J. Fluid Mech. (2002), *vol.* 452, *pp.* 231–262.
DOI: 10.1017/S0022112001006632 Printed in the United Kingdom

Improvement of the Stokesian Dynamics method for systems with a finite number of particles

By KENGO ICHIKI

Department of Applied Physics, University of Twente, 7500 AE Enschede, The Netherlands

(Received 2 August 2000 and in revised form 27 July 2001)

An improvement of the Stokesian Dynamics method for many-particle systems is presented. A direct calculation of the hydrodynamic interaction is used rather than imposing periodic boundary conditions. The two major difficulties concern the accuracy and the speed of calculations. The accuracy discussed in this work is not concerned with the lubrication correction but, rather, focuses on the multipole expansion which until now has only been formulated up to the so-called FTS version or the first order of force moments. This is improved systematically by a real-space multipole expansion with force moments and velocity moments evaluated at the centre of the particles, where the velocity moments are calculated through the velocity derivatives; the introduction of the velocity derivatives makes the formulation and its extensions straightforward. The reduction of the moments into irreducible form is achieved by the Cartesian irreducible tensor. The reduction is essential to form a well-defined linear set of equations as a generalized mobility problem. The order of truncation is not limited in principle, and explicit calculations of two-body problems are shown with order up to 7. The calculating speed is improved by a conjugate-gradient-type iterative method which consists of a dot-product between the generalized mobility matrix and the force moments as a trial value in each iteration. This provides an $O(N^2)$ scheme where N is the number of particles in the system. Further improvement is achieved by the fast multipole method for the calculation of the generalized mobility problem in each iteration, and an $O(N)$ scheme for the non-adaptive version is obtained. Real problems are studied on systems with $N = 400\,000$ particles. For mobility problems the number of iterations is constant and an $O(N)$ performance is achieved; however for resistance problems the number of iterations increases as almost $N^{1/2}$ with a high accuracy of 10^{-6} and the total cost seems to be $O(N^{3/2})$.

1. Introduction

The microstructure of suspensions is governed by the hydrodynamic interactions among particles immersed in a viscous fluid, which is modelled using the Stokes approximation, and have attracted much attention from researchers in physics and chemical engineering. The hydrodynamic interactions have a long-range nature varying as $1/r$, where r is distance measured from a particle, and further they have a many-body feature, that is, they must be the solution of a boundary-value problem on the surface of all objects in the system. Therefore, analytical approaches are difficult. In fact, even for rigid spherical particles, the exact solution has been obtained only for two-body problems (Jeffrey & Onishi 1984); of course, this is partially because the symmetry of the geometry of surfaces for two-body problems is much simpler than that on systems with three or more particles. Therefore, numerical approaches have

an important role in investigations of such many-body problems. In this context, the Stokesian Dynamics method was developed (Brady & Bossis 1988). The Stokesian Dynamics method is based on Faxén's law and the multipole expansion to obtain the so-called far-field mobility matrix which includes the effect of only low orders of force moments. In addition, the lubrication correction is introduced with the help of the exact solution of the two-body problem. The method gives us very accurate results for particle concentrations from dilute to dense limits.

Many-body problems under the Stokes approximation have two different forms – a finite number of particles in an unbounded fluid and in infinite number of particles using periodic boundary conditions. The difference causes a qualitative change in the behaviour of particles: with a constant force applied, in an unbounded case the particles fall faster as the separation among particles is smaller, while under periodic boundary conditions they fall slower as the concentration of particles is more dense. However the governing equation – the Navier–Stokes equation under the assumption of zero Reynolds number – is linear and the same for both cases, and we could study the problems within the same framework where the appropriate Green function for each situation is used. The Stokesian Dynamics method is one such framework; Durlofsky, Brady & Bossis (1987) describe unbounded cases, and Brady *et al.* (1988) consider periodic boundary conditions. Over ten years have passed since the method was developed, and we are now recognizing two major difficulties with it. One is the limitation of its approximation to the so-called FTS version where only forces, torques and stresslets are considered in the multipole expansion; higher versions were not formulated. The other is that it is a very large and time-intensive calculation which restricts the size of the system that can be simulated to a few hundred particles.

The aim of this paper is to establish a general formulation as a framework for Stokes flows, which can handle problems with as much accuracy as required with less cost of the calculation and without the need for artificial assumptions. We try to formulate such numerical schemes in as simple a manner as possible; this simplicity gives us a good perspective of the physics and easy extensions for various applications.

In this paper, we study a system with a finite number of rigid spherical particles in an unbounded fluid where Brownian motion is negligible, that is, the particle Péclet number is infinite. These problems attract less interest than the problems with periodic boundary conditions. However, this does not mean that the problems are solved; interesting problems remain, for example, breakup of sedimenting agglomerates of particles in a fluid (Nitsche & Batchelor 1997) and dispersion by shear flows (Kao & Mason 1975). Because the main purpose of this paper is to establish the formulation and implementation of numerical schemes, the applications for these phenomena are outside the scope.

Though the lubrication correction is one of the main features of the traditional Stokesian Dynamics method, we do not consider it in this paper. This is because the lubrication correction is an approximation without theoretical justifications for more than two-body systems, and here we aim to establish a framework without unclear assumptions. Although we do not specifically discuss the lubrication correction in this paper, it can be added into this formulation if desired (see § 3.1 for details).

The practical goal of this paper is to improve the accuracy and the calculating speed of the method by Durlofsky *et al.* (1987). The improvement in the accuracy is achieved by a multipole expansion in real space with the proper reduction into the moments which contain only independent elements; we derive a generalized mobility problem relating the force moments to the velocity moments for an arbitrary order of the truncation. The velocity moments are calculated through the velocity derivatives;

this makes the formulation and the extensions straightforward. The improvement in the speed is achieved by iterative methods which are widely used for problems with sparse matrices. Further improvement in the speed is also obtained by the fast multiple method (FMM) developed by Greengard & Rokhlin (1987). The formulation of the FMM in this paper is rather different from the original and is an extension of the multipole expansion for the improvement in the accuracy. We utilize only a plain iterative method and a non-adaptive version of FMM; the preconditioning techniques for iterative methods and the adaptive scheme for the FMM are not discussed here and there is room to improve the present formulation using these techniques.

Because the present formulation is general, the extensions are straightforward. We could extend this formulation to problems with periodic boundary conditions, replacing the Green function from the Oseen tensor by the tensor with the Ewald summation (Beenakker 1986 and Brady *et al.* 1988). Using the proper reduction of the moments, we could also extend this formulation to systems of non-spherical objects. An extension to non-rigid objects could be done by including the double layer potential. Because the formulation does not utilize any special properties of hydrodynamics, this could be also the framework for other problems, such as Laplace problems, linear elastic problems, gravitational systems, and vortex dynamics.

Finally, we comment on the difference of the present formulation from other works. Regarding the improvement of the accuracy, Mazur & van Saarloos (1982) developed the multipole expansion in Fourier space. Replacing the Fourier integral by the Fourier series, Ladd (1988) implemented the formulation for periodic boundary conditions. While the present formulation can be recognized as their real-space version, there is a difference: Mazur & van Saarloos (1982) write the relation between moments formally, and Ladd (1988) implemented that formulation, that is, they treat the moments directly. In our formulation, we calculate the velocity moments through the velocity derivatives. As a result, there are no harmonics or trigonometric functions.

Regarding the improvement of the speed, Sangani & Mo (1996) have applied the FMM to Stokes flows under periodic boundary conditions based on their formulation previously given in Mo & Sangani (1994); their formulation is based on an expansion using harmonic functions. In the present formulation of the FMM, instead of harmonic functions we use an extension of the multipole expansion for the accuracy. Although the spherical harmonics have an advantage for systems with a spherical symmetry, the advantage could be a restriction for systems without such symmetry.

In § 2, we reformulate the multipole expansion and derive a generalized mobility problem. This gives us a systematic improvement on the accuracy of the Stokesian Dynamics method. We demonstrate this for two-body problems. In § 3, we describe efficient numerical schemes – an $O(N^2)$ scheme by an iterative method, and an $O(N)$ scheme by the fast multipole method. The performance of these schemes is demonstrated in § 4, first for a single calculation of a generalized mobility problem which appears in each iteration in the solution of the boundary condition of the physical problems, and second for the physical mobility and resistance problems.

2. Multipole expansion

In this section, we reformulate the multipole expansion method procedure for the hydrodynamic interactions among rigid spherical particles in Stokes flows and give the generalized mobility problem which is an extension of the grand mobility problem in the original Stokesian Dynamics method.

2.1. *Expansion of the velocity field*

The velocity disturbance $\boldsymbol{v}(\boldsymbol{x})$ caused by rigid particles is written in terms of the so-called single-layer potentials (Ladyzhenskaya 1969) as

$$v_i(\boldsymbol{x}) = u_i(\boldsymbol{x}) - u_i^{\infty}(\boldsymbol{x}) = -\frac{1}{8\pi\mu}\sum_{\alpha=1}^{N}\int_{S_\alpha} \mathrm{d}S(\boldsymbol{y})\; J_{ij}(\boldsymbol{x}-\boldsymbol{y}) f_j(\boldsymbol{y}), \tag{2.1}$$

where N is the number of particles, S_α is the surface of particle α, $\boldsymbol{u}$ is the fluid velocity, $\boldsymbol{u}^{\infty}$ is the velocity in the case without particles, μ is the viscosity of the fluid, $\boldsymbol{f}(\boldsymbol{y})$ is a force density on the surface $\boldsymbol{y}$, and $\boldsymbol{J}(\boldsymbol{r})$ is the Oseen tensor defined by

$$J_{ij}(\boldsymbol{r}) = \frac{1}{r}\left(\delta_{ij} + \frac{r_i r_j}{r^2}\right). \tag{2.2}$$

