its surface, a particle induces a specific structure in the neighboring flow which manifests itself in certain non-local relations among the

* Corresponding author. Tel.: +1-410-516-8534; fax: +1-410-516-7254.
E-mail address: prosper@landau.me.jhu.edu (A. Prosperetti).

0021-9991/03/$ - see front matter
doi:10.1016/S0021-9991(03)00077-9

flow fields. If this structure is enforced directly, the flow "becomes aware" of the presence of the particle with no need to account for the particle surface explicitly. One can then use a regular grid and, effectively, remove the particle as an internal boundary from the computation. If the flow takes place in a regular domain, this circumstance permits the use of highly efficient direct solvers for the pressure Poisson equation. Here we will describe two implementations of this idea and, in the last section, we briefly indicate how the same approach may be extended to particles with an arbitrary shape.

The most useful feature of the method is the fact that, for a given computational domain, the computation time is only weakly dependent on the total number of particles, which permits relatively large computations to be carried out with relatively modest computational resources. Additionally, unlike other methods, the no-slip condition at the particle surface is enforced exactly and, as the number of degrees of freedom per particle is increased, the error decreases faster than algebraically. A preliminary analysis of the convergence properties of this approach is presented in [4] (see also [5]).

The direct numerical simulation of disperse particle flows is an important topic in contemporary multiphase flow research due to its relevance for a variety of applications such as slurries, fluidized beds, pneumatic transport, suspensions, and many other industrial and environmental problems. In a large fraction of this work, the dispersed particles are approximated as points (see, e.g. [6–8]), a procedure that is only valid for small particles at dilute concentrations. Methods for finite-size particles are less well developed and, in general, require considerable computational resources. Some recent examples include [9–17]. An alternative approach, based on the lattice Boltzmann method, has also been developed in the last few years [18,19].

2. Reduction to the rest frame

As will be clear in the following, the method requires the consideration of the flow in the neighborhood of each particle separately. Let $\mathbf{U}$ be the flow velocity in the inertial frame and $\mathbf{w}$, Ω the translational and angular velocity of a particle. The first step is to express the Navier–Stokes equations in the particle rest frame, where the flow velocity $\mathbf{u}$ is related to $\mathbf{U}$ by

$$\mathbf{U} = \mathbf{u} + \mathbf{w} + \Omega \times \mathbf{x}, \tag{1}$$

in which $\mathbf{x}$ is the position relative to the particle center. In this frame, we have

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

$$\rho\left[\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u}\cdot\nabla)\mathbf{u} + 2\Omega \times \mathbf{u}\right] = -\nabla p + \mu\nabla^2\mathbf{u} + \rho\mathbf{g} - \rho\left[\dot{\mathbf{w}} + \dot{\Omega} \times \mathbf{x} + \Omega \times (\Omega \times \mathbf{x})\right], \tag{3}$$

with the boundary condition

$$\mathbf{u} = 0 \tag{4}$$

on the particle surface. In (3) ρ and μ are the fluid density and viscosity, p is the pressure, and $\mathbf{g}$ the body force; dots denote Lagrangian time derivatives following the particle.

We now let

$$\mathbf{u} = \mathbf{v} + \tilde{\mathbf{u}}, \quad p = \tfrac{1}{2}\rho(\Omega \times \mathbf{x})^2 - \rho(\dot{\mathbf{w}} - \mathbf{g})\cdot\mathbf{x} + q + \tilde{p}, \tag{5}$$

where the fields $(\mathbf{v}, q)$ are chosen to satisfy

$$0 = -\nabla q + \mu\nabla^2\mathbf{v} - \rho\dot{\Omega} \times \mathbf{x}, \quad \nabla \cdot \mathbf{v} = 0, \tag{6}$$

with

$$\mathbf{v} = 0 \tag{7}$$

on the particle surface. As a consequence, the new fields $(\tilde{\mathbf{u}}, \tilde{p})$ satisfy

$$\rho\left[\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u}\cdot\nabla)\mathbf{u} + 2\Omega\times\mathbf{u}\right] = -\nabla\tilde{p} + \mu\nabla^2\tilde{\mathbf{u}}, \tag{8}$$

with

$$\tilde{\mathbf{u}} = 0 \tag{9}$$

on the particle surface. Since $\mathbf{u} = 0$ on the particle surface, by continuity, it will be small near the particle and, therefore, there is a region adjacent to the particle where the left-hand side of (8) is small. Thus, locally, $(\tilde{\mathbf{u}}, \tilde{p})$ approximately satisfy

$$-\nabla\tilde{p} + \mu\nabla^2\tilde{\mathbf{u}} = 0, \quad \nabla\cdot\tilde{\mathbf{u}} = 0, \tag{10}$$

i.e., the Stokes equations. Naturally, the extent of the spatial region where (10) are a good approximation to (8) becomes smaller and smaller as the Reynolds number increases but, for any finite Reynolds number, there is a non-vanishing region where (10) are applicable with but a small error.

The general solution of the Stokes problem (10) can be expressed analytically in terms of coefficients that depend on the flow incident on the particle. We shall use this fact to transform the boundary condition at the particle surface into a relation among the values of the flow fields at the neighboring grid points, as will be described in Sections 4 and 7.

The solution of the auxiliary problem (6) is readily found in both two and three dimensions; in the present two-dimensional case, it is

$$\mathbf{v} = \frac{r^4 - a^4}{8\nu r^2}\dot{\Omega}\times\mathbf{x}, \quad q = 0. \tag{11}$$

From a knowledge of $\tilde{\mathbf{u}}$ and $\tilde{p}$, the force $\tilde{\mathbf{F}}$ and couple $\tilde{\mathbf{L}}$ acting per unit length on the particle can be calculated from the usual definitions:

$$\tilde{\mathbf{F}} = \int \mathrm{d}S\Big(-\tilde{p}\mathbf{n} + \tilde{\boldsymbol{\tau}}\cdot\mathbf{n}\Big), \quad \tilde{\mathbf{L}} = \int \mathrm{d}S\mathbf{x}\times\Big(-\tilde{p}\mathbf{n} + \tilde{\boldsymbol{\tau}}\cdot\mathbf{n}\Big), \tag{12}$$

in which the integration is over the particle surface, $\mathbf{n}$ is the unit outward normal, and $\tilde{\boldsymbol{\tau}}$ the viscous stress tensor associated with $\tilde{\mathbf{u}}$. These quantities are related to the force $\mathbf{F}$ and couple $\mathbf{L}$ in terms of the original variables by

$$\mathbf{F} = \rho v(\dot{\mathbf{w}} - \mathbf{g}) + \tilde{\mathbf{F}}, \quad \mathbf{L} = \tilde{\mathbf{L}} + \rho v a^2\dot{\Omega}, \tag{13}$$

where $v = \pi a^2$ is the volume (area) of a particle of radius a. In the examples shown in this paper, the translational and angular velocity of the particles are specified. In the more general case, these quantities are to be found from the standard equations of rigid-body mechanics; several examples of this type are given in [5].

3. General solution of the two-dimensional stokes equations

The general solution of the Stokes equations (10) subject to the boundary condition (9) is readily found by introducing a dimensionless stream function ψ, defined so that $\tilde{\mathbf{u}} = \nu\nabla\times(\psi\mathbf{k})$ (with ν the kinematic

viscosity and **k** the unit vector normal to the flow plane), and using a Fourier series representation in the angular variable; it is

$$\psi = \left(s^2 - 2\log s - 1\right)A_0 + \sum_{n=1}^{\infty}\left[\psi_n\left(A_n\cos n\theta + \tilde{A}_n\sin n\theta\right) + \tilde{\psi}_n\left(B_n\cos n\theta + \tilde{B}_n\sin n\theta\right)\right], \tag{14}$$

where

$$\psi_1 = s^3 - 2s + \frac{1}{s}, \quad \tilde{\psi}_1 = s\log s - \frac{s}{2} + \frac{1}{2s}, \tag{15}$$

$$\psi_n = ns^{n+2} - (n+1)s^n + s^{-n}, \quad \tilde{\psi}_n = ns^{-n+2} - (n-1)s^{-n} - s^n, \tag{16}$$

with $s = r/a$. An important point to stress is that, thanks to the as yet undetermined coefficients $A_n, B_n, \tilde{A}_n, \tilde{B}_n$, the stream function (14) is able to accommodate any (locally Stokes) flow in the neighborhood of the particle. Thus, no assumptions or restrictions about this flow have been introduced. As explained below, these coefficients are determined iteratively by matching the velocity field given by (14) to the numerically computed Navier–Stokes flow away from the particle.

