PHYSICAL REVIEW E **80**, 025701(R) (2009)

Accelerated algorithm for computing the motion of solid particles suspended in fluid

E. J. Ding*
1530 Belmont Hills Drive, Suwanee, Georgia 30024, USA

(Received 19 January 2009; revised manuscript received 14 April 2009; published 19 August 2009)

A fast algorithm for computing the motion of solid particles suspended in fluid is presented. The motion of solid particles suspended in Stokes flow can be calculated without fully calculating the fluid motion. When the steady-state simulation is sufficient, this algorithm can greatly accelerate the simulation of solid particle suspension in Stokes flow.

DOI: 10.1103/PhysRevE.80.025701 PACS number(s): 47.11.−j, 83.80.Hj, 47.10.−g

In computational fluid dynamics (CFD) the motion of the particles suspended in fluid is simulated by Navier-Stokes (N-S) equation coupled with Newtonian equation of motion. In each time step of the simulation, the state of the fluid and the configuration of the solid particles are alternately updated. In the Stokes flow, as Reynolds number $\mathrm{Re}\rightarrow 0$, [1,2] both the fluid inertia and the solid particle inertia are negligible, and the governing equations become $-\nabla p+\mu\nabla^2\mathbf{u}=0$ (the Stokes equation) and $\nabla\cdot\mathbf{u}=0$ (the continuous equation) for the fluid and $\mathbf{F}=0$ (the force-free condition) for the solid particles [1,2], where μ is the viscosity of the fluid, p and $\mathbf{u}$ are the local pressure and local velocity of the fluid, respectively, and $\mathbf{F}$ is the total force on the solid particle. Finite element method (FEM) is widely used for a steady-state simulation because a relatively large time step size can be selected to accelerate the simulation even though the remeshing process may cost a large portion in the computational time. In the recent two decades the lattice-Boltzmann method (LBM) has become a choice of CFD method. Combining lattice-Boltzmann equation (LBE) and Newtonian equation of motion, the motion of the fluid and the solid particles suspended in fluid can be simulated effectively [3–10]. However, LBE solves time-dependent flow, and the fluid state and the motion of the solid particles are updated using very small time steps. For situations where steady-state simulations are sufficient, LBM becomes too expensive in computational time. Attempts to obtain the steady state of fluid in LBM so far are in two ways. One way is the time-dependent LBE approach, where the time-dependent LBE is continuously solved with solid particles unmoved in the fluid, until a steady state is obtained [11], while the second way is the time-independent-matrix approach, in which a matrix equation, converted from the steady-state LBE, is solved using standard matrix methods [12].

In this Rapid communication a fast algorithm for solving steady state of LBE in Stokes flow is presented. A background flow is calculated and saved as a database before the simulation. Then the velocity of solid particle in each time step can be very efficiently calculated by using the background flow. The scheme is applicable to a suspension of solid particles, as well as for just one particle. For simplicity only D2Q9 model [4] in a $L\times H$ rectangular domain will be discussed. Periodic boundary condition is applied to x direction, and a simple shear flow is generated by two parallel walls moving in opposite directions on the top and bottom, respectively. Generalization to other models in two-dimensional (2D) or three-dimensional (3D) cases is straightforward.

In LBM the state of the fluid is characterized by the distribution function, $\phi_k(\mathbf{x},t)$ at time t, at node $\mathbf{x}$, with velocity $\mathbf{e}_k$ defined as (0,0), (1,0), (1,1), (0,1), (−1,1), (−1,0), (−1,−1), (0,−1), and (1,−1) for $k=0-8$, respectively. The equilibrium distribution function for zero Reynolds number flow is [5]