We adopt the Einstein convention for repeated indices throughout this paper. We can expand $\boldsymbol{y}$ on the right-hand side of (2.1) at the centre of particle $\boldsymbol{x}^\alpha$ as

$$v_i(\boldsymbol{x}) = \sum_{\alpha=1}^{N}\sum_{m=0}^{p'} \mathscr{J}^{(m)}_{ij,k\ldots}(\boldsymbol{x}-\boldsymbol{x}^\alpha)\mathscr{F}^{(m)}_{j,k\ldots}(\alpha), \tag{2.3}$$

where p' is the order of truncation (discussed in §2.4 in detail), $\mathscr{F}^{(m)}_{j,k\ldots}(\alpha)$ is the force moment of particle α defined by

$$\mathscr{F}^{(m)}_{j,k\ldots}(\alpha) = -\int_{S_\alpha} \mathrm{d}S(\boldsymbol{y})\; (\boldsymbol{y}-\boldsymbol{x}^\alpha)^m_{k\ldots} f_j(\boldsymbol{y}), \tag{2.4}$$

and $\mathscr{J}^{(m)}_{ij,k\ldots}(\boldsymbol{r})$ is the derivative of the Oseen tensor defined by

$$\mathscr{J}^{(m)}_{ij,k\ldots}(\boldsymbol{r}) = \frac{1}{8\pi\mu}\frac{1}{m!}[(-\nabla)^m_{k\ldots} J_{ij}](\boldsymbol{r}). \tag{2.5}$$

The force F_i, torque T_i, and stresslet S_{ij} are related to the zeroth-order and the first-order force moments as

$$F_i^\alpha = \mathscr{F}^{(0)}_i(\alpha), \tag{2.6}$$

$$T_i^\alpha = \epsilon_{ijk}\mathscr{F}^{(1)}_{k,j}(\alpha), \tag{2.7}$$

and

$$S_{ij}^\alpha = \tfrac{1}{2}\{\mathscr{F}^{(1)}_{i,j}(\alpha) + \mathscr{F}^{(1)}_{j,i}(\alpha) - \tfrac{2}{3}\delta_{ij}\mathscr{F}^{(1)}_{k,k}(\alpha)\}. \tag{2.8}$$

The inverse relations are given by

$$\mathscr{F}^{(0)}_i(\alpha) = F_i^\alpha, \tag{2.9}$$

$$\mathscr{F}^{(1)}_{i,j}(\alpha) - \tfrac{1}{3}\delta_{ij}\mathscr{F}^{(1)}_{k,k}(\alpha) = \tfrac{1}{2}\epsilon_{jik}T_k^\alpha + S_{ij}^\alpha. \tag{2.10}$$

2.2. *Boundary conditions*

Boundary conditions for the velocity on the surface of the particles are satisfied by $\boldsymbol{f}$ in (2.1) or $\mathscr{F}$ in (2.3). In order to specify all elements of $\mathscr{F}$ in (2.3), we need the same number of boundary conditions on the velocity. There are, at least, three approaches – the boundary collocation method, the method using velocity derivatives, and the method using velocity moments.

In the boundary collocation method (Gluckman, Pfeffer & Weinbaum 1971), we directly apply the boundary conditions on a finite number of points on the surface called the collocation points. Therefore, only (2.3) is required to apply the boundary

conditions. While this approach is straightforward, it is not suitable for dynamic problems where the configuration is changing. This is because the results of the boundary collocation method are sensitive to the choice of collocation points and the scheme may fail for some cases. An alternative is to consider velocity derivatives $\mathscr{V}$ at the centre of the particle defined by

$$\mathscr{V}^{(n)}_{i,l\ldots}(\boldsymbol{x}^\alpha) = \frac{1}{n!}[\nabla^n_{l\ldots} v_i](\boldsymbol{x}^\alpha). \tag{2.11}$$

However, the velocity derivatives $\mathscr{V}$ have two disadvantages: the symmetry is different from that of the force moments $\mathscr{F}$, and the velocity derivatives for the 'self-part' becomes singular in the expansion (2.3). The former causes a difference between the numbers of given and unknown parameters to solve, and then the problem may be ill-defined. The latter singularity has to be avoided to obtain the regular solution.

As yet another approach, we introduce velocity moments $\mathscr{U}$ defined by

$$\mathscr{U}^{(n)}_{i,l\ldots}(\alpha) = \frac{1}{4\pi a^2}\int_{S_\alpha} \mathrm{d}S(\boldsymbol{y})\ (\boldsymbol{y}-\boldsymbol{x}^\alpha)^n_{l\ldots} v_i(\boldsymbol{y}), \tag{2.12}$$

where a is the radius of the particles. The velocity moments $\mathscr{U}$ are more complicated than the velocity derivatives $\mathscr{V}$, but the two difficulties are removed. The velocity at the surface is given by $\boldsymbol{v}(\boldsymbol{y}) = \boldsymbol{U}^\alpha + \boldsymbol{\Omega}^\alpha \times (\boldsymbol{y}-\boldsymbol{x}^\alpha) + \boldsymbol{E}^\alpha \cdot (\boldsymbol{y}-\boldsymbol{x}^\alpha)$, where $\boldsymbol{U}^\alpha$, $\boldsymbol{\Omega}^\alpha$, and $\boldsymbol{E}^\alpha$ are the translational velocity, the angular velocity, and the rate of strain for particle α relative to the imposed flow $\boldsymbol{u}^\infty$. Therefore, zeroth-order and first-order velocity moments are written as

$$\mathscr{U}^{(0)}_i(\alpha) = U^\alpha_i, \tag{2.13}$$

and

$$\mathscr{U}^{(1)}_{i,j}(\alpha) = \frac{a^2}{3}(\epsilon_{ikj}\Omega^\alpha_k + E^\alpha_{ij}), \tag{2.14}$$

or equivalently

$$\Omega^\alpha_i = \frac{3}{2a^2}\epsilon_{ijk}\mathscr{U}^{(1)}_{k,j}(\alpha), \tag{2.15}$$

$$E^\alpha_{ij} = \frac{3}{2a^2}\{\mathscr{U}^{(1)}_{i,j}(\alpha) + \mathscr{U}^{(1)}_{j,i}(\alpha)\}. \tag{2.16}$$

If we apply the surface integral in (2.12) to (2.3), the linear set of equations relating the velocity moments and the force moments are obtained as

$$\mathscr{U}^{(n)}_{i,l\ldots}(\alpha) = \sum_{\beta=1}^{N}\sum_{m=0}^{p'} \mathscr{M}^{(n,m)}_{i,l\ldots;j,k\ldots}(\alpha,\beta)\mathscr{F}^{(m)}_{j,k\ldots}(\beta), \tag{2.17}$$

where

$$\mathscr{M}^{(n,m)}_{i,l\ldots;j,k\ldots}(\alpha,\beta) = \frac{1}{4\pi a^2}\int_{S_\alpha} \mathrm{d}S(\boldsymbol{y})(\boldsymbol{y}-\boldsymbol{x}^\alpha)^n_{l\ldots}\ \mathscr{J}^{(m)}_{ij,k\ldots}(\boldsymbol{y}-\boldsymbol{x}^\beta). \tag{2.18}$$

We call (2.17) or its abbreviated form

$$\mathscr{U} = \mathscr{M}\cdot\mathscr{F} \tag{2.19}$$

the generalized mobility problem and the matrix $\mathscr{M}$ the generalized mobility matrix. In the following, for simplicity we often omit indices and arguments in this way.

To solve (2.19), we split the velocity moments $\mathscr{U}$ into two parts – a self-part $\mathscr{U}^s$ and

a non-self-part $\mathscr{U}'$ – as

$$\mathscr{U} = \mathscr{U}^s + \mathscr{U}', \tag{2.20}$$

where a prime denotes the non-self-part. This is because $\mathscr{J}$ is much easier to calculate than $\mathscr{M}$. The self-part $\mathscr{U}^s$ is written as

$$\mathscr{U}^s = \mathscr{M}^s \cdot \mathscr{F}, \tag{2.21}$$

where the self-part of the mobility matrix $\mathscr{M}^s$ is given by

$$\mathscr{M}^{s(n,m)}_{i,l\ldots;j,k\ldots} = \mathscr{M}^{(n,m)}_{i,l\ldots;j,k\ldots}(\alpha,\alpha) = \frac{1}{4\pi a^2}\int_{|\boldsymbol{r}|=a} \mathrm{d}S(\boldsymbol{r}) r^n_{l\ldots}\, \mathscr{J}^{(m)}_{ij,k\ldots}(\boldsymbol{r}). \tag{2.22}$$

It is shown in Appendix A that $\mathscr{M}^{s(n,m)}$ has the following properties:

(i) $\mathscr{M}^{s(n,m)}$ is non-zero only when n and m are both odd or both even;

(ii) $\mathscr{M}^{s(n,m)}$ is zero for $m \geqslant n+2$.

The explicit forms for the zeroth order and the first order are given as

$$\mathscr{M}^{s(0,0)}_{i:j} = \frac{\delta_{ij}}{6\pi\mu a}, \tag{2.23}$$

and

$$\mathscr{M}^{s(1,1)}_{i,l:j,k} = \frac{1}{60\pi\mu a}[4\delta_{ij}\delta_{kl} - \delta_{ik}\delta_{jl} - \delta_{il}\delta_{jk}]. \tag{2.24}$$

For the non-self-part, it is convenient to consider the relation between velocity moments $\mathscr{U}'$ and velocity derivatives $\mathscr{V}'$. The non-self-part of the velocity derivatives $\mathscr{V}'$ is defined by

$$\mathscr{V}'^{(m)}(\alpha) = \frac{1}{m!}[\nabla^m \boldsymbol{v}'^{\alpha}](\boldsymbol{x}^\alpha), \tag{2.25}$$

where the non-self-part of the velocity disturbance caused by particles $\beta \neq \alpha$ is given by

$$v'^{\alpha}_i(\boldsymbol{x}) = \sum_{\beta\neq\alpha}\sum_{m=0}^{p'} \mathscr{J}^{(m)}_{ij,k\ldots}(\boldsymbol{x}-\boldsymbol{x}^\beta)\mathscr{F}^{(m)}_{j,k\ldots}(\beta). \tag{2.26}$$