The corresponding pressure field is

$$\tilde{p} = p_0 + \frac{\mu v}{a^2}\left\{8s\left(-A_1\sin\theta + \tilde{A}_1\cos\theta\right) + \frac{2}{s}\left(B_1\sin\theta - \tilde{B}_1\cos\theta\right)\right.$$
$$\left. + \sum_{n=2}^{\infty}\left[4n(n+1)s^n\left(-A_n\sin n\theta + \tilde{A}_n\cos n\theta\right) + 4n(n-1)s^{-n}\left(\tilde{B}_n\cos n\theta - B_n\sin n\theta\right)\right]\right\}, \tag{17}$$

where p_0 is a constant, different for each particle, and the vorticity field

$$\omega = \frac{v}{a^2}\left\{-4A_0 - 8s\left(A_1\cos\theta + \tilde{A}_1\sin\theta\right) - \frac{2}{s}\left(B_1\cos\theta + \tilde{B}_1\sin\theta\right)\right.$$
$$\left. - \sum_{n=2}^{\infty}\left[4n(n+1)s^n\left(A_n\cos n\theta + \tilde{A}_n\sin n\theta\right) + 4n(-n+1)s^{-n}\left(B_n\cos n\theta + \tilde{B}_n\sin n\theta\right)\right]\right\}. \tag{18}$$

The contributions to the hydrodynamic force and couple per unit length on the cylinder given by (14) are

$$\tilde{F}_x = 4\pi\frac{\mu v}{a}\tilde{B}_1, \quad \tilde{F}_y = -4\pi\frac{\mu v}{a}B_1, \quad \tilde{L} = -8\pi\mu v A_0. \tag{19}$$

If the previous expansions are truncated to the first term, the present approach might appear to have some similarity with the force-coupling method of Maxey and Patel [16]. Actually, even with this severe truncation, the two methods are different. As can be seen in Fig. 1 of [16], very near a sedimenting particle, that approach produces an error while ours is capable of reproducing the exact result. Furthermore, in Maxey and Patel's method, the no-slip boundary condition at the particle surface is not exactly satisfied.

Our approach shares with the so-called *CHIMERA* method (see, e.g. [20,21]), the matching of solution(s) valid near the particle(s) with a "global" solution valid elsewhere in the computational domain. However, our use of an analytical form for the local solution(s) replaces the low-order (algebraic) projection errors between the local and global grids which affect the *CHIMERA* method by high-order (spectral) ones.

4. Two-cage implementation

The general idea described in Section 1 can now be made more precise. We describe two different implementations, one using only the velocity field, the other (in Section 7) using velocity, pressure, and vorticity.

The entire domain, irrespective of the presence of the bodies, is covered by a regular finite-difference grid. Each cylinder is surrounded by two nested "cages" of grid points constructed so as to be near the body surface; an example is shown in Fig. 1. Here, the nodes of the outer cage (dots) are located at cell centers and the arrows are the velocity points on the inner cage.

The procedure can be summarized as follows. Suppose that a provisional estimate of the velocity field at the grid nodes is available (this could be, e.g., the velocity field at the previous time step). Then:

1. For each particle, let $J = 1, 2, \ldots, N_o$ be the nodes of the *outer cage*. Match the analytic expressions for the velocity obtained from the stream function (14) with the estimated velocity field at these grid nodes using (1):

$$\mathbf{u}_J = \mathbf{U}_J - \mathbf{w} - \Omega \times \mathbf{x}_J. \tag{20}$$

This system of equations is solved for the coefficients $A_n, B_n, \tilde{A}_n, \tilde{B}_n$ of the particle. In principle, the maximum number of coefficients that can be determined in this way equals the number of cage nodes, although in practice we use fewer and solve (20) in a least square sense;

2. Using the values of the coefficients determined at the previous step, compute from the analytic formulae the velocity field at the points $K = 1, 2, \ldots, N_i$ of the *inner cage* (Fig. 1) surrounding each particle:

$$\mathbf{U}_K = \mathbf{u}_K + \mathbf{w} + \Omega \times \mathbf{x}_K. \tag{21}$$

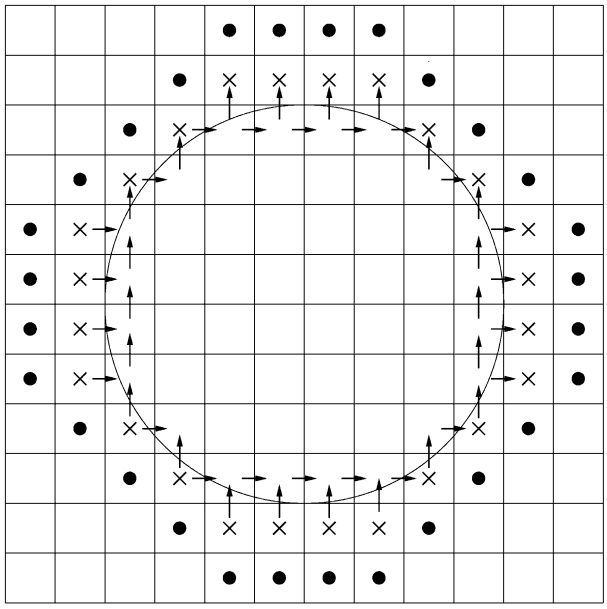

Fig. 1. An example of the inner and outer cages for a grid with 8 mesh lengths per cylinder diameter. The nodes of the outer cage used to calculate the coefficients of (14) are located at cell centers; the arrows are the velocity points on the inner cage; the crosses mark the points closest to the particle where the pressure Poisson equation is solved.

3. Solve the full Navier–Stokes equations on the finite-difference grid imposing this velocity field as boundary condition on the inner cage nodes of each particle.
4. Evaluate this updated velocity field at the nodes of the outer cage of each particle and return to step 1, repeating the cycle until convergence.

As will be explained in the following section, in executing step 3, it is efficient to solve for the flow field over the entire grid, disregarding the presence of the particles. The velocity field outside the inner cages is the one that is desired. The field inside the inner cages is the correct solution of another flow problem, in which the flow is driven by the imposed velocity on the cage nodes: this solution is not unphysical – it is simply irrelevant for the purposes of the calculation and can be disregarded. The final flow field is given by the finite-difference solution outside the inner cages, and by the analytic representation in the thin region between each particle and the surrounding inner cage. It should be stressed that the solution procedure is devised in such a way that, other than for satisfying a common velocity boundary conditions at the cage nodes, the inner and outer solutions are unrelated, so that any 'contamination' of the latter by the former is avoided. In particular, there is no relation satisfied by the stresses across the cage surface.

A possible criticism of the method is its reliance on an approximate solution in the fluid regions between each particle and the surrounding outer cage. The accuracy can be improved by refining the grid, which has the effect of putting the cage nodes closer and closer to the particle surface. Another possibility (which we have not yet explored) would be to improve on the Stokes flow solution by approximating the solution of the full nonlinear equation (8) by a regular perturbation expansion. It may be noted, however, that in practice some control of the error is built into the procedure, as convergence requires that the flow be described by the *same* set of coefficients (A_k, B_k) on *both* the inner and outer cages of each particle. This requirement would not be satisfied in the presence of strong nonlinear effects which are not accounted for in the analytic solution (14).