$$\phi_k^{eq}=w_k\rho[1+3\mathbf{e}_k\cdot\mathbf{u}]. \tag{1}$$

The coefficient w_k is 4/9 for $k=0$, 1/9 for $k=1$, 3, 5, 7, and 1/36 for $k=2$, 4, 6, 8, respectively. The density $\rho(\mathbf{x},t)$ and velocity $\mathbf{u}(\mathbf{x},t)$ at a fluid node is represented as $\rho(\mathbf{x},t)=\Sigma_{k=0}^{8}\phi_k(\mathbf{x},t)$ and $\rho(\mathbf{x},t)\mathbf{u}(\mathbf{x},t)=\Sigma_{k=0}^{8}\mathbf{e}_k\phi_k(\mathbf{x},t)$, respectively. The local density can be expressed as $\rho(\mathbf{x},t)=\rho_0+\delta\rho(\mathbf{x},t)$, with the global density $\rho_0=(1/N_f)\Sigma_{\mathbf{x}}\rho(\mathbf{x},t)$ a constant, where $N_f=LH$ is the total number of fluid nodes, and the summation is taken over every node. Denoting $\phi_k=w_k\rho_0+\delta\phi_k$, we have

$$\delta\rho(\mathbf{x},t)=\sum_{k=0}^{8}\delta\phi_k(\mathbf{x},t),$$
$$\rho\mathbf{u}(\mathbf{x},t)=\sum_{k=0}^{8}\mathbf{e}_k\delta\phi_k(\mathbf{x},t). \tag{2}$$

Adjacent fluid nodes are connected by links, along the directions of the velocities $\mathbf{e}_k$. The fluid particles may move only along these links. The evolution of the distribution function is determined by LBE

$$\phi_k^*(\mathbf{x},t)=\phi_k(\mathbf{x},t)-(1/\tau)[\phi_k(\mathbf{x},t)-\phi_k^{eq}(\mathbf{x},t)],$$
$$\phi_k(\mathbf{x}+\mathbf{e}_k,t+1)=\phi_k^*(\mathbf{x},t), \tag{3}$$

where τ is the relaxation time scale, which is related to fluid kinematic viscosity $\nu=(2\tau-1)/6$. Since ν is not an important parameter for zero-Reynolds number flow (see [12] and references therein), we may take $\tau=1$. The advantage of this choice is that the after-collision status is simply the equilibrium state, and Eqs. (3) and (1) are reduced to

*eding.simufast@gmail.com

$$\delta\phi_k^*(\mathbf{x},t) = \delta\phi_k^{eq}(\mathbf{x},t),$$
$$\delta\phi_k(\mathbf{x}+\mathbf{e}_k,t+1) = \delta\phi_k^*(\mathbf{x},t), \tag{4}$$

and

$$\delta\phi_k^{eq}(\mathbf{x},t) = w_k[\delta\rho + 3\rho\mathbf{e}_k\cdot\mathbf{u}], \tag{5}$$

respectively. At a node adjacent to a wall,

$$\delta\phi_k(\mathbf{x}+\mathbf{e}_k,t+1) = \delta\phi_{k'}^*(\mathbf{x},t) - 6w_k\rho_0\mathbf{u}_b\cdot\mathbf{e}_{k'}, \tag{6}$$

where $\mathbf{u}_b$ is the velocity of the wall [5]. In this Rapid Communication k' always denotes the opposite direction of k.

The pure fluid flow without disturbance from moving solid particles is called *the background flow*. In D2Q9 model, the state of fluid can be denoted by a column vector Φ (and Φ^*) with $N_c=9N_f$ elements, standing for component $\delta\phi_k$ (and $\delta\phi_k^*$) at every node in the whole domain. If the velocities of the walls are u_{top} and u_{bottom}, respectively, combining Eqs. (4) and (6), the column vectors Φ at time t and $t+1$ are related as

$$\Phi(t+1) = T\Phi^*(t) + WU_w, \tag{7}$$

where T is a $N_c\times N_c$ translation matrix, W is a $N_c\times 2$ matrix, and $U_w=(u_{top},u_{bottom})^T$. The state of fluid can also be expressed by a column vector M in *the fluid phase space*, which is a $N_m=3N_f$ dimensional space spanned by the local density variation and momentum $(\delta\rho,\rho u_x,\rho u_y)$ at each node. The vectors Φ (and Φ^*) and M are related by $M=E^+\Phi$ and $\Phi^*=EM$, where E and E^+ are $N_c\times N_m$ and $N_m\times N_c$ block-diagonal matrices, with each block defined by Eqs. (2) and (5), respectively [12]. Equation (7) can be written as