To obtain the relation between $\mathscr{V}'$ and $\mathscr{U}'$, we expand the velocity $\boldsymbol{v}'^{\alpha}$ at the centre $\boldsymbol{x}^\alpha$ as

$$v'^{\alpha}_i(\boldsymbol{y}) = \sum_{m=0} \mathscr{V}'^{(m)}_{i,k\ldots}(\alpha)(\boldsymbol{y}-\boldsymbol{x}^\alpha)^m_{k\ldots}. \tag{2.27}$$

Applying the surface integral, we obtain the non-self-part of the velocity moment for a particle α in terms of the velocity derivatives as

$$\mathscr{U}'^{(n)}_{i,l\ldots}(\alpha) = \sum_{m=0}^{n+2} \mathscr{V}'^{(m)}_{i,k\ldots}(\alpha)\frac{1}{4\pi a^2}\int_{S_\alpha} \mathrm{d}S(\boldsymbol{y})(\boldsymbol{y}-\boldsymbol{x}^\alpha)^{n+m}_{l\ldots k\ldots}. \tag{2.28}$$

The explicit relations up to second order are

$$\mathscr{U}'^{(0)}_i = \mathscr{V}'^{(0)}_i + \frac{a^2}{3}\mathscr{V}'^{(2)}_{i,jj}, \tag{2.29}$$

$$\mathscr{U}'^{(1)}_{i,k} = \frac{a^2}{3}\mathscr{V}'^{(1)}_{i,k} + \frac{a^4}{5}\mathscr{V}'^{(3)}_{i,kjj}, \tag{2.30}$$

$$\mathscr{U}'^{(2)}_{i,kl} = \frac{a^2}{3}\delta_{kl}\mathscr{V}'^{(0)}_i + \frac{a^4}{15}(\delta_{kl}\mathscr{V}'^{(2)}_{i,jj} + 2\mathscr{V}'^{(2)}_{i,kl}) + \frac{4a^6}{35}\mathscr{V}'^{(4)}_{i,kljj}. \tag{2.31}$$

Three remarks can be made pertaining to (2.28). First, only even (odd) m are required when n is even (odd), because the integral on the right-hand side is a linear combination of Kronecker's delta (see Appendix A). Second, the upper limit of the m-summation comes from the biharmonic nature of $\boldsymbol{v}$,

$$\nabla^2\nabla^2\boldsymbol{v} = \boldsymbol{0}. \tag{2.32}$$

Therefore, the relation (2.28) gives the exact transformation from $\mathscr{V}'$ to $\mathscr{U}'$. The fact that the upper limit of the summation of m in (2.28) is not n but $n+2$ means that the finite-size effect is taken into account in $\mathscr{U}'^{(m)}$ by the trace of $\mathscr{V}'^{(m+2)}$. The final remark concerns the incompressibility

$$\nabla\cdot\boldsymbol{v} = 0. \tag{2.33}$$

The velocity derivatives satisfy $\mathscr{V}'^{(n)}_{i,ij\ldots} = 0$ for $n \geqslant 1$. On the other hand, the similar condition for the velocity moments is required only at the first order, that is,

$$\mathscr{U}'^{(1)}_{i,i} = 0, \tag{2.34}$$

and $\mathscr{U}'^{(n)}_{i,ij\ldots} \neq 0$ for $n \geqslant 2$ in general. The terms in $\mathscr{U}'^{(n)}_{i,ij\ldots}$ always contain non-zero velocity derivatives such as $\mathscr{V}'^{(n)}_{j,iik\ldots} \neq 0$ for $n \geqslant 2$. Therefore, we do not need to consider the reduction of the velocity moments due to the incompressibility on orders $n \geqslant 2$.

2.3. *Reduction of moments*

In resistance problems, for example, where particle velocities are obtained from the applied forces, we solve the generalized mobility problem (2.19) for the force moment $\mathscr{F}$ using the velocity moment $\mathscr{U}$. Even in mobility problems, higher elements of the force moments are unknown, while those of the velocity moments are given from the rigidity of the particles (see § 2.5 for details). The elements of force and velocity moments are not independent of each other as we see in (2.34), and the linear set of equations (2.19) is ill-defined. To obtain the right solution, we need to reduce the equations to those relating the irreducible moments whose elements are all independent. The reduction is related to the nature of the velocity field itself – the incompressibility and the biharmonic nature – as was discussed in § 2.2. There is another dependence among elements of the moments from the nature of spherical particles. We discuss only the velocity moments $\mathscr{U}$ here, but the following discussion is applicable for the force moments $\mathscr{F}$ as well.

For velocity moments $\mathscr{U}^{(m)}_{i,l\ldots}$, interchange of any two indices on $l\ldots$ makes no difference. From this property, the independent number of elements at mth order becomes $(m+1)(m+2)/2$. We call the form of this reduction the 'symmetric form' (see table 1).

From the definition of the moments, the higher rank depends on the lower rank in the following way:

$$\mathscr{U}^{(n+2)}_{j,ssk\ldots} = a^2\mathscr{U}^{(n)}_{j,k\ldots}. \tag{2.35}$$

To reduce this dependence, it is convenient to introduce the irreducible tensor which is symmetric and traceless. The reduction for a p-rank tensor $A^p_{i\ldots}$ is given by Damour & Iyer 1991 as

$$\hat{A}^p_{i\ldots} = \sum_{k=0}^{[p/2]} a^p_k \delta_{(i_1 i_2}\delta_{i_3 i_4}\ldots\delta_{i_{2k-1}i_{2k}}A^p_{i_{2k+1}\ldots i_p)s_1 s_1\ldots s_k s_k}, \tag{2.36}$$

Order m	Full form	Symmetric form	Irreducible form
0	$\mathcal{U}^{(0)}_{x}, \mathcal{U}^{(0)}_{y}, \mathcal{U}^{(0)}_{z}$	$\mathcal{U}^{(0)}_{x}, \mathcal{U}^{(0)}_{y}, \mathcal{U}^{(0)}_{z}$	$\hat{\mathcal{U}}^{(0)}_{x}, \hat{\mathcal{U}}^{(0)}_{y}, \hat{\mathcal{U}}^{(0)}_{z}$
1	$\mathcal{U}^{(1)}_{x,x}, \mathcal{U}^{(1)}_{x,y}, \mathcal{U}^{(1)}_{x,z}$	$\mathcal{U}^{(1)}_{x,x}, \mathcal{U}^{(1)}_{x,y}, \mathcal{U}^{(1)}_{x,z}$	$-^{a}, \hat{\mathcal{U}}^{(1)}_{x,y}, \hat{\mathcal{U}}^{(1)}_{x,z}$
	$\mathcal{U}^{(1)}_{y,x}, \mathcal{U}^{(1)}_{y,y}, \mathcal{U}^{(1)}_{y,z}$	$\mathcal{U}^{(1)}_{y,x}, \mathcal{U}^{(1)}_{y,y}, \mathcal{U}^{(1)}_{y,z}$	$\hat{\mathcal{U}}^{(1)}_{y,x}, \hat{\mathcal{U}}^{(1)}_{y,y}, \hat{\mathcal{U}}^{(1)}_{y,z}$
	$\mathcal{U}^{(1)}_{z,x}, \mathcal{U}^{(1)}_{z,y}, \mathcal{U}^{(1)}_{z,z}$	$\mathcal{U}^{(1)}_{z,x}, \mathcal{U}^{(1)}_{z,y}, \mathcal{U}^{(1)}_{z,z}$	$\hat{\mathcal{U}}^{(1)}_{z,x}, \hat{\mathcal{U}}^{(1)}_{z,y}, \hat{\mathcal{U}}^{(1)}_{z,z}$
2	$\mathcal{U}^{(2)}_{\cdot,xx}, \mathcal{U}^{(2)}_{\cdot,xy}, \mathcal{U}^{(2)}_{\cdot,xz}$	$\mathcal{U}^{(2)}_{\cdot,xx}, \mathcal{U}^{(2)}_{\cdot,xy}, \mathcal{U}^{(2)}_{\cdot,xz}$	$-^{b}, \hat{\mathcal{U}}^{(2)}_{\cdot,xy}, \hat{\mathcal{U}}^{(2)}_{\cdot,xz}$
	$\mathcal{U}^{(2)}_{\cdot,yx}, \mathcal{U}^{(2)}_{\cdot,yy}, \mathcal{U}^{(2)}_{\cdot,yz}$	$-^{c}, \mathcal{U}^{(2)}_{\cdot,yy}, \mathcal{U}^{(2)}_{\cdot,yz}$	$-^{c}, \hat{\mathcal{U}}^{(2)}_{\cdot,yy}, \hat{\mathcal{U}}^{(2)}_{\cdot,yz}$
	$\mathcal{U}^{(2)}_{\cdot,zx}, \mathcal{U}^{(2)}_{\cdot,zy}, \mathcal{U}^{(2)}_{\cdot,zz}$	$-^{c}, -^{c}, \mathcal{U}^{(2)}_{\cdot,zz}$	$-^{c}, -^{c}, \hat{\mathcal{U}}^{(2)}_{\cdot,zz}$
3	$\mathcal{U}^{(3)}_{\cdot,xxx}, \mathcal{U}^{(3)}_{\cdot,xxy}, \mathcal{U}^{(3)}_{\cdot,xxz}$	$\mathcal{U}^{(3)}_{\cdot,xxx}, \mathcal{U}^{(3)}_{\cdot,xxy}, \mathcal{U}^{(3)}_{\cdot,xxz}$	$-^{b}, -^{b}, -^{b}$
	$\mathcal{U}^{(3)}_{\cdot,xyx}, \mathcal{U}^{(3)}_{\cdot,xyy}, \mathcal{U}^{(3)}_{\cdot,xyz}$	$-^{c}, \mathcal{U}^{(3)}_{\cdot,xyy}, \mathcal{U}^{(3)}_{\cdot,xyz}$	$-^{c}, \hat{\mathcal{U}}^{(2)}_{\cdot,xyy}, \hat{\mathcal{U}}^{(2)}_{\cdot,xyz}$
	$\mathcal{U}^{(3)}_{\cdot,xzx}, \mathcal{U}^{(3)}_{\cdot,xzy}, \mathcal{U}^{(3)}_{\cdot,xzz}$	$-^{c}, -^{c}, \mathcal{U}^{(3)}_{\cdot,xzz}$	$-^{c}, -^{c}, \hat{\mathcal{U}}^{(2)}_{\cdot,xzz}$
	$\mathcal{U}^{(3)}_{\cdot,yxx}, \mathcal{U}^{(3)}_{\cdot,yxy}, \mathcal{U}^{(3)}_{\cdot,yxz}$	$-^{c}, -^{c}, -^{c}$	$-^{c}, -^{c}, -^{c}$
	$\mathcal{U}^{(3)}_{\cdot,yyx}, \mathcal{U}^{(3)}_{\cdot,yyy}, \mathcal{U}^{(3)}_{\cdot,yyz}$	$-^{c}, \mathcal{U}^{(3)}_{\cdot,yyy}, \mathcal{U}^{(3)}_{\cdot,yyz}$	$-^{c}, \hat{\mathcal{U}}^{(2)}_{\cdot,yyy}, \hat{\mathcal{U}}^{(2)}_{\cdot,yyz}$
	$\mathcal{U}^{(3)}_{\cdot,yzx}, \mathcal{U}^{(3)}_{\cdot,yzy}, \mathcal{U}^{(3)}_{\cdot,yzz}$	$-^{c}, -^{c}, \mathcal{U}^{(3)}_{\cdot,yzz}$	$-^{c}, -^{c}, \hat{\mathcal{U}}^{(2)}_{\cdot,yzz}$
	$\mathcal{U}^{(3)}_{\cdot,zxx}, \mathcal{U}^{(3)}_{\cdot,zxy}, \mathcal{U}^{(3)}_{\cdot,zxz}$	$-^{c}, -^{c}, -^{c}$	$-^{c}, -^{c}, -^{c}$
	$\mathcal{U}^{(3)}_{\cdot,zyx}, \mathcal{U}^{(3)}_{\cdot,zyy}, \mathcal{U}^{(3)}_{\cdot,zyz}$	$-^{c}, -^{c}, -^{c}$	$-^{c}, -^{c}, -^{c}$
	$\mathcal{U}^{(3)}_{\cdot,zzx}, \mathcal{U}^{(3)}_{\cdot,zzy}, \mathcal{U}^{(3)}_{\cdot,zzz}$	$-^{c}, -^{c}, \mathcal{U}^{(3)}_{\cdot,zzz}$	$-^{c}, -^{c}, \hat{\mathcal{U}}^{(2)}_{\cdot,zzz}$