A rough idea of the grid size Δ necessary for a good numerical accuracy may be found by noting that the grid points should be inside the boundary layer for the Stokes approximation to be valid. If the boundary layer thickness is estimated as $a/\sqrt{Re}$, where Re is the Reynolds number expressed in terms of a characteristic velocity and the cylinder diameter, we thus have that $N_\Delta = 2a/\Delta$ should be sufficiently larger than $\sqrt{Re}$. This limit is not different from that applicable to a standard finite-difference calculation.

5. Miscellaneous details

While the general idea of the method is sufficiently clear from the description of the previous section, its successful implementation hinges on several details, to which we now turn.

5.1. Flow solver

As described in the previous section, each computational cycle consists of two steps, an inner one that updates the coefficients and an outer one that calculates the flow field for a given set of coefficients.

The outer step is executed by a standard finite-difference MAC discretization on a staggered grid and a suitably modified first-order accurate projection method. The time-discretized Navier–Stokes equation is written as

$$\frac{\mathbf{U}^{n+1} - \mathbf{U}^n}{\Delta t} = \Lambda(\mathbf{U}^n) - \frac{1}{\rho}\nabla p^{n+1}, \tag{22}$$

where Λ stands for all the terms that need not be shown explicitly; the superscripts n, $n+1$ denote time levels and we assume that everything is known at time level n. In the usual approach, one defines an auxiliary field $\mathbf{U}_*$ such that, at the interior nodes,

$$\frac{\mathbf{U}_* - \mathbf{U}^n}{\Delta t} = \Lambda(\mathbf{U}^n), \tag{23}$$

while

$$\mathbf{U}_* = \mathbf{U}^{n+1} \tag{24}$$

on the boundaries. With $\mathbf{U}_*$ thus defined, the momentum equation (22) is reduced to

$$\frac{\mathbf{U}^{n+1} - \mathbf{U}_*}{\Delta t} = -\frac{1}{\rho}\nabla p^{n+1}, \tag{25}$$

which will determine $\mathbf{U}^{n+1}$ once p^{n+1} is known. To obtain this latter quantity, one takes the divergence of (25), recalling the equation of continuity, to find a Poisson equation

$$\nabla^2 p^{n+1} = \frac{\rho}{\Delta t}\nabla \cdot \mathbf{U}_*, \tag{26}$$

to be solved subject to homogeneous Neumann conditions on the boundary. Here, we cannot execute the step (24) because $\mathbf{U}^{n+1}$ at the cage boundary must be calculated from the stream function (14), the coefficients $(A_k, B_k, \tilde{A}_k, \tilde{B}_k)$ of which are still unknown. Thus, we proceed iteratively as follows.

Denote by κ be the index of this iteration, and write, for brevity, $\mathbf{U}^\kappa$, p^κ, and $\mathbf{A}^\kappa$ in place of $\mathbf{U}^{n+1,\kappa}$, $p^{n+1,\kappa}$, and $(A_k^\kappa, B_k^\kappa, \tilde{A}_k^\kappa, \tilde{B}_k^\kappa)$. As in the usual method, we define $\mathbf{U}_*^\kappa$ from (23) and, since approximate values $\mathbf{A}^{\kappa-1}$ of the coefficients are available, in place of (24), we use these values in (14) to set $\mathbf{U}_*^\kappa$ on the cage nodes (see below for a further note on this step). Before taking the divergence, we multiply (25) by the characteristic function χ of the domain external to the inner cages. Near the generic particle α we may take $\chi = H(S^\alpha)$, where H is the Heaviside distribution and $S^\alpha(\mathbf{x}) = 0$ is the inner cage surrounding particle α, with $S^\alpha > 0$ outside the cage. The result is

$$H\nabla^2 p^{\kappa+1} + \nabla H \cdot \nabla p^{\kappa+1} = \frac{\rho}{\Delta t}\left[H\nabla \cdot \mathbf{U}_*^\kappa + \left(\mathbf{U}^{n+1} - \mathbf{U}_*^\kappa\right) \cdot \nabla H\right]. \tag{27}$$

Thanks to the factor ∇H, the last term of the equation only contributes on the cage where, at convergence, $\mathbf{U}^{n+1} = \mathbf{U}_*$ as shown by (24). Thus, we drop this term and further approximate (27) by evaluating the second term at iteration level κ rather than $\kappa + 1$

$$H\nabla^2 p^{\kappa+1} = \frac{\rho}{\Delta t} H\nabla \cdot \mathbf{U}_*^\kappa - \nabla H \cdot \nabla p^\kappa. \tag{28}$$

The solution of this equation gives a new estimate $p^{\kappa+1}$ of p^{n+1}, which enables us to calculate $\mathbf{U}^{\kappa+1}$ from (25), from which new coefficients $\mathbf{A}^{\kappa+1}$ can be found by matching on the outer cage, and so on.

If (28) were solved as it stands, it would be necessary to implement a procedure that skips the nodes inside the inner cage, which would prevent the use of a fast solver. Thus, it proves convenient to add to (28) the divergence of the momentum equation written for a hypothetical fluid filling the inner cage, multiplied by $(1 - H)$. The result is

$$\nabla^2 p^{\kappa+1} = \frac{\rho}{\Delta t}\nabla \cdot \mathbf{U}_*^\kappa - \nabla H \cdot \nabla p^\kappa, \tag{29}$$

which, after suitable discretization, can be solved over the entire domain disregarding the presence of cages and bodies. The validity of this procedure may be better illustrated by considering the discretized version of this equation, which is found by integration over a pressure cell. Consider, for example, the (i, j) cell shown in Fig. 2. In this case, writing explicitly only the terms needed to explain this point, we have from (29):

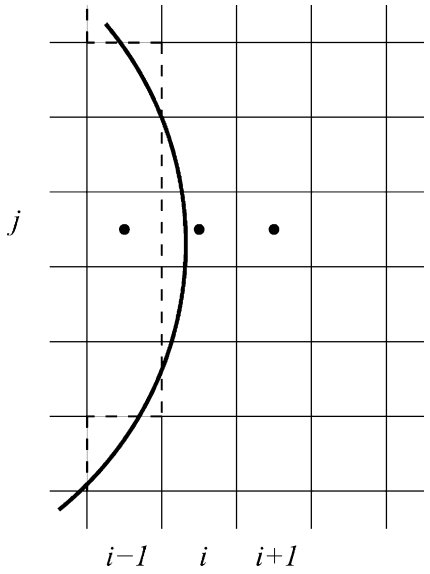

Fig. 2. Situation to which the numerical discretization (30) of the pressure equation (29) refers.

$$\frac{1}{\Delta x}\left(\frac{p_{i+1,j}^{\kappa+1}-p_{i,j}^{\kappa+1}}{\Delta x}-\frac{p_{i,j}^{\kappa+1}-p_{i-1,j}^{\kappa+1}}{\Delta x}\right)+\left(\frac{\delta^2 p^{\kappa+1}}{\delta y^2}\right)_{i,j}=\frac{\rho}{\Delta t}\left(\nabla\cdot\mathbf{U}_*^{\kappa}\right)_{i,j}-\frac{p_{i,j}^{\kappa}-p_{i-1,j}^{\kappa}}{\Delta x^2}. \tag{30}$$

As the iteration converges, the last term on the right-hand side will cancel the second term on the left-hand side, which is equivalent to solving the Poisson equation with a homogeneous Neumann boundary condition as in the standard procedure (see, e.g. [22]); this same procedure is applied at all the points marked by crosses in Fig. 1.