$$M(t+1) = E^+TEM(t) + E^+WU_w. \tag{8}$$

The steady state $M=M(t\to\infty)$ satisfies

$$(I - E^+TE)M = E^+WU_w. \tag{9}$$

Equation (9) cannot be solved because the coefficient matrix $I-E^+TE$ is singular. The singularity comes from the undetermined value of the total fluid mass in the system [12]. The mass conservation condition $\Sigma\delta\rho=0$ (the summation over N_f nodes) can be expressed by $JM=0$, where J is a $N_m\times N_m$ matrix with only N_f nonzero elements which are equal to one. These nonzero elements are on the first row, at the columns associated with $\delta\rho$ at each node. The steady state can now be determined by

$$AM = E^+WU_w, \tag{10}$$

where A is a $N_m\times N_m$ matrix

$$A = I - E^+TE + J. \tag{11}$$

The solution of Eq. (10) is $M_0=M_0^RU_w$, where

$$M_0^R = A^{-1}E^+W. \tag{12}$$

Both A^{-1} and M_0^R are called *the background matrices*.

If there are some solid particles suspended in fluid, some lattice nodes might be covered by solid particle. To keep the dimension of the matrices A and M unchanged, the elements associated with the covered nodes should not be removed from the matrices. However, the interaction between the solid surface and the covered nodes would not be counted. A link connecting a fluid node, which is outside of the solid particle and a node covered by the solid particle, is called a *boundary link*. The bounce-back rule [Eq. (6)] applies on each boundary link, and Eq. (8) then becomes

$$M(t+1) = E^+T_pEM(t) + E^+SV + E^+WU_w, \tag{13}$$

where matrix T_p is different from T only at the elements associated with the boundary links. The matrix S is a $N_c\times 3N_p$ matrix determined by the second term in the bounce-back rule [Eq. (6)]. The E and E^+ are same with or without particles inside the flow.

In LBM the force on solid bodies can be easily calculated in finite schemes. Taking summation of the forces on each link [5]

$$f_k(\mathbf{x}_i)\mathbf{e}_{k'} = [2\delta\phi_{k'}^*(\mathbf{x}_i) - 6w_k\rho_0\mathbf{u}_b\cdot\mathbf{e}_{k'}]\mathbf{e}_{k'}, \tag{14}$$

we have the total force and torques on solid particles in the matrix form as

$$F = QEM + RV + F^{ext}. \tag{15}$$

Both V and F are row vectors with $3N_p$ elements. V is the velocity and angular velocity of each solid particle, while F is the force and torque on each solid particle. In Eq. (15) the $3N_p\times N_c$ matrix Q and the $3N_p\times 3N_p$ matrix R are determined by the first and second terms on the right-hand side of Eq. (14), respectively, while F^{ext} is the external force and torque on the solid particle.

For Stokes flow the force-free and torque-free condition gives [1,2]

$$V = -R^{-1}(QEM + F^{ext}). \tag{16}$$

From Eqs. (11), (13), and (16) the steady state M will be determined by

$$(A + A_1)M = E^+WU_w - E^+SR^{-1}F^{ext}, \tag{17}$$

with

$$A_1 = E^+(T - T_p)E + E^+SR^{-1}QE + J_p. \tag{18}$$

The additional $N_m\times N_m$ matrix J_p is defined as that $J_pM=0$ contains N_p independent constraints for N_p solid particles. Each constraint restricts the total mass at nodes covered by a solid particle to be a constant.

Since the Newtonian equation of motion is coupled with LBE, it is generally believed that the motion of the solid particles could not be calculated unless the motion of the fluid is simultaneously calculated. However, with splitting the coefficients matrix as $A+A_1$, the matrix A_1, after rows reordering, can be converted to a block matrix, with only the first block containing nonzero elements and all other blocks equal to zero. This feature of matrix A_1 leads to a fast algorithm.