TABLE 1. Elements of the moments in three forms – 'full', 'symmetric', and 'irreducible'. The reduction denoted by $-^{a}$ is of the incompressibility $\mathcal{U}_{i,i} = 0$. The reduction denoted by $-^{b}$ and $-^{c}$ are of the irreducibility and the symmetry on $k\ldots$ in $\mathcal{U}_{i,k\ldots}$ respectively. The first index on the moments with $m \geqslant 2$ is omitted because they are the same.

where

$$a^{p}_{k} = (-1)^{k} \frac{p!}{(p-2k)!} \frac{(2p-2k-1)!!}{(2p-1)!!(2k)!!}, \tag{2.37}$$

and $n!!$ means $n(n-2)(n-4)\ldots 3\times 1$ for odd n and $n(n-2)(n-4)\ldots 4\times 2$ for even n, and $0!! = 1$. The parentheses around the indices in (2.36) indicate the symmetrization for the indices. For example, we have the following relations for $p = 2$ and 3:

$$\hat{A}^{2}_{ij} = A^{2}_{(ij)} - \tfrac{1}{3}\delta_{ij}A^{2}_{ss}, \tag{2.38}$$

and

$$\hat{A}^{3}_{ijk} = A^{3}_{(ijk)} - \tfrac{1}{5}(\delta_{ij}A^{3}_{(kss)} + \delta_{jk}A^{3}_{(iss)} + \delta_{ki}A^{3}_{(jss)}). \tag{2.39}$$

For the moments in our case, the indices are symmetric by definition, so that we need not consider symmetrization. By this reduction, the number of independent elements at mth order becomes $2m+1$. The reduced elements are summarized in table 1.

We write this reduction operator as $\mathscr{P}$ and the inverse operator (recovery operator) as $\mathscr{Q}$. The explicit forms of $\mathscr{P}$ and $\mathscr{Q}$ are given in Appendix B. By these operators, irreducible moments $\hat{\mathscr{U}}$ and $\hat{\mathscr{F}}$ are related as

$$\hat{\mathscr{U}} = \mathscr{P}\cdot\mathscr{M}\cdot\mathscr{Q}\cdot\hat{\mathscr{F}}, \tag{2.40}$$

which we call the irreducible generalized mobility problem. The procedure to calculate (2.40) is discussed in § 2.4 and the application of the boundary conditions to it is discussed in § 2.5.

2.4. *Truncation*

The truncation implicitly introduced in (2.3) should be considered in the irreducible form (2.40) where the independent elements are explicitly specified; we introduce the order of truncation p as the maximum order of $\hat{\boldsymbol{\mathcal{U}}}$ and $\hat{\boldsymbol{\mathcal{F}}}$ in (2.40).

Equation (2.40) is calculated using the following six-step procedure for particles $\alpha = 1, \ldots, N$ where we write the order of truncation p explicitly:

(i) Recover the force moments $\boldsymbol{\mathcal{F}}$ from the irreducible force moments $\hat{\boldsymbol{\mathcal{F}}}$ as the input by

$$\begin{bmatrix} \boldsymbol{\mathcal{F}}^{(0)} \\ \vdots \\ \boldsymbol{\mathcal{F}}^{(p)} \\ \boldsymbol{\mathcal{F}}^{(p+1)} \\ \boldsymbol{\mathcal{F}}^{(p+2)} \end{bmatrix}(\alpha) = \begin{bmatrix} \boldsymbol{\mathcal{Q}}^{(0,0)} & \cdots & \boldsymbol{\mathcal{Q}}^{(0,p)} \\ \vdots & \ddots & \vdots \\ \boldsymbol{\mathcal{Q}}^{(p,0)} & \cdots & \boldsymbol{\mathcal{Q}}^{(p,p)} \\ \boldsymbol{\mathcal{Q}}^{(p+1,0)} & \cdots & \boldsymbol{\mathcal{Q}}^{(p+1,p)} \\ \boldsymbol{\mathcal{Q}}^{(p+2,0)} & \cdots & \boldsymbol{\mathcal{Q}}^{(p+2,p)} \end{bmatrix} \cdot \begin{bmatrix} \hat{\boldsymbol{\mathcal{F}}}^{(0)} \\ \vdots \\ \hat{\boldsymbol{\mathcal{F}}}^{(p)} \end{bmatrix}(\alpha). \tag{2.41}$$

(ii) Calculate the non-self-part of the velocity derivatives $\boldsymbol{\mathcal{V}}'$ from $\boldsymbol{\mathcal{F}}$ by

$$\begin{bmatrix} \boldsymbol{\mathcal{V}}'^{(0)} \\ \vdots \\ \boldsymbol{\mathcal{V}}'^{(p+2)} \end{bmatrix}(\alpha) = \sum_{\beta \neq \alpha} \begin{bmatrix} \boldsymbol{\mathcal{K}}^{(0,0)} & \cdots & \boldsymbol{\mathcal{K}}^{(0,p+2)} \\ \vdots & \ddots & \vdots \\ \boldsymbol{\mathcal{K}}^{(p+2,0)} & \cdots & \boldsymbol{\mathcal{K}}^{(p+2,p+2)} \end{bmatrix}(\boldsymbol{x}^\alpha - \boldsymbol{y}^\beta) \cdot \begin{bmatrix} \boldsymbol{\mathcal{F}}^{(0)} \\ \vdots \\ \boldsymbol{\mathcal{F}}^{(p+2)} \end{bmatrix}(\beta), \tag{2.42}$$

where

$$\mathcal{K}^{(n,m)}_{i,l\cdots;j,k\cdots}(\boldsymbol{r}) = \frac{1}{n!}[\nabla^n_{l\cdots}\mathcal{J}^{(m)}_{ij,k\cdots}](\boldsymbol{r}) = \frac{1}{8\pi\mu}\frac{1}{n!}\frac{1}{m!}[\nabla^n_{l\cdots}(-\nabla)^m_{k\cdots}J_{ij}](\boldsymbol{r}). \tag{2.43}$$

(iii) Convert the velocity derivatives $\boldsymbol{\mathcal{V}}'$ to the velocity moments $\boldsymbol{\mathcal{U}}'$ by (2.28) as

$$\begin{bmatrix} \boldsymbol{\mathcal{U}}'^{(0)} \\ \vdots \\ \boldsymbol{\mathcal{U}}'^{(p)} \end{bmatrix}(\alpha) = \begin{bmatrix} \mathcal{D}^{(0,0)} & \cdots & \mathcal{D}^{(0,p)} & \mathcal{D}^{(0,p+1)} & \mathcal{D}^{(0,p+2)} \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ \mathcal{D}^{(p,0)} & \cdots & \mathcal{D}^{(p,p+2)} & \mathcal{D}^{(p,p+1)} & \mathcal{D}^{(p,p+2)} \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{\mathcal{V}}'^{(0)} \\ \vdots \\ \boldsymbol{\mathcal{V}}'^{(p)} \\ \boldsymbol{\mathcal{V}}'^{(p+1)} \\ \boldsymbol{\mathcal{V}}'^{(p+2)} \end{bmatrix}(\alpha), \tag{2.44}$$

where

$$\mathcal{D}^{(n,m)} = \frac{a^{n+m}}{4\pi}\int_{|\hat{\boldsymbol{r}}|=1} \mathrm{d}S(\hat{\boldsymbol{r}})\hat{\boldsymbol{r}}^{n+m}. \tag{2.45}$$

(iv) Calculate the self-part of the velocity moment $\boldsymbol{\mathcal{U}}^s$ by

$$\begin{bmatrix} \boldsymbol{\mathcal{U}}^{s(0)} \\ \vdots \\ \boldsymbol{\mathcal{U}}^{s(p)} \end{bmatrix}(\alpha) = \begin{bmatrix} \boldsymbol{\mathcal{M}}^{s(0,0)} & \cdots & \boldsymbol{\mathcal{M}}^{s(0,p)} \\ \vdots & \ddots & \vdots \\ \boldsymbol{\mathcal{M}}^{s(p,0)} & \cdots & \boldsymbol{\mathcal{M}}^{s(p,p)} \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{\mathcal{F}}^{(0)} \\ \vdots \\ \boldsymbol{\mathcal{F}}^{(p)} \end{bmatrix}(\alpha), \tag{2.46}$$

where $\boldsymbol{\mathcal{M}}^s$ is given by (2.22).