A steady-state flow may be considered as the limit of a time-dependent one as $t\to\infty$ but, if the transient is not of interest, the previous algorithm can be accelerated by avoiding to bring the κ-iteration to convergence. For example, for the simulation of flow past 1024 cylinders described below, we only took 3–5 iterations as opposed to the 10 or more that are typically necessary to accurately compute time-dependent problems. We found that the error decreases at each time step and becomes sufficiently small when the steady regime is attained.

5.2. Matching

According to step 1 of the procedure described in Section 4, once $\mathbf{u}^{\kappa+1}$ has been determined, it is necessary to update the coefficients $(A_k, B_k, \tilde{A}_k, \tilde{B}_k)$. For this purpose, we start by calculating, from the Cartesian components of $\mathbf{u}^{\kappa+1}$ on the outer cage, the radial and tangential components u_r and u_θ with respect to a temporary origin placed at the particle center. Since the Cartesian components are defined at cell edges, prior to this step we calculate them at cell centers by averaging. Then, the summation in (14) is truncated to a finite value N_c, velocity components are calculated, and equated to the values of $\mathbf{u}^{\kappa+1}$ at cell centers according to (20). This step involves solving a linear system and, in principle, one could retain as many coefficients as there are cells around the outer cage. In practice, however, on a finite-difference discretization with cells of side Δ, the shortest feature that can be resolved has a length scale $\sim 2\Delta$ so that, retaining modes with a shorter wavelength, would contaminate the calculation with aliasing errors rather

than increase accuracy. Truncating the summation in (14) at N_c is equivalent to allowing a shortest wavelength of the order of $2a/N_c$. Thus, we expect that a near-optimal choice would be $2a/N_c \simeq 2\Delta$ or $N_c \simeq \frac{1}{2}N_\Delta$, where $N_\Delta = 2a/\Delta$ is the number of mesh lengths per diameter. Since the number of available grid points is of the order of $2\pi a/\Delta \simeq 6N_\Delta$, and the number of coefficients $\simeq 4N_c$, in a typical calculation there are about three times as many grid points (each one with two pieces of data, u_r and u_θ) as coefficients. The system determining the coefficients has therefore a rectangular matrix, and we solve it by the Singular Value Decomposition algorithm, which is equivalent to a least-squares procedure when all singular values are retained (see, e.g. [23,24]).

It is apparent that the accuracy of the method can be increased arbitrarily by increasing the number of nodes (with the effect, among others, of reducing the extent of the region where applicability of the Stokes equations is assumed) and the number of coefficients in the exact solution. On the other hand, in some applications, it might be useful to use a relatively small number of nodes and coefficients while greatly increasing the number of particles. Thus, the method permits to shift the balance between accuracy and number of particles according to need. It should also be noted that, for a smooth velocity field, the analytic expansion (14) converges faster than algebraically and, therefore, the present method makes a very efficient use of the degrees of freedom retained for each particle.

Another significant advantage is that the force and torque on each particle are found directly once the low-order coefficients $A_0, B_1, \tilde{B}_1$ are known. This avoids the difficulty often encountered with other methods which require high-order extrapolations to obtain the stress distribution on the particles.

5.3. Cage

Fig. 1 shows an example of the inner and outer cages for a situation in which the cylinder center coincides with a cell center and there are eight mesh lengths per cylinder diameter. The matching of velocities on the outer cage is effected at cell centers. The arrows indicate the positions where the velocity boundary conditions are imposed on the inner cage. It will be noted that some of these points fall inside the particle, but this does not create difficulties – practical or conceptual – as, in principle, the function defined by (14) is defined for both $r > a$ and $r < a$ as long as $r \neq 0$.

In a more general situation in which the cylinder center does not coincide with a cell center, the cages are constructed as follows. A circle centered at the cylinder center, with radius $a + \Delta$ (where Δ is the mesh spacing), is drawn, and the pressure points (cell centers) falling between this circle and the cylinder surface are used to build the inner cage. With this construction, the pressure points are always outside the particle. The outer cage is constructed by taking the pressure points of the inner cage and moving outward by one cell; this process results in an outer cage that is, at most, 2 mesh sizes away from the cylinder surface.

6. Some examples

We now present some results obtained by the method described before. We first illustrate its accuracy by comparison with the results for flow past a single cylinder of [25] (see also [26]). Then we consider the flow past many cylinders up to 1024.

6.1. Flow past a single cylinder

Table 1 shows, for different Reynolds numbers $Re = 2aU/\nu$, a comparison of the drag coefficient $C_D = F_D/\rho U^2 a$ with different discretizations (expressed in terms of $N_\Delta = 2a/\Delta$, the ratio of cylinder diameter to mesh size) and different truncations of the analytical solution (14), expressed in terms of N_c, the order of the highest Fourier mode retained (recall that the total number of coefficients is $4N_c + 1$).

Table 1
Some numerical results for a single cylinder with the two-cage implementation and comparison with [25]

Re	N_Δ	N_c	C_D		Wake length	
			Present	Ref. [25]	Present	Ref. [25]
5	8	4	4.159	4.116		
		6	4.123			
	16	4	4.310			
		8	4.270			
10	8	4	2.971	2.846	0.375	0.53
		6	2.981			
	16	4	2.961		0.49	
		8	2.920			
20	8	4	2.095	2.045	1.87	1.88
	16	4	2.160		1.88	
		6	2.120			
		8	2.105		1.88	1.88
40	8	4	1.457	1.522	4.59	4.69
		5	1.689			
		6	1.687			
	16	4	1.607		4.60	
		8	1.587		4.60	

In these calculations the domain was a square with a side equal to 40 cylinder diameters; a constant velocity was imposed at the inlet and on the lateral boundaries, and free outflow conditions at the outlet. For the simulation with $Re = 10$ or larger the cylinder was centered 10 diameters downstream from the inlet. If this same location was used for $Re = 5$, the calculated C_D was in error by nearly 5%; by placing the cylinder at the center of the domain, instead, we find the results shown in the table which, in the best case presented ($N_\Delta = 8$), differs by less than 0.2% from those of [25].

In Table 1 we also compare the calculated wake length (measured in units of the cylinder radius) for $Re = 20$ and 40; the agreement is excellent, which suggests that the flow field is as well captured as the force on the cylinder. For $Re = 13$ and 26 (with $N_\Delta = 8$, $N_c = 4$) our computed wake lengths are 0.875 and 2.625. The corresponding experimental values, as estimated from the photographs in [27], are 0.88 and 2.4, respectively.

6.2. Randomly distributed cylinders in a periodic system

We now consider simulations of the steady flow through a bed of many cylinders randomly distributed in a square cell at the edges of which periodic boundary conditions are applied. This is a standard device to simulate the flow through a large cylinder bed which, physically, might represent the flow through a fibrous material (see, e.g. [28–31]). The flow is driven by an imposed pressure gradient $\mathbf{P}$ by writing, in the Navier–Stokes equation, $\nabla p = \mathbf{P} + \nabla \hat{p}$, where $\hat{p}$ is subjected to periodicity boundary conditions. For most of the simulations we use a discretization with eight meshes per cylinder diameter ($N_\Delta = 8$) and $N_c = 4$.