The nodes at both ends of the boundary link are called *boundary nodes*. Except for boundary nodes all other nodes covered by the solid particle is called *inside nodes*. Nonzero elements in the matrices $E^+(T-T_p)E$ and $E^+SR^{-1}QE$ come from the boundary nodes, while any nonzero element in the

matrix J_p come from the element $\delta\rho$ at either a boundary node or an inside node. If there are b_j boundary nodes and d_j inside nodes for solid particle j, the components $(\delta\rho, \rho u_x, \rho u_y)$ at each boundary node and $\delta\rho$ at each inside node will span a N_s dimensional space, where $N_s = \Sigma n_j \equiv \Sigma(3b_j + d_j)$. This is a subspace of the fluid phase space, called *the reduced phase space*. Then matrix A_1, after rows reordering, can be written as $A_1 = BCD$, where D is a $N_s \times N_m$ projection matrix with $D_{ij} = \delta_{ij}$, $B = D^T$, and C is a $N_s \times N_s$ matrix in the reduced phase space. Noting that $DB = \hat{I}$ is the $N_s \times N_s$ identity matrix, while BD is a $N_m \times N_m$ block matrix, with the first block equal to $\hat{I}$ and all other blocks equal to zero, Eq. (17) then becomes

$$(A + BCD)M = E^+ W U_w - E^+ S R^{-1} F^{ext}. \tag{19}$$

According to the binomial inverse theorem [13]

$$(A + BCD)^{-1} = A^{-1} - A^{-1}B[C^{-1} + DA^{-1}B]^{-1}DA^{-1},$$

we have

$$M = (I + A^{-1}B\hat{H}D)(M_0^R U_w - A^{-1}E^+ S R^{-1} F^{ext}), \tag{20}$$

where $\hat{H} = -(\hat{I} + C\hat{A}_r)^{-1}C$ and $\hat{A}_r = DA^{-1}B$. Denoting $\hat{Q} = QEB$ and $\hat{M} = DM$, the velocity V can be written as

$$V = -R^{-1}\hat{Q}\hat{M} - R^{-1}F^{ext}, \tag{21}$$

and noting $E^+ S = BDE^+ S$, $\hat{M}$ reads as

$$\hat{M} = (\hat{I} + \hat{A}_r\hat{H})(\hat{M}_0^R U_w - \hat{A}_r\hat{S}R^{-1}F^{ext}), \tag{22}$$

where $\hat{M}_0^R = DM_0^R$ and $\hat{S} = DE^+ S$. Matrices $\hat{A}_r$ and $\hat{M}_0^R$ can be retrieved from background matrices A^{-1} and M_0^R, respectively. Now $\hat{M}$ in the reduced phase space can be calculated from Eq. (22), and V can be obtained from Eq. (21) without full information of the fluid state M. In the calculation we need to invert only an $N_s \times N_s$ matrix $\hat{I} + C\hat{A}_r$, instead of an $N_m \times N_m$ matrix. Compared to full inverting the coefficient matrix, the speedup is proportional to $(N_m/N_s)^3$. Because N_s/N_m is roughly the area (in 2D) or the volume (3D) fraction, this algorithm will result in greatly accelerated simulation speed for a dilute suspension.