(v) Calculate the velocity moments $\boldsymbol{\mathcal{U}}$ summing the self-part and the non-self-part

as

$$\begin{bmatrix} \mathscr{U}^{(0)} \\ \vdots \\ \mathscr{U}^{(p)} \end{bmatrix}(\alpha) = \begin{bmatrix} \mathscr{U}^{s(0)} \\ \vdots \\ \mathscr{U}^{s(p)} \end{bmatrix}(\alpha) + \begin{bmatrix} \mathscr{U}^{\prime(0)} \\ \vdots \\ \mathscr{U}^{\prime(p)} \end{bmatrix}(\alpha). \tag{2.47}$$

(vi) Reduce the velocity moments $\mathscr{U}$ into $\hat{\mathscr{U}}$ as

$$\begin{bmatrix} \hat{\mathscr{U}}^{(0)} \\ \vdots \\ \hat{\mathscr{U}}^{(p)} \end{bmatrix}(\alpha) = \begin{bmatrix} \mathscr{P}^{(0,0)} & \cdots & \mathscr{P}^{(0,p)} \\ \vdots & \ddots & \vdots \\ \mathscr{P}^{(p,0)} & \cdots & \mathscr{P}^{(p,p)} \end{bmatrix} \cdot \begin{bmatrix} \mathscr{U}^{(0)} \\ \vdots \\ \mathscr{U}^{(p)} \end{bmatrix}(\alpha). \tag{2.48}$$

The above six-step procedure can be recognised as a subroutine of (2.40) which has the irreducible force moment $\hat{\mathscr{F}}$ as the input and returns the irreducible velocity moment $\hat{\mathscr{U}}$ as output.

In the Stokesian Dynamics method, three types of truncation – F, FT and FTS versions – are presented. The F version corresponds to the truncation $p = 0$ where the generalized mobility matrix relates the force $\boldsymbol{F}$ to the translational velocity $\boldsymbol{U}$. In the FT version, the torque $\boldsymbol{T}$ that is the asymmetric part of the first-order force moments is also taken into account and the corresponding angular velocity $\boldsymbol{\Omega}$ is considered. In the FTS version, the stresslet $\boldsymbol{S}$ that is the remnant symmetric part of the first-order force moments and the corresponding rate of strain $\boldsymbol{E}$ are considered, corresponding to the truncation of $p = 1$. The formulation in this paper is completely equivalent to that in Durlofsky *et al.* (1987) up to the FTS version.

We note that even in the truncation at order p of $\hat{\mathscr{F}}$ and $\hat{\mathscr{U}}$ in (2.40), we have to recover the force moments up to the order $p + 2$, calculate the velocity derivatives at order $p + 2$, and convert them into the velocity moments at the order p. Otherwise, we would lose the finite-size effect. In fact, even for $p = 0$, the finite-size effect is considered as the trace of the second-order derivatives of the Green function and we obtain the Rotne–Prager tensor as the generalized mobility matrix. Therefore, the truncation order p' in (2.3), (2.17), and (2.26), is $p + 2$; however, the recovered moments $\mathscr{F}^{(p+1)}$ and $\mathscr{F}^{(p+2)}$ do not contain the contribution of $\hat{\mathscr{F}}^{(p+1)}$ nor $\hat{\mathscr{F}}^{(p+2)}$.

2.5. *Higher-order versions for rigid particles*

From the rigidity of the surface, there are $6N$ degrees of freedom for rigid particles; independent variables are force $\boldsymbol{F}$, torque $\boldsymbol{T}$, translational velocity $\boldsymbol{U}$ and angular velocity $\boldsymbol{\Omega}$, and the irreducible moments of the velocity with higher orders should vanish. This means that the higher moments of velocity are prescribed and the corresponding higher moments of forces are solved for both mobility and resistance problems. The contracted mobility and resistance matrices which relate the independent variables for rigid particles have $6N \times 6N$-dimensions, depend on the order of the truncation p, and converge to the exact solution as $p \to \infty$. In linear flows including the rate of strain, the degrees of freedom are the same as FTS version. We call the former the FT-contraction and the latter the FTS-contraction.

We denote the lower moments by subscript l which correspond to the relevant parts ($\boldsymbol{F}$ and $\boldsymbol{T}$ in the FT-contraction and $\boldsymbol{F}$, $\boldsymbol{T}$, and $\boldsymbol{S}$ in the FTS-contraction) and the higher moments by subscript h which could be contracted. Then, we can rewrite

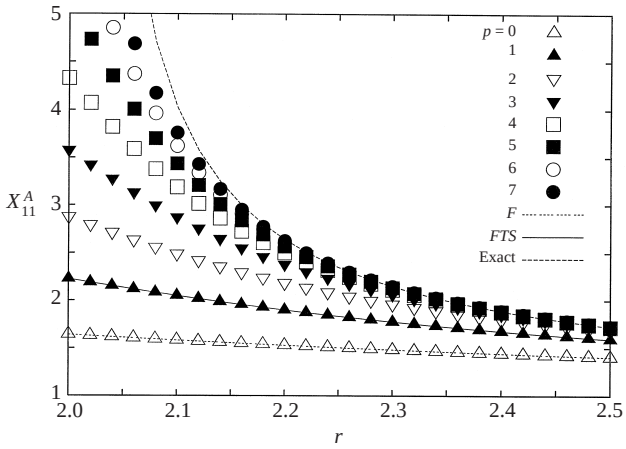

FIGURE 1. Scalar functions $X_{11}^A(r;p)$ of the two-body resistance problem for various orders of the truncation. 'F' and 'FTS' show the corresponding values obtained by the analytical expressions, and 'Exact' shows the exact solution by Jeffrey & Onishi (1984). The results by the present formulation are shown as $p = 0, 1, \ldots, 7$.

(2.40) as

$$\begin{bmatrix} \hat{\boldsymbol{\mathcal{U}}}_l \\ \hat{\boldsymbol{\mathcal{U}}}_h \end{bmatrix} = \begin{bmatrix} \hat{\boldsymbol{\mathcal{M}}}_{ll} & \hat{\boldsymbol{\mathcal{M}}}_{lh} \\ \hat{\boldsymbol{\mathcal{M}}}_{hl} & \hat{\boldsymbol{\mathcal{M}}}_{hh} \end{bmatrix} \cdot \begin{bmatrix} \hat{\boldsymbol{\mathcal{F}}}_l \\ \hat{\boldsymbol{\mathcal{F}}}_h \end{bmatrix}. \tag{2.49}$$

From the rigidity $\hat{\boldsymbol{\mathcal{U}}}_h = 0$, the corresponding force moment $\hat{\boldsymbol{\mathcal{F}}}_h$ is solved as

$$\hat{\boldsymbol{\mathcal{F}}}_h = -(\hat{\boldsymbol{\mathcal{M}}}_{hh})^{-1} \cdot \hat{\boldsymbol{\mathcal{M}}}_{hl} \cdot \hat{\boldsymbol{\mathcal{F}}}_l. \tag{2.50}$$

Therefore, we have the contracted mobility problem as

$$\hat{\boldsymbol{\mathcal{U}}}_l = \hat{\boldsymbol{\mathcal{M}}}^*(p) \cdot \hat{\boldsymbol{\mathcal{F}}}_l, \tag{2.51}$$

where the contracted mobility matrix $\hat{\boldsymbol{\mathcal{M}}}^*(p)$ is given by

$$\hat{\boldsymbol{\mathcal{M}}}^*(p) = \hat{\boldsymbol{\mathcal{M}}}_{ll} - \hat{\boldsymbol{\mathcal{M}}}_{lh} \cdot (\hat{\boldsymbol{\mathcal{M}}}_{hh})^{-1} \cdot \hat{\boldsymbol{\mathcal{M}}}_{hl}. \tag{2.52}$$

This matrix depends on the truncation p and is exact as $p \to \infty$. Its inverse $\hat{\boldsymbol{\mathcal{R}}}^*(p) = (\hat{\boldsymbol{\mathcal{M}}}^*)^{-1}$ is the corresponding contracted resistance matrix which gives the contracted resistance problem

$$\hat{\boldsymbol{\mathcal{F}}}_l = \hat{\boldsymbol{\mathcal{R}}}^*(p) \cdot \hat{\boldsymbol{\mathcal{U}}}_l. \tag{2.53}$$

The dimensions of $\hat{\boldsymbol{\mathcal{M}}}^*(p)$ and $\hat{\boldsymbol{\mathcal{R}}}^*(p)$ are $6N \times 6N$ for the FT-contraction and $11N \times 11N$ for the FTS-contraction.

In the calculation of the contracted problems, we do not want to treat the irreducible mobility matrix $\hat{\boldsymbol{\mathcal{M}}}$ nor its decomposition in (2.49) explicitly. The six-step procedure in § 2.4 meets this requirement; the procedure to calculate $\hat{\boldsymbol{\mathcal{U}}}$ from $\hat{\boldsymbol{\mathcal{F}}}$ is sufficient to solve the linear equations by iterative methods. (Examples of treatments are shown in Appendix C.)