A picture of the flow through a random distribution of 25 cylinders occupying a volume (area) fraction of 20% is shown in Fig. 3; the Reynolds number defined in terms of the average velocity through the bed and cylinder diameter is about 20. The phenomenon of channeling along particular flow paths is clearly visible here. By taking ensemble averages of results of this type one can calculate the average flow conductance through a bank of cylinders. As an example we show in Fig. 4 the running average of the Reynolds number based on the average velocity over a small ensemble of 10 realizations for 25 cylinders and 3

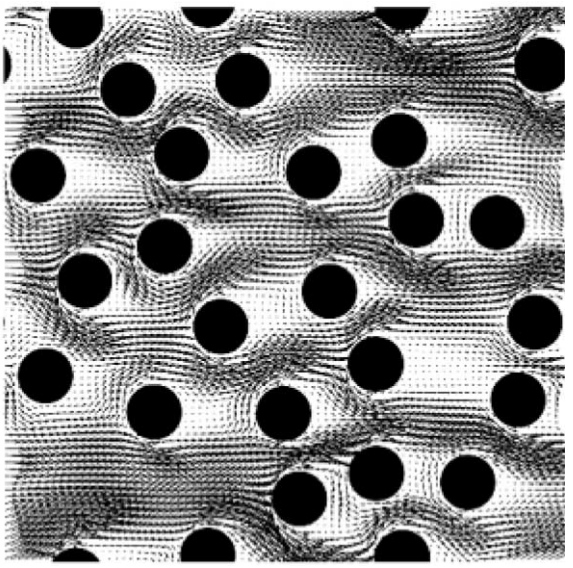

Fig. 3. Periodic flow over 25 randomly oriented cylinders occupying an area fraction of 20%; the Reynolds number based on the average velocity is about 20 (two-cage implementation).

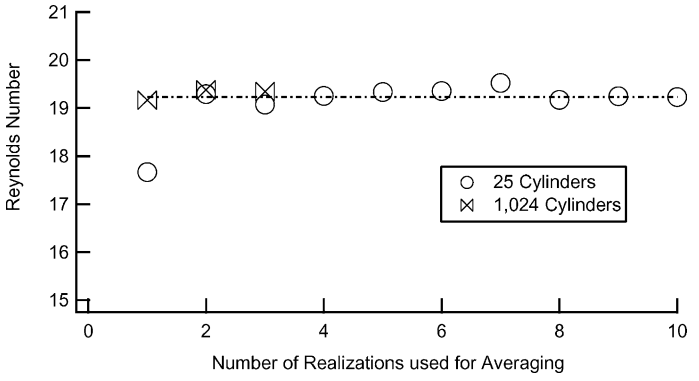

Fig. 4. Running average of the pressure-driven mean velocity through a bed of cylinders with a volume (area) fraction of 20%. Circles: 25 cylinders in the unit cell; crosses: 1024 cylinders; in the latter case, only three configurations were used. The imposed dimensionless pressure gradient is $a^3\mathbf{P}/(\rho v^2) = 50$.

for 1024 cylinders, again for a volume fraction of 20%, and for a dimensionless pressure gradient $a^3\mathbf{P}/(\rho v^2) = 50$. As expected, the result with a large number of cylinders settles down to the ensemble average very quickly with a much smaller standard deviation.

Next, we picked one particular realization of the 25-cylinder ensemble (the same one shown in Fig. 3) and studied the relation between the average force on the cylinders and the flow rate which, according to the well-known Ergun formula, should have the form

$$\frac{F_D}{\mu U} = c_1 + c_2 Re, \tag{31}$$

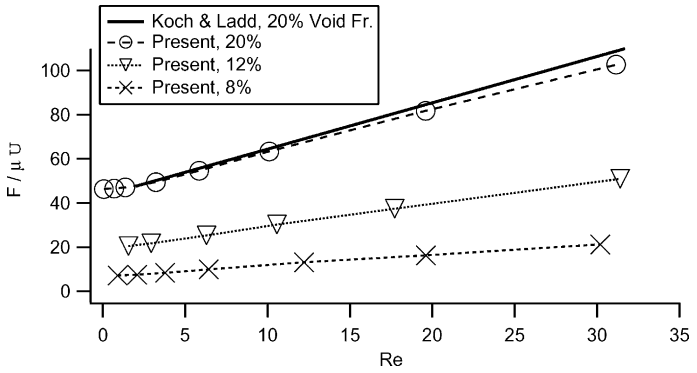

Fig. 5. Normalized mean force per unit cylinder length vs. Reynolds number (based on the mean, rather than the superficial velocity) for the pressure-driven flow through a cylinder bed. The lines correspond to different volume fractions; the thick line is the result of [32].

in which c_1, c_2 are constants dependent on the volume fraction. Ref. [32] shows by a lattice–Boltzmann calculation that this relation holds for $Re \geqslant 5$ while, for smaller Reynolds numbers, a quadratic dependence of the second term on Re is a better approximation. The results of the present simulations, shown in Fig. 5 for volume fractions of 20%, 12%, and 8% (the latter two obtained by randomly removing 10 and then 5 cylinders from the domain), support these conclusions; the solid line is the result of [32]. (Note that here, and only here, following [32], the Reynolds number is expressed in terms of the mean, rather than the superficial velocity; the former equals the volume fraction times the latter.)

A useful check of the numerical accuracy of the method can be done here because, in view of the use of a periodic domain, the sum of all the hydrodynamic forces acting on the cylinders must equal the total hydrodynamic force on the boundary of the computational domain: in other words, the force exerted by the pressure gradient must equal the total force on the cylinders. A comparison between these two quantities is shown in Table 2, the last column of which also gives the mean dimensionless force per unit length.

It can be seen in the table that, for very small pressure gradients, there is an excellent agreement, which however deteriorates as Re increases reaching about 3% for $Re \simeq 24$. According to the estimate given at the end of Section 4, one may expect that a discretization with a certain N_Δ would be adequate up to $Re \simeq \frac{1}{4}N_\Delta^2$ which, in the present case, would be $64/4 = 16$. Thus, the deviation encountered in the table at the larger

Table 2
Comparison between the imposed pressure gradient and the total drag force for flow past a bed of 25 cylinders with a 20% volume (area) fraction; L is the cell size (two-cage implementation)

Re	$a^3\mathbf{P}/(\rho\nu^2)$	$(a^3/\rho\nu^2L^2)\sum F_D$	% Difference	$\overline{F}_D/\mu\overline{U}$
0.086	0.1	0.1	0.00	37.29
0.85	1	1.0	0.00	37.42
1.69	2	2.001	0.00	37.85
4.02	5	5.017	0.34	39.81
7.28	10	10.075	0.75	43.9
12.57	20	20.13	0.65	50.778
24.39	50	48.6	2.80	65.71
38.76	100	90.1	9.90	82.53
59.54	200	159.2	20.40	107.53

Reynolds numbers is not surprising. For example, if the calculation with $a^3\mathbf{P}/(\rho v^2) = 100$ is repeated with $N_\Delta = 16$ and $N_c = 8$, one finds an error of 1.2% in place of nearly 10% as in the table.

In Section 5.2 we presented an estimate of the optimal number of coefficients that should be retained for a given discretization of the domain. Table 3 illustrates the effect of changing the number of coefficients for a discretization with 16 meshes per diameter and $Re \simeq 32$. The results support the estimate $N_c \sim (1/2)N_\Delta$ given before.

As a last example, we show in Fig. 6 transient results for the flow past the 25 cylinders of Fig. 3. Here the flow starts from rest and, at time 0, a pressure gradient $a^3\mathbf{P}/(\rho v^2) = 50$ is imposed.

6.3. CPU time

An interesting feature of the present scheme is its computational efficiency. For the 25 cylinder simulations described before, typical computation times to steady state were about 10 min on a 600 MHz Pentium III PC. For the 1024 cylinder simulation at the same volume fraction of 20% (and, therefore, on a much bigger domain), on the same PC, the time required was typically of the order of 6 h.

Fig. 7 illustrates the increase in computation time as the number of cylinders is increased. Here, we have started with a domain large enough to contain 1024 cylinders with a volume (area) fraction of 20%, and we have then gradually removed cylinders at random maintaining the same computational domain discretized with 512×512 nodes with 8 mesh lengths per cylinder diameter. Since the complexity of the flow structure (which affects the convergence rate) increases with the Reynolds number, in order to have similar conditions

Table 3
Effect of increasing the number of coefficients for the flow past 25 cylinders shown in Fig. 3 with $a^3\mathbf{P}/(\rho v^2) = 100$ ($Re \simeq 38.76$)

N_c	% Error
2	11.4
4	2.6
6	1.0
8	1.2
10	5.3
12	9.6
13	13.2
14	No convergence

The optimal value estimated in Section 5.2 is $N_c \sim \frac{1}{2}N_\Delta$ which equals 8 in this case.