In LBM the hydrodynamic force between two surfaces are calculated by patching the leading term of the lubrication force onto the force determined by the lattice-Boltzmann model [3]. For simplicity we consider the lubrication force between two circles. This method can be extended to other particle shapes. According to the lubrication theory the leading term of the lubrication force between the two closely spaced circles i and j is [14] $\mathbf{f}_i^{lub} = -\mathbf{f}_j^{lub} = -\kappa_{ij}(\mathbf{u}_{ij} \cdot \mathbf{g}_{ij})\mathbf{g}_{ij}$, where $\mathbf{f}_i^{lub}$ is the lubrication force on particle i, $\mathbf{u}_{ij}$ is the relative velocity between the two surfaces, and $\mathbf{g}_{ij}$ is the unit vector from the contact point on surface i to the contact point on surface j. The coefficient $\kappa_{ij} = \lambda_1 r_c^{3/2}[\epsilon_{ij}^{-3/2}(1 + \lambda_2\epsilon_{ij}/r_c) - \delta^{-3/2}(1 + \lambda_2\delta/r_c)]$ if $\epsilon_{ij} < \delta$ or $\kappa_{ij} = 0$ otherwise. $r_c = r_i + r_j$, where r_i and r_j are the radius of the two circles, respectively. The two constants here are $\lambda_1 = \sqrt{2}\pi/16 \approx 0.277\,68$ and $\lambda_2 = 3.85$, respectively. $\delta = 2$ is the cut-off range. Then the total lubrication force is calculated as $F^{lub} = R^{lub}V$, where R^{lub} is a $3N_p \times 3N_p$ matrix. When lubrication force is included the matrix R^{-1} in Eqs. (21) and (22) should be replaced by $(R + R^{lub})^{-1}$.

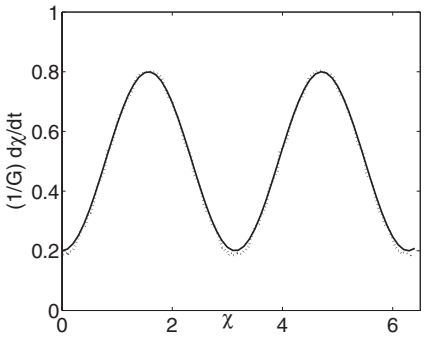

FIG. 1. Rotation of an elliptical cylinder in shear flow. The simulation results (dotted line) are compared with the analytical prediction (solid line) [Eq. (23)].

The positions and the orientations of the solid particles are expressed by a vector X with $3N_p$ components. For given X, all the boundary links can be figured out. Then V, as a function of X, is determined by Eq. (21). Finally, the position of solid particle at the next time step could be obtained by solving the equation $dX/dt = V(X)$ with any numerical solver, such as Euler's method or Runge-Kutta method.

The algorithm presented here is tested by two examples with a computational domain $L \times H = 240 \times 120$. Background matrices A^{-1} and M_0^R are calculated before the simulation. For each test the matrices $\hat{A}_r$ and $\hat{M}_0^R$ are retrieved from the same set of background matrices. Although A is an $N_m \times N_m$ matrix because the computational domain is translational invariant in x direction, we need only calculate $3H$ columns of the inverse matrix. This is a small portion of A^{-1} and takes only about 250 MB storage.

The first test is the rotation of an elliptic cylinder in shear flow. According to [15] the angular velocity of the elliptical cylinder is given by

$$\frac{d\chi}{dt} = \frac{G}{b^2 + c^2}(b^2 \cos^2\chi + c^2 \sin^2\chi), \tag{23}$$

where χ is the angle of the rotation and G is the shear rate. The calculation is carried out for an elliptical cylinder with principal semiaxes $b = 4$ and $c = 8$, respectively. With the ratio $H/c = 15$ the wall effect is negligible [14]. The numerical results obtained by the current method are compared to the theoretical prediction, as shown in Fig. 1. They are in agreement.

A crucial problem for testing the validity of the lubrication force modeling is the motion of a chain of closely spaced particles with a single force along the axis of the chain applied on one particle at the end. The motion of a chain of spheres and a 3D slender body have been studied by Stokesian dynamics [1]. The results show that the single force is sufficient to move the entire chain, as if it were a slender body, and that the dependence of the drag coefficients V/F on the aspect ratio (defined as the total number of the spheres in the chain or the length-to-width ratio of the