For the tests of the formulation and the implementation, we solve the two-body problems and compare them to the exact solution by Jeffrey & Onishi (1984). First, we calculate the resistance problem for $p = 0, 1, \ldots, 7$. Figure 1 shows one of the scalar functions in the resistance matrix $X_{11}^A(r;p)$ where r is the distance between

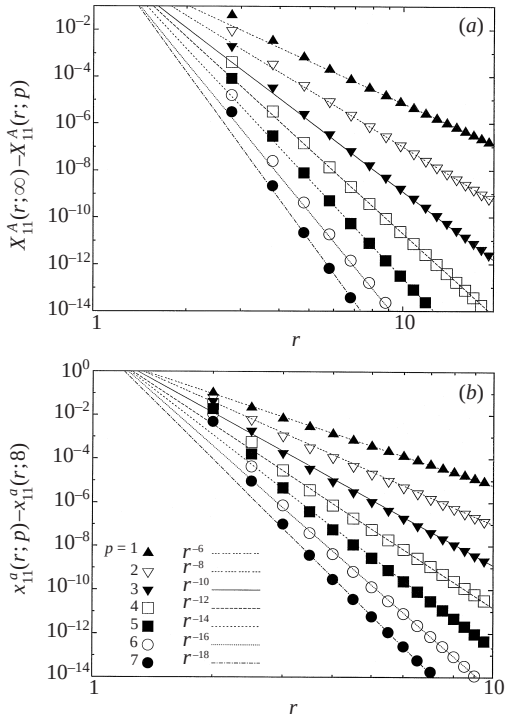

FIGURE 2. Errors of a scalar function on two-body problems: (*a*) the error of resistance functions $X^A_{11}(r;p)$ from the exact solution $X^A_{11}(r;p=\infty)$ and (*b*) that of mobility functions $x^a_{11}(r;p)$ from a higher solution $x^a_{11}(r;p=8)$.

the spheres divided by the radius. This shows that the results converge to the exact solution as $p \to \infty$, and those of $p = 0$ and $p = 1$ are completely identical to the analytical expression in the F version

$$X^A_{11}(r;0) = \frac{4r^6}{4r^6 - (3r^2-2)^2}, \tag{2.54}$$

and that in the FTS version

$$X^A_{11}(r;1) = \frac{20r^6(-2880 + 2208r^2 - 260r^4 - 75r^6 + 20r^{10})}{2304 - 21120r^2 + 55600r^4 - 90600r^6 + 45945r^8 - 800r^{10} - 1800r^{12} - 900r^{14} + 400r^{16}}. \tag{2.55}$$

To estimate the truncation errors of the formulation quantitatively, we compare the truncated solutions to the exact solution for resistance problems in figure 2(*a*), and the truncated solutions to that with a higher truncation $p = 8$ for mobility problems in figure 2(*b*). We see that for both resistance and mobility problems, the errors are scaled by $r^{-2(p+2)}$. We note that the errors of the resistance function Y^A_{11} and the mobility function y^a_{11} have the same scaling and others have higher orders, so that the leading error in the formulation is $O(r^{-2(p+2)})$ for large r.

We can understand the order as follows. We consider the mobility problem first. On the truncation p, the leading error comes from the contribution of $\mathscr{F}^{(p+1)}$ which would appear in $\hat{\mathscr{M}}_{lh} \cdot (\hat{\mathscr{M}}_{hh})^{-1} \cdot \hat{\mathscr{M}}_{hl}$ with higher truncations. Because the self-part

which has r^0 scaling appears only in $\hat{\mathscr{M}}_{hh}$ (and, therefore, its inverse), the leading error relating to $\mathscr{F}^{(p+1)}$ comes from the lowest order in $\hat{\mathscr{M}}_{lh}$ and $\hat{\mathscr{M}}_{hl}$ that has $r^{-(p+2)}$ scaling. Therefore, the error in mobility problems is scaled by $r^{-2(p+2)}$. In fact, not only for mobility problems but also for resistance problems, $\hat{\mathscr{U}}_h$ is always specified and $\hat{\mathscr{F}}_h$ is unknown for rigid particles. Therefore, even in resistance problems, the truncation error comes from the contribution of $\mathscr{F}^{(p+1)}$ and the error on the velocity moments $\hat{\mathscr{U}}_l$ has $r^{-2(p+2)}$ scaling. Because the connection with the lowest order from $\hat{\mathscr{U}}_l$ to $\hat{\mathscr{F}}_l$ is in the self-part, the error of resistance problems is also scaled by $r^{-2(p+2)}$.

3. Fast scheme

The bottleneck in the Stokesian Dynamics method is in the inversion of the mobility matrix which has $O(N^3)$ cost of calculations. This calculation appears to introduce the lubrication correction and also to solve the linear equations as in (2.52). Therefore, we need to improve the calculation of the linear set of equations to be faster than $O(N^3)$. As suggested by Ichiki & Brady (2001), the application of conjugate-gradient-type iterative methods is the first step in the improvement. The iterative method for the Stokesian Dynamics method gives an $O(N^2)$ scheme which consists of the calculation of the dot-product between a mobility matrix and a force moment. This calculation is the next bottleneck. The fast multipole method, which is a simple extension of the conventional multipole expansion, is applicable for the calculation and gives an $O(N)$ scheme.

To eliminate confusion, we note that the terms such as $O(N^2)$ and $O(N)$ in this paper are for the calculation cost under fast convergence of the iterative procedure, or more precisely, for that of a single iteration. Therefore, if we need a large number of iterations to solve the problem with the scaling N^α, for example, the total calculation has an $O(N^{2+\alpha})$ cost for the $O(N^2)$ scheme and an $O(N^{1+\alpha})$ cost for the $O(N)$ scheme. Further discussion is given in § 4.

3.1. *Iterative method – $O(N^2)$ scheme*

We summarize what the iterative methods are, and discuss their application to the current problems. Let us consider a standard form of a linear set of equations,

$$\boldsymbol{b} = \boldsymbol{A} \cdot \boldsymbol{x}, \tag{3.1}$$

where a coefficient matrix $\boldsymbol{A}$ and a vector $\boldsymbol{b}$ are given and the vector $\boldsymbol{x}$ is to be determined. The conjugate-gradient-type iterative method consists only of the calculation of the dot-product between the coefficient matrix $\boldsymbol{A}$ and an arbitrary vector $\boldsymbol{y}$. Therefore, it becomes very efficient when the calculation of $\boldsymbol{A} \cdot \boldsymbol{y}$ is as fast as for sparse-matrix problems. This also means that even for dense matrices like the current problems, the method gives the result at the cost of the dot-product calculation under fast convergence.

Now we consider the linear equation (2.49). For resistance problems, the coefficient matrix is $\hat{\mathscr{M}}$, the given vector is $\hat{\mathscr{U}}$, and the vector to be determined is $\hat{\mathscr{F}}$. Therefore, we can apply the iterative method directly, giving the calculation of (2.40) by the six-step procedure in § 2.3. Because all six steps in the calculation could be done at most with $O(N^2)$ cost, the total cost of the calculation would be $O(N^2)$. For mobility problems and for mixed problems where both mobile and fixed particles exist, the situation is a little different. However, this difference is not crucial and we can also solve those problems by the iterative method with the same six-step procedure (see

Appendix C for details). The detailed results on the number of iterations for mobility and resistance problems are given in § 4.2.

Inclusion of the lubrication correction in the original Stokesian Dynamics method can be treated as follows. The resistance matrix is approximated by the lubrication matrix $\mathscr{L}$ as $(\mathscr{M})^{-1} + \mathscr{L}$, where $\mathscr{M}$ is the mobility matrix with a certain truncation as in FTS. The lubrication matrix $\mathscr{L}$ could be constructed by the two-body exact solution of Jeffrey & Onishi (1984). By this approximation for the resistance matrix, we have

$$\{(\mathscr{M})^{-1} + \mathscr{L}\} \cdot \mathscr{U} = \mathscr{F}. \tag{3.2}$$

Multiplying by $\mathscr{M}$ on both sides, we obtain the inverse-free equation

$$(\mathscr{I} + \mathscr{M} \cdot \mathscr{L}) \cdot \mathscr{U} = \mathscr{M} \cdot \mathscr{F}. \tag{3.3}$$

This is a modification of the generalized mobility problem (2.19) and a generalized linear set of equations for $\mathscr{U}$ and $\mathscr{F}$. The treatment of the generalized linear set of equations is shown in Appendix C, where we need two types of dot-products – $\mathscr{M} \cdot \mathscr{F}$ and $\mathscr{L} \cdot \mathscr{U}$ for arbitrary moments $\mathscr{U}$ and $\mathscr{F}$. We can utilize the six-step procedure for the former and we could calculate the latter with an $O(N)$ cost because of the short-range nature of $\mathscr{L}$. We note that the breakdown of the lubrication correction in the Stokesian Dynamics method and the empirical prescription to overcome the breakdown were recently reported by Cichocki, Ekiel-Jeżewska & Wajnryb (1999). Because this prescription is only on the lubrication matrix $\mathscr{L}$, their correction could be applied to the current formulation: We do not go into a detailed discussion here.

We next comment briefly on the variety of iterative methods. The generalized minimum residual method (GMRES) by Saad & Shultz (1986) is widely used, but it works well only for symmetric matrices. In our problems, the bare matrix $\hat{\mathscr{M}}$ for higher versions could be slightly non-symmetric because of the reduction. Even in the FTS version, if we use the vectors with 11 elements for each particle as

$$^t(U_x, U_y, U_z, \Omega_x, \Omega_y, \Omega_z, E_{xx}, E_{xy}, E_{xz}, E_{yy}, E_{yz}) \tag{3.4}$$

and

$$^t(F_x, F_y, F_z, T_x, T_y, T_z, S_{xx}, S_{xy}, S_{xz}, S_{yy}, S_{yz}), \tag{3.5}$$

the mobility matrix in the original Stokesian Dynamics method also becomes non-symmetric. While the coefficient matrix is $\hat{\mathscr{M}}$ for resistance problems, it is a certain composition of the sub-matrices of $\hat{\mathscr{M}}$ for mobility problems (see (C 2) in Appendix C). For non-symmetric but definite matrices, the method called GPBi-CG by Zhang (1997) which is a variant of the bi-conjugate gradients stabilized method (BiCGSTAB) by van der Vorst (1992) would be suitable. We utilize the method in this paper. We do not go into the details of iterative methods further, but refer to a textbook by Weiss (1996) where various methods are described.