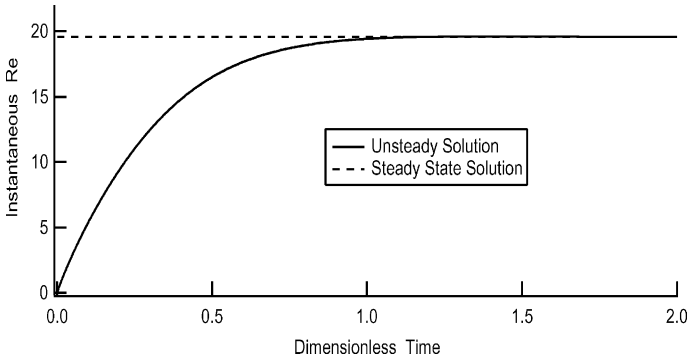

Fig. 6. Mean velocity vs. time for the flow past the 25 cylinders of Fig. 3 for an impulsively applied pressure gradient $a^3\mathbf{P}/(\rho v^2) = 50$.

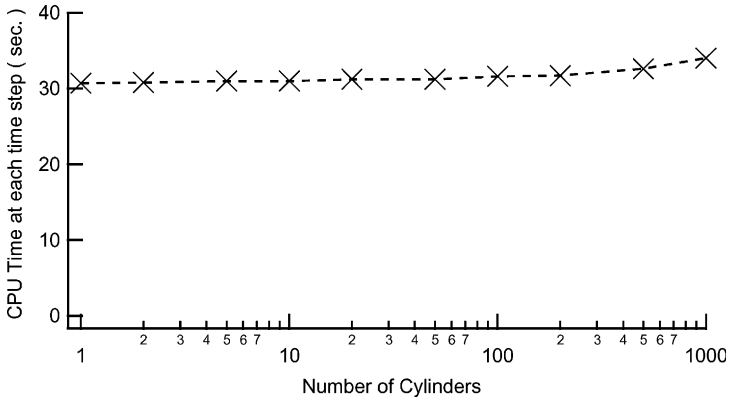

Fig. 7. CPU time on a 600 MHz Pentium III PC per time step vs. number of cylinders keeping the total number of iterations constant.

in each case, the imposed pressure gradient was adjusted so that the mean force $(a^3\overline{F}_D/\rho v^2 L^2)$ per cylinder was kept constant and equal to the same value calculated for 1024 cylinders with $Re = 5.6$. This particular way of performing the comparison had the consequence that the Reynolds number varied with the number of cylinders. Furthermore, in order to have a more meaningful comparison, for all cases we executed 3 pressure iterations and 5 coefficient iterations. It may be noted that the number of iterations necessary for convergence to steady state increases as the number of cylinders decreases because the flow becomes less constrained. Thus, it may be said that the result presented in this figure is perhaps biased *in favor of the smaller computations*.

The striking feature of this figure is that, as the number of cylinders is increased, the computational time increases only very slightly because most of the CPU time is spent solving the pressure Poisson equation. Adding particles increases the number of linear systems that must be solved to calculate the coefficients of the analytical solution, but since these are small their computational cost is limited. Thus, although in principle the computational time asymptotically must grow linearly with the number of linear systems determining the coefficients – and hence with the number of particles – in practice a large number of particles requires a large computational domain, and the cost of the finite-difference flow solver on such a domain is superior to that of solving for the coefficients. Hence we may say that, in practice, *for a given computational domain*, the total computational cost is only weakly dependent on the number of particles.

7. Single-cage implementation

An attractive variant of the general idea described before is based on the use of a single cage, with the coefficients computed from the pressure and vorticity distributions, rather than velocity on the outer cage as in Section 4. As shown in [5], this formulation has several advantages. In the first place, it requires validity of the Stokes equations only between the particle surface and the single cage, rather than all the way to the outer cage: this benefits accuracy and permits the use of a somewhat coarser discretization. Secondly, since $\tilde{\mathbf{u}}$ vanishes on the particle surface, there is a 'signal-to-noise' problem if this quantity is used, as in the previous method, to calculate the coefficients: vorticity and pressure, on the other hand, are finite near the particle. Thirdly, the cage can be placed *inside* the particle, with the advantage that particle contact and collision can be accounted for with greater fidelity; an example is given in [5]. As before, we use the SVD algorithm to solve for the coefficients. We have checked that the typical residual error with which the system is satisfied is a fraction of 1%.

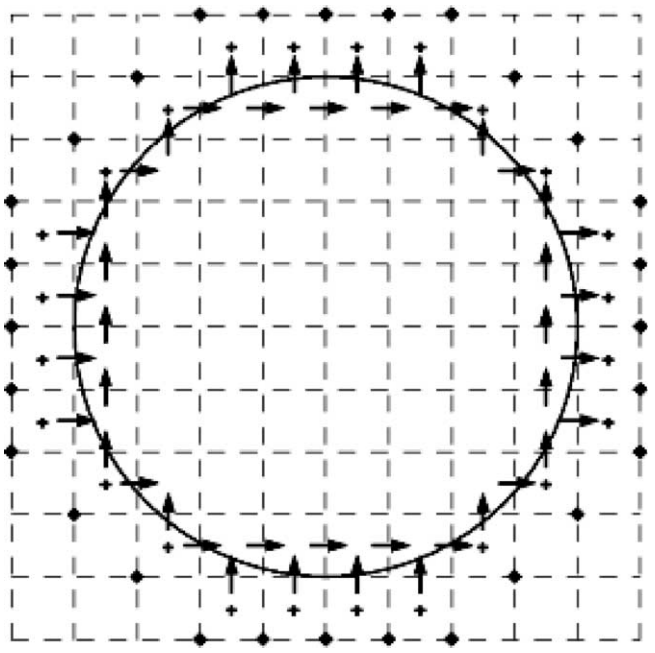

Fig. 8. One of the cages used in the one-cage formulation; crosses: pressure nodes; diamonds: vorticity nodes; arrows: velocity nodes.

Figs. 8 and 9 show typical examples of the cages that we use. Here the crosses are the pressure nodes, the diamonds the vorticity nodes, and the arrows the velocity nodes; the vorticity at the node (i, j) is calculated from

$$\omega_{ij} = \left[\frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}\right]_{i,j} \simeq \frac{v_{i+1,j} - v_{i,j}}{\Delta x} - \frac{u_{i,j+1} - u_{i,j}}{\Delta y}. \tag{32}$$

In the first figure the pressure nodes are all outside the particle, whereas in the second one they are just inside. Convergence is more robust when the first cage is used, but the second one is preferable when there are many particles as it allows them to come closer to one another. At present the reasons for the different numerical behavior of the two cages are not fully understood.

Tables 4 and 5 show some results to be compared with those presented earlier in Tables 1 and 2, respectively; here the cage is that of Fig. 8 and, for the case of Table 5, 10 mesh spacings per cylinder diameter were used. The results are essentially the same as before.

We next compare with some results for the flow induced by a line of cylinders held between two parallel walls moving with velocity $2aU/v = 1$ as obtained by the finite-element calculation of [33] and the lattice–Boltzmann calculation of [34]. The cylinders are spaced by a distance L, D denotes the spacing between the walls, and $f_c = |\mathbf{F}|/2\pi\rho U^2 a$, with $\mathbf{F}$ the total fluid-dynamic force. Table 6 illustrates the near-perfect agreement among the three computations. Here the column labelled $a/\Delta x$ refers to the computations of [34,33]; the third column denotes the conditions for the present computation. Note that in the last two cases our results have been obtained with 1/2 and 1/3 of the nodes of the lattice–Boltzmann calculation, respectively. The last line in Table 6 is for the reciprocal case in which the cylinders move between stationary walls. The results given by the present method in the two cases are virtually indistinguishable. The numbers shown have been obtained with the cage of Fig. 9; use of the cage of Fig. 8 gave differences of 1% or less.