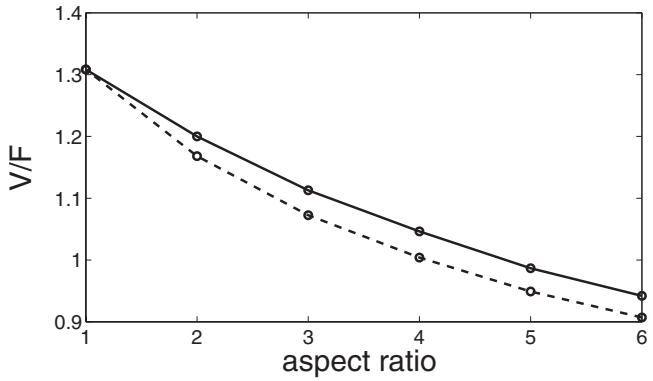

FIG. 2. The drag coefficient of a chain of circles (dashed line) and a slender body (solid line) vs the aspect ratio.

slender body) display similar scaling behavior. However, no LBM simulation to date gives qualitatively correct result. We consider 2D motion of a chain of circles of radius $r=5.0$ with center-to-center space $d_c=10.1$. The results for total number of circles $N_p=1$ to 6 as well as a slender body with various aspect ratios are shown in Fig. 2. It is significant that the values of the drag coefficients decrease as the aspect ratio increases with a similar scaling behavior for both cases. The difference between these two curves may be explained by their different shapes. If the lubrication force were not modeled correctly in this example, then the particles would overlap with one another at the next time step, which is not physically acceptable.

On summary, we developed a fast algorithm to simulate the particle suspension. This algorithm exactly solves steady-state LBE and does not introduce any further approximation. The relevant parameter determining the speedup is the ratio N_s/N_m. For the tests in this Rapid Communication, with $N_s \approx 500$ and $N_m=86\,400$, the ratio $N_s/N_m \approx 10^{-2}$. Hence the calculation speed is approximately 10^6 times faster. On the other hand, to display the snapshot of the fluid flow at a desired moment instead of tracing the fluid motion step by step, it is much faster to calculate the fluid state from Eq. (20) after the motion of the solid particles is already discovered.

I wish to express my gratitude to Dr. M. Ding and Dr. L. Ding for their kind help and inspiring discussion and Professor Per C. Hemmer for his helpful suggestions to this work.

[1] J. F. Brady and G. Bossis, Annu. Rev. Fluid Mech. **20**, 111 (1988).
[2] H. Happel and H. Brenner, *Low Reynolds Number Hydrodynamics with Special Applications to Particulate* (Noordhoff, Lynden, 1973).
[3] N. Q. Nguyen and A. J. C. Ladd, Phys. Rev. E **66**, 046708 (2002).
[4] S. Succi, *The Lattice Boltzmann Equation for Fluid Dynamics and Beyond* (Clarendon Press, Oxford, 2001).
[5] A. J. C. Ladd, J. Fluid Mech. **271**, 285 (1994).
[6] X. He and L. S. Luo, Phys. Rev. E **56**, 6811 (1997).
[7] P. Lallemand and L. S. Luo, Phys. Rev. E **61**, 6546 (2000).
[8] Y. H. Qian, D. D'Humières, and P. Lallemand, Europhys. Lett. **17**, 479 (1992).
[9] H. Chen, S. Chen, and W. H. Matthaeus, Phys. Rev. A **45**, R5339 (1992).
[10] C. K. Aidun, Y. Lu, and E. Ding, J. Fluid Mech. **373**, 287 (1998).
[11] Z. Guo, T. S. Zhao, and Y. Shi, Phys. Rev. E **70**, 066706 (2004).
[12] R. Verberg and A. J. C. Ladd, Phys. Rev. E **60**, 3366 (1999).
[13] G. Strang, *Introduction to Linear Algebra*, 3rd ed. (Wellesley-Cambridge Press, Wellesley, MA, 2003).
[14] J. Kromkamp, D. van den Ende, D. Kandhai, R. van der Sman, and R. Boom, Chem. Eng. Sci. **61**, 858 (2006).
[15] G. B. Jeffery, Proc. R. Soc. London, Ser. A **102**, 161 (1922).