Before proceeding, we examine the cost of calculation in the six-step procedure to see where the next bottleneck is. All steps except for (ii) are a calculation for each particle, and the cost is $O(N)$. On the other hand, the calculation of the step (ii) contains the summation for $N - 1$ particles, so that the cost is $O(N^2)$, which is the current bottleneck.

3.2. *Fast multipole method – $O(N)$ scheme*

In this section, we discuss a further improvement using the fast multiple method (FMM). FMM was originally developed by Greengard & Rokhlin (1987) for Laplace

problems in two and three dimensions with a non-adaptive cell structure, and was then extended to the adaptive one by Carrier, Greengard & Rokhlin (1988). The application to low-Reynolds-number flows is given by Sangani & Mo (1996) for periodic boundary conditions.

While the practical aim of this formulation is to construct an $O(N)$ scheme for a finite number of particles in an unbounded fluid, we would like to implement the FMM in a simple way; we reformulate it by extending the previous multipole expansion for a particle to the expansion for a group of particles using the velocity derivatives. This is the difference of this formulation from both the original FMM and Sangani & Mo's formulation; while they use spherical harmonics, we use Cartesian moments and Cartesian derivatives which consist only of algebraic manipulations. We only consider the non-adaptive scheme in this paper.

3.2.1. *Procedure for the FMM*

The next step to improve the $O(N^2)$ scheme is in the calculation of step (ii), that is, (2.42), or its abbreviated form

$$\mathscr{V}' = \mathscr{K} \cdot \mathscr{F}, \tag{3.6}$$

which contains $(N-1)$ summation for N particles. We note that in this calculation the force moment $\mathscr{F}$ is given as a trial value in the iterative method. The point of the FMM is that we treat particles as a group both for the β in the force moments $\mathscr{F}(\beta)$ and for the α in the velocity derivatives $\mathscr{V}'(\alpha)$, rather than treat particles individually.

We introduce a hierarchical cell structure and formulate the calculations between the levels of the cell structure. The primary cell at level 0 contains all the particles. At the next level 1, we divide the primary cell into 2^3 cells called 'children'. The division is repeated up to the maximum level l_m, where the cells are called 'leaves'. All cells except for leaves have eight children and all cells except for the primary cell have their 'parent' cell. In this cell structure, the procedure of the FMM has two stages – upward-pass and downward-pass. In the upward-pass, we calculate the force moments from all particles in a cell C

$$\mathscr{F}(C) = \sum_{\beta \in C} \mathscr{S}_F(\boldsymbol{x}_C, \boldsymbol{x}_\beta) \cdot \mathscr{F}(\beta), \tag{3.7}$$

for all cells in all levels where $\boldsymbol{x}_C$ is the centre of cell C. The operator $\mathscr{S}_F(\boldsymbol{x}_2, \boldsymbol{x}_1)$ transforms the origin of a force moment from $\boldsymbol{x}_1$ to $\boldsymbol{x}_2$. From the definition of force moments, $\mathscr{S}_F$ is obtained by the binomial theorem (see Appendix D.1). In the downward-pass, we calculate the velocity derivatives in a recursive way. For this purpose, we define the operator $\mathscr{S}_V(\boldsymbol{x}_2, \boldsymbol{x}_1)$ which transforms the position of the velocity derivatives from $\boldsymbol{x}_1$ to $\boldsymbol{x}_2$. The derivation is straightforward, because the derivatives at $\boldsymbol{x}_1$ are the coefficients of the Taylor expansion at $\boldsymbol{x}_1$. The detailed derivation and the explicit form of $\mathscr{S}_V$ are given in Appendix D.2. We also introduce the velocity derivatives of the contributions from the 'well-separated' (i.e. not adjacent) cells of C and C's 'ancestors' defined by

$$\mathscr{W}(C) = \sum_{\beta \notin N^C} \mathscr{K}(C, \beta) \cdot \mathscr{F}(\beta), \tag{3.8}$$

where N^C is the nearest cell to cell C to maintain a certain accuracy for the expansion.

The procedure for the FMM consists of the following five steps:

(i) Calculate the force moment for leaves L directly by the definition (3.7) as

$$\mathscr{F}(L) = \sum_{\beta \in L} \mathscr{S}_F(\boldsymbol{x}_L, \boldsymbol{x}_\beta) \cdot \mathscr{F}(\beta), \tag{3.9}$$

where $\mathscr{F}(\beta)$ is a known variable for all particles β.

(ii) Calculate the force moment of a cell P at a level l from its children C at the level $l+1$ as

$$\mathscr{F}(P) = \sum_{C}^{8} \mathscr{S}_F(\boldsymbol{x}_P, \boldsymbol{x}_C) \cdot \mathscr{F}(C), \tag{3.10}$$

where the summation is taken for cells whose parent is P. By this recursive relation, we can calculate all force moments at the levels from $l_m - 1$ to 2.

(iii) Set all $\mathscr{W}$ at the level 1 (at least) to be zero, because the cells at that level have no well-separated cell.

(iv) Calculate $\mathscr{W}(C)$ from the parent's $\mathscr{W}(P)$ as

$$\mathscr{W}(C) = \mathscr{S}_V(\boldsymbol{x}^C, \boldsymbol{x}^P) \cdot \mathscr{W}(P) + \sum_{W^C} \mathscr{K}(C, W) \cdot \mathscr{F}(W), \tag{3.11}$$

where W^C is a cell which is not N^C on the same level as C, and whose parent is N^P; the second term in the right-hand side is the contribution not included in $\mathscr{W}(P)$. Figure 3 shows the situation in the two-dimensional case for simplicity. By this relation, we can calculate $\mathscr{W}$ for all cells at the levels from 2 to l_m.

(v) Add the contribution from the particles in the near cells, and obtain the velocity derivatives for particle α as

$$\mathscr{V}'(\alpha) = \mathscr{S}_V(\boldsymbol{x}^\alpha, \boldsymbol{x}^L) \cdot \mathscr{W}(L) + \sum_{\beta \in N^L, \beta \neq \alpha} \mathscr{K}(\alpha, \beta) \cdot \mathscr{F}(\beta). \tag{3.12}$$

This five-step procedure replaces the direct calculation of (3.6) and gives an $O(N)$ scheme.

We note that the transformation of $\mathscr{F}$ by $\mathscr{S}_F$ is exact, and there is no approximation at steps (i) and (ii). Therefore, the calculated values in the upward-pass are exactly the same as those from the definition (2.4) for the cells, in principle; this gives a good test of the programs.

3.2.2. *Cost estimation*

We estimate the calculation cost of the above non-adaptive FMM scheme. Giving force moments for all particles as a trial value in the iteration, we can calculate the velocity derivatives with the following cost for each step:

(i) calculations of (3.9) for leaves L are $O(N)$;

(ii) calculations of (3.10) for cells at the levels from $l_m - 1$ to 2 are $O(8n_C)$, where n_C is the number of all cells in the hierarchy;

(iii) clearing of $\mathscr{W}$ at level 1 is $O(1)$;

(iv) calculations of (3.11) at the levels from $l = 2$ to l_m are $O(n_C(1 + n_W))$, where n_W is the number of well-separated cells for a cell.

(v) calculations of (3.12) for all particles are $O(N(1 + n_L))$, where n_L is the number of particles in a leaf cell.

The cost of step (iii) is negligible for large N. The number of well-separated cells n_W is constant with N. The number of all cells n_C and the number of particles in a leaf

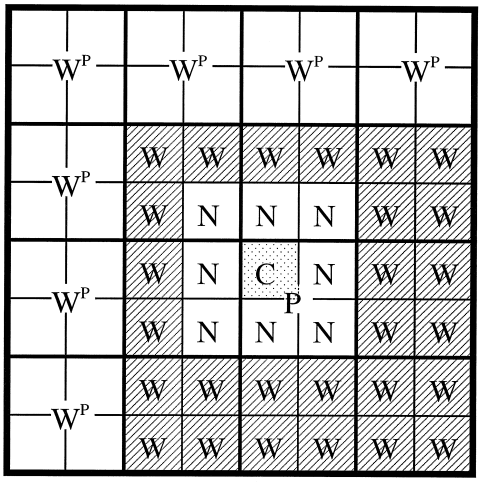

FIGURE 3. Cell structure in two dimensions. Well-separated cells from cell C are denoted by W and nearest cells to C including itself are denoted as N. Well-separated cells from C's parent cell P are denoted W^P whose children are not W.

cell n_L are given as

$$n_C = \sum_{l=0}^{l_m} 8^l = \frac{8^{l_m+1}-1}{7}, \tag{3.13}$$

$$n_L \approx \frac{N}{8^{l_m}}, \tag{3.14}$$

where we expect that the configuration is homogeneous in the primary cell. With l_m fixed, the cost of step (v) scaled by N^2 dominates for large N. On the other hand, if we choose l_m as

$$l_m \approx \log N, \tag{3.15}$$

it is expected that n_C is $O(N)$ and n_L is $O(1)$. Therefore, the costs of steps (ii), (iv) and (v) are scaled by N, and we could calculate $\mathscr{V}'$ for all particles from given force moments $\mathscr{F}$ with the cost of $O(N)$.

3.2.3. *Truncations*

The expansion of the surface force density at the particle centre discussed in § 2 and that of the force moments of particles at the centre of a group in the FMM are independent. As shown in § 2.5, the truncation error at the order p is $O(r^{-2(p+2)})$. This means that the accuracy is better for larger r, and the maximum error occurs on the pair with the smallest separation.