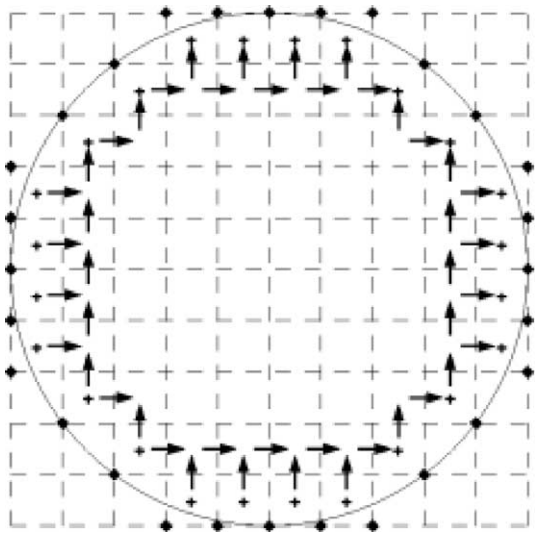

Fig. 9. Another cage for the one-cage formulation; crosses: pressure nodes; diamonds: vorticity nodes; arrows: velocity nodes.

Table 4
Some numerical results for a single cylinder with the single-cage implementation and comparison with [25]

Re	N_Δ	N_c	C_D		Wake length	
			Present	Ref. [25]	Present	Ref. [25]
5	16	4	4.172	4.116		
5	16	8	4.210			
20	16	6	2.139	2.045	1.88	1.88
20	16	8	2.140		1.88	
40	8	4	1.613	1.522	4.38	4.69
40	16	4	1.595		4.58	
40	16	8	1.597		4.57	

Table 5
Comparison between the imposed pressure gradient and the total drag force for flow past a bed of 25 cylinders with a 20% volume (area) fraction; L is the cell size (single-cage implementation with 10 mesh spacings per cylinder diameter)

Re	$a^3\mathbf{P}/(\rho\nu^2)$	$(a^3/\rho\nu^2L^2)\sum F_D$	% Difference	$\overline{F}_D/\mu\overline{U}$
0.85	1.0	1.00	0.07	37.71
1.68	2.0	2.00	0.07	38.20
3.96	5.0	5.00	0.07	40.39
7.21	10.0	10.01	0.06	44.40
12.48	20.0	20.00	0.02	51.29
23.52	50.0	49.80	0.40	67.74

As a final example, we study the uniform flow past a cylinder at a Reynolds number of 60. We performed two different calculations with computational domains $20a \times 60a$ and $40a \times 100a$ with periodic boundary conditions at the upper and lower boundaries, a prescribed velocity at the inlet, and free outflow conditions

Table 6
Comparison among different calculations of the force on a line of cylinders between two walls moving with a relative velocity $2aU/v = 1$

L/D	$a/\Delta x$	Nodes per cylinder	f_c		
			Present	Ref. [34]	Ref. [33]
11.8	10.8	10	1.035	1.053	0.966
6.1	20.8	10	1.235	1.251	1.158
2.09	60.8	20	2.081	2.093	2.067
2.09	60.8	20	2.083	(fixed walls)	

The present results have been obtained with the single-cage implementation of Section 7, the first three assuming fixed cylinders and moving walls, the last one for moving cylinders between fixed walls as in [33,34].

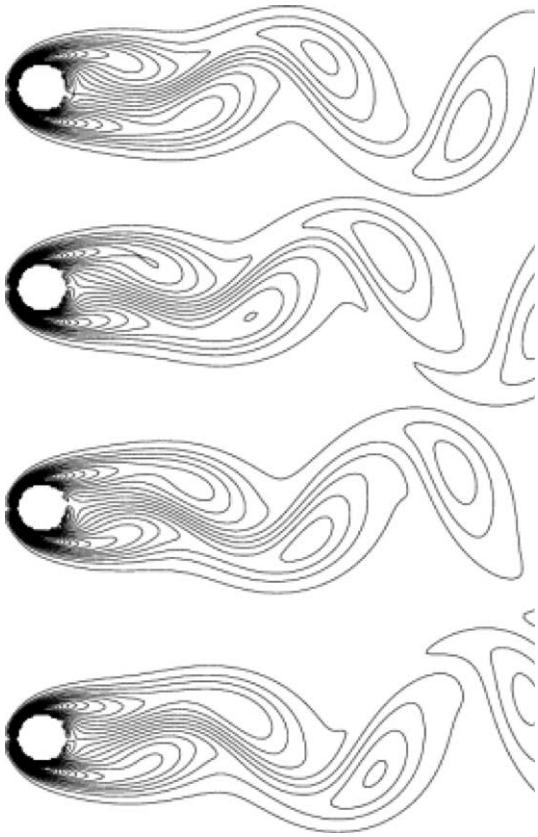

Fig. 10. Instantaneous vorticity contours for the flow past a cylinder at $Re = 60$.

at the right boundary. We used 16 nodes per cylinder, and a dimensionless time step $v\Delta t/a^2 = 0.001$. For both calculations, steady conditions were reached at about $vt/a^2 = 12$. Fig. 10 shows the instantaneous vorticity contours for the first calculation. We observe the expected self-excited wake oscillations and periodic vortex shedding. Fig. 11 shows the dimensionless lift force on the cylinder, which oscillates very regularly with time. The Strouhal number is 0.153, while the calculation with the bigger domain gives 0.148. These results are in good agreement with the experimental value of 0.135 [26].

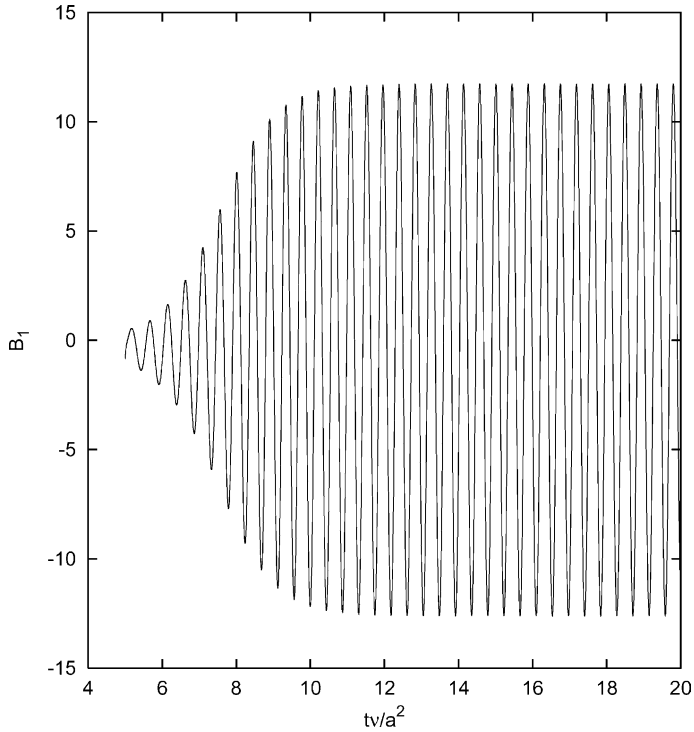

Fig. 11. Dimensionless lift force $F_l/4\pi\nu\mu$ (i.e., the coefficient B_1 in Eq. (19)) for the flow past a cylinder at $Re = 60$ shown in the previous figure.

8. Conclusions

We have presented a new numerical method for the direct simulation of particle flows. We have focused here on the two-dimensional case, but it is clear that a similar approach carries over to three dimensions; the necessary computer code is currently being tested. The method has been shown to be accurate and numerically efficient. In particular, as can be seen in Fig. 7, the computational time exhibits a very favorable scaling with the number of particles, which makes the method suitable for simulations with many particles. For example, we were able to solve the flow past 1024 cylinders in a few hours on a simple PC. Rather than time, the main limitation stems from memory requirements, but these are not significantly different from those that would limit a single-phase calculation at the same Reynolds number.

Another advantage of the approach is the accuracy with which the force and torque on the particles can be computed, thanks to the spectral representation of the solution near the particles. The fast convergence of this representations has also the effect that the degrees of freedom assigned to each particle – i.e., the expansion coefficients – represent the effect of the local flow inhomogeneity in a very efficient (perhaps even optimal) way. In particular, if desired, this circumstance permits relatively inexpensive, although still reasonably accurate, simulations with many particles. Of course, like any other consistent numerical method, by a sufficient refinement of the finite-difference grid and concomitant increase of the number of coefficients retained per particle, an arbitrary accuracy can also be achieved.

The method has been demonstrated in a finite-difference context, but its extension to finite volume or finite element flow solvers does not seem to present particular problems. It is also obvious that the procedure can be parallelized in a straightforward manner assigning one particle or group of particles to each

processor, and using the same strategies as for single-phase computations to handle the rest of the computation.

As apparent from the material presented in this paper, a limitation of our approach is the need for an exact analytical solution of the Stokes equations, which only exists for a few body shapes. A way to overcome this shortcoming would be to use a boundary-integral solution for the local Stokes problem; other possibilities are the use of a body-fitted, or finite-element, mesh near the body which would match a regular grid on the rest of the domain.

Acknowledgements

This study has been supported by DOE under Grant DE-FG02-99ER14966. S.T. acknowledges support by the Japan Ministry of Education.

References

[1] A. Prosperetti, H.N. Oğuz, PHYSALIS: a new o(N) method for the numerical simulation of disperse flows of spheres. Part I: potential flow, J. Comput. Phys. 167 (2001) 196–216.
[2] S. Takagi, H.N. Oğuz, A. Prosperetti, PHYSALIS: a new method for particle simulation, in: E.E. Michaelides (Ed.), Proceedings of the 4th International Conference on multiphase flow, 2001.
[3] E. Ory, H.N. Oğuz, A. Prosperetti, *PHYSALIS*: a new O(N) method for the numerical simulation of particle flows, in: Proceedings of the ASME FED Summer Meeting, American Society of Mechanical Engineers, New York, 2000.
[4] H. Huang, S. Takagi, PHYSALIS: a new method for particle flow simulation. Part III: Convergence analysis, J. Comput. Phys. (submitted).
[5] Z. Zhang, A. Prosperetti, A method for particle simulation, J. Appl. Mech., 2003 (in press).
[6] D.E. Stock, Particle dispersion in flowing gases, J. Fluids Engrg. 118 (1996) 4–17.
[7] C.T. Crowe, T.R. Troutt, J.N. Chung, Numerical models for two-phase turbulent flows, Ann. Rev. Fluid Mech. 28 (1996) 11–41.
[8] A. Ahmed, S. Elghobashi, Direct numerical simulation of particle dispersion in homogeneous turbulent shear flows, Phys. Fluids 12 (2000) 2906–2930.
[9] J. Feng, H.H. Hu, D.D. Joseph, Direct simulation of initial value problems for the motion of solid bodies in a Newtonian fluid, Part1. Sedimentation, J. Fluid Mech. 261 (1994) 95–134.
[10] A.S. Sangani, G.B. Mo, An O(N) for Stokes and Laplace interactions of spheres, Phys. Fluids 8 (1996) 1990–2010.
[11] Y. Pan, S. Banerjee, Numerical investigation of the effect of large particles on wall turbulence, Phys. Fluids 9 (1997) 3786–3807.
[12] A.A. Johnson, T. Tezduyar, 3-D simulation of fluid–particle interactions with the number of particles reaching 100, Comput. Methods Appl. Mech. Engrg. 145 (1997) 301–321.
[13] R. Glowinski, T.W. Pan, T.I. Hesla, D.D. Joseph, A distributed Lagrange multiplier/fictitious domain method for particulate flows, Int. J. Multiphase Flow 25 (1999) 755–794.
[14] S. Takiguchi, T. Kajishima, Y. Miyake, Numerical scheme to resolve the interaction between solid particles and fluid turbulence, JSME Int. J. B 42 (1999) 411–418.
[15] N.A. Patankar, P. Singh, D.D. Joseph, R. Glowinski, T.-W. Pan, A new formulation of the distributed Lagrange multiplier/fictitious domain method for particulate flows, Int. J. Multiphase Flow 26 (2000) 1509–1524.
[16] M.R. Maxey, B.K. Patel, Localized force representations for particles sedimenting in stokes flow, Int. J. Multiphase Flow 27 (2001) 1603–1626.
[17] P. Bagchi, S. Balachandar, Steady planar straining flow past a rigid sphere at moderate Reynolds number, J. Fluid Mech. 466 (2002) 365–407.
[18] S. Chen, G.D. Doolen, Lattice Boltzmann method in fluid flows, Annu. Rev. Fluid Mech. 30 (1998) 329–364.
[19] E.-J. Ding, C.K. Aidun, The dynamics and scaling law for particles suspended in shear flow with inertia, J. Fluid Mech. 42 (2000) 317–344.
[20] H. Nirschl, H.A. Dwyer, V. Denk, Three-dimensional calculations of the simple shear flow around a single particle between two moving walls, J. Fluid Mech. 283 (1995) 273–285.
[21] J.J. Chattot, Y. Wang, Improved treatment of intersecting bodies with the Chimera method and validation with a simple and fast flow solver, Comput. Fluids 27 (1998) 721–740.
[22] C.A.J. Fletcher, Computational Techniques for Fluid Dynamics, Springer, Berlin, 1988.

[23] R. Kress, Linear Integral Equations, Springer, Berlin, 1989.
[24] W.H. Press, W.T. Vetterling, S.A. Teukolsky, B.P. Flannery, Numerical Recipes in FORTRAN, second ed., Cambridge University Press, Cambridge, 1992.
[25] S.C.R. Dennis, G.Z. Chang, Numerical solutions for steady flow past a circular cylinder at Reynolds numbers up to 100, J. Fluid Mech. 42 (1970) 471–489.
[26] M.M. Zdravkovich, Flow Around Circular Cylinders. Vol. 1: Fundamentals, Oxford University Press, Oxford UK, 1997.
[27] M. Van Dyke, An Album of Fluid Motion, Parabolic Press, Stanford, CA, 1982.
[28] A.J.C. Ladd, Sedimentation of homogeneous suspensions of non-Brownian spheres, Phys. Fluids 9 (1997) 491–499.
[29] M.P. Allen, D.J. Tildesley, Computer Simulation of Liquids, Clarendon Press, Oxford, 1987.
[30] G. Mo, A.S. Sangani, A method for computing Stokes flow interactions among spherical objects and its application to suspensions of drops and porous particles, Phys. Fluids 6 (1994) 1637–1652.
[31] A.S. Sangani, C. Yao, Transport processes in random arrays of cylinders. II. Viscous flow, Phys. Fluids 31 (1988) 2435–2444.
[32] D.L. Koch, A.J.C. Ladd, Moderate Reynolds number flow through periodic and random arrays of aligned cylinders, J. Fluid Mech. 349 (1997) 31–66.
[33] C.K. Aidun, Y.Lu. Lattice, Boltzmann simulation of solid particles suspended in a fluid, J. Stat. Phys. 81 (1995) 49–61.
[34] T. Inamuro, K. Maeba, F. Ogino, Flow between parallel walls containing the lines of neutrally buoyant circular cylinders, Int. J. Multiphase Flow 26 (2000) 1981–2004.