The expansion in the FMM requires a certain condition to obtain good estimations. For this purpose, we have introduced N^C in (3.8) and have defined the cell W^C in (3.11) by N^C as follows: W^C is at the same level as C; W^C's parent is N^P; W^C is not N^C. These definitions ensure that the well-separated cells of C, those of C's ancestors, and the nearest cell to C completely cover the whole region of the primary cell without overlaps. The typical definition of N^C is the nearest-neighbour cells including cell C itself, and there are 3^3 cells at most. Defining n_s as the number of cells between it and the nearest well-separated cell, we can denote this situation as $n_s = 1$. We note that this is not the only choice. For example, we can define N^C as the cells inside the

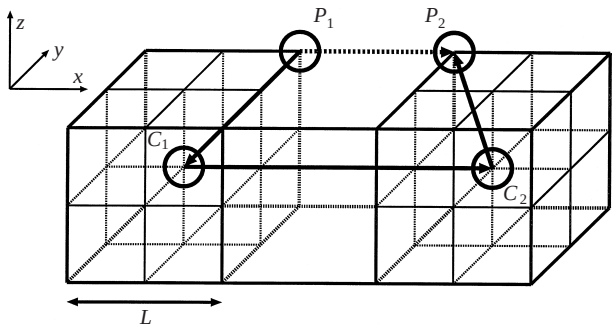

FIGURE 4. The configuration of points for the worst case in the cell structure in three dimensions for $n_s = 1$. Points C_1 and C_2 are the centres of the cell with length L and they are well-separated from each other. Points P_1 and P_2 are the farthest points in cells C_1 and C_2 respectively. The ratio of the distance between P_1 and C_1 and that between C_1 and C_2 is $\sqrt{3}/2(n_s + 1)$.

cubic region centred on C with 5 times the size of C ($n_s = 2$), where N^C would be 5^3 cells and the results in this situation would be more accurate.

Errors on the multipole expansion are characterized by the order of the truncation and the ratio r/R, where r is the distance between the source and the expansion point, and R is that between the expansion point and the observation point. In the FMM, the ratio is controlled by n_s as

$$\frac{r}{R} \leqslant \frac{\sqrt{3}}{2(n_s + 1)}. \tag{3.16}$$

The worst situation for $n_s = 1$ is shown in figure 4. If we truncate the force moments in (3.8) at the order q, $\mathscr{K}^{(n,m)}$ for $n + m > q$ would be negligible, and the maximum error would be scaled by $(\sqrt{3}/2(n_s + 1))^{q+1}$. Therefore, the truncation error in the FMM is independent of the smallest separation among particles.

To estimate the empirical error in the current scheme, we solve the two-body resistance problem where the expansion points are selected as in the worst situation in figure 4. We estimate the relative error in the scalar function X_{11}^A defined as

$$\left| \frac{[X_{11}^A]^{FMM}(r; p, q, n_s) - X_{11}^A(r; \infty)}{X_{11}^A(r; \infty) - 1} \right|, \tag{3.17}$$

where $[X_{11}^A]^{FMM}(r; p, q, n_s)$ is the result by the $O(N)$ scheme. The unity in the denominator is the single-body contribution. Figure 5 shows the relative errors for $p = 1$. For small r the error in $X_{11}^A(r; p = 1)$ dominates, and for large r the error of the FMM dominates which is independent of r. Figure 6 shows the scaling of the error at $r = 100$ with $(n_s + 1)^{-(q+1)}$ for various parameters including $p = 2, \ldots, 5$.

The separation of the crossover between these two regimes increases as q and n_s increase. For the standard choice of parameters of $n_s = 1$ and $q = 2(p + 2)$, the crossover occurs around $r = 4$ for $p = 1$; this means that the $O(N)$ scheme gives a solution equivalent to the $O(N^2)$ scheme for relatively dense configurations where the minimum separation of particles is smaller than $r = 4$. On the other hand, the $O(N)$ scheme gives a less accurate solution than the $O(N^2)$ scheme for dilute configurations, because the $O(N^2)$ scheme becomes very accurate there.

We note that the expansion in § 2 is conceptually different from the expansion in the FMM. In (2.42) we take into account $\mathscr{K}^{(n,m)}$ for $n \leqslant p + 2$ and $m \leqslant p + 2$, that

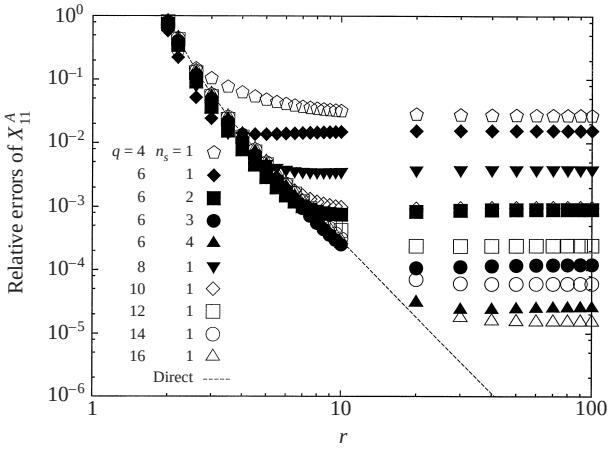

FIGURE 5. Relative errors of X^A_{11} for $p = 1$ are plotted against the separation r. The broken line denoted 'Direct' shows the relative error of $X^A_{11}(r; p = 1)$ in (2.56) which is r^{-4} for large r.

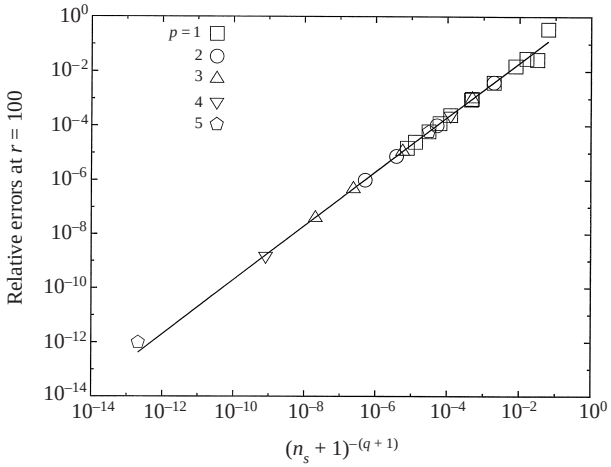

FIGURE 6. Relative errors of X^A_{11} are plotted against scaling factor $(n_s + 1)^{-(q+1)}$. The solid line shows the scaling as a reference.

is, up to $\mathscr{K}^{(p+2,p+2)}$. This is because the resultant mobility problem (2.40) must be well-defined.

4. Results

We put into practice the $O(N^2)$ and the $O(N)$ schemes and check the performance. The calculations are done on a personal computer running the FreeBSD operating system on dual Pentium III processors of 550 MHz with 1 GB memory. The programs are compiled by the GNU C compiler optimized for Pentium processors.

4.1. *Benchmarks*

We started, not by solving the physical problems such as those of mobility (2.51) or resistance (2.53), but, rather, by calculating (2.40) giving all elements of force moments. Figure 7 shows the CPU times of the calculation for $p = 1$. The result denoted by $O(N^2)$ uses the six-step iterative procedure in § 2.4, and the results denoted

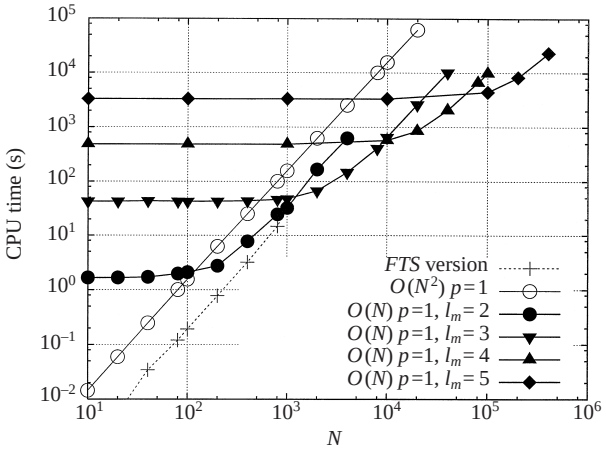

FIGURE 7. CPU times for a single calculation of (2.40) are shown against the number of particles N. The truncation order in these calculations is $p = 1$, equivalent to the FTS version. The truncation order on the FMM is $q = 2(p + 2)$, the number of spacing cells is $n_s = 1$, and the maximum levels are $l_m = 2, 3, 4$ and 5. 'FTS' denotes the original Stokesian Dynamics for a comparison.

by $O(N)$ use the five-step FMM procedure with equations (3.9) to (3.12) for step (ii) of the iterative procedure. We see that the CPU time of the $O(N^2)$ scheme is scaled by N^2. The result denoted by 'FTS' is the calculation with the explicit form of the mobility matrix given in Durlofsky *et al.* (1987). The generalization of the truncation p in the $O(N^2)$ scheme adds an extra cost. The reason for the N^2 scaling on the 'FTS' scheme is that we did not solve the physical problems nor we did not invert the matrix. For $O(N)$ schemes with a fixed l_m, we see two regions where the CPU time is almost constant with N and where it is almost scaled by N^2. The crossover occurs where the direct particle-to-particle calculation in step (v) for near cells with an $O(N^2)$ cost dominates the calculation for cells with an $O(N^0)$ cost. As suggested in § 3.2.2, we need to divide the system into finer cells for larger N in the way of (3.15). In fact, figure 7 shows that the crossover of CPU times for l_m occurs around $N \approx 10^{l_m}$, and the envelope line for $O(N)$ schemes is almost scaled by N. A similar behaviour is also observed for higher versions ($p > 1$).

4.2. *Physical problems*

Next, we consider physical problems. Figure 8 shows the average sedimentation velocities for particles placed on a simple cubic lattice with a lattice spacing $r = 3$ with a constant force applied. In the calculations, the truncation of the FMM is $q = 2(p + 2)$ and the number of spacing cells is $n_s = 1$. The velocities are all proportional to $N^{2/3}$; if the conglomerate of particles is equivalent to a single object with the same size, the sedimentation velocity would be the ratio of the applied force scaled by N to the drag coefficient scaled by the linear dimension of the object $N^{1/3}$. The differences between the $O(N)$ and $O(N^2)$ schemes for each p are the same order as the error on the $O(N^2)$ scheme.

In these physical problems, we need to solve the linear set of equations. By the iterative method, the total calculation cost is proportional to the number of iterations n_i. For any problem, we need n_i times the CPU time for a single calculation of (2.40) shown in figure 7, because all mobility, resistance, and mixed problems consist of (2.40) as the core calculation. To observe the convergent behaviour during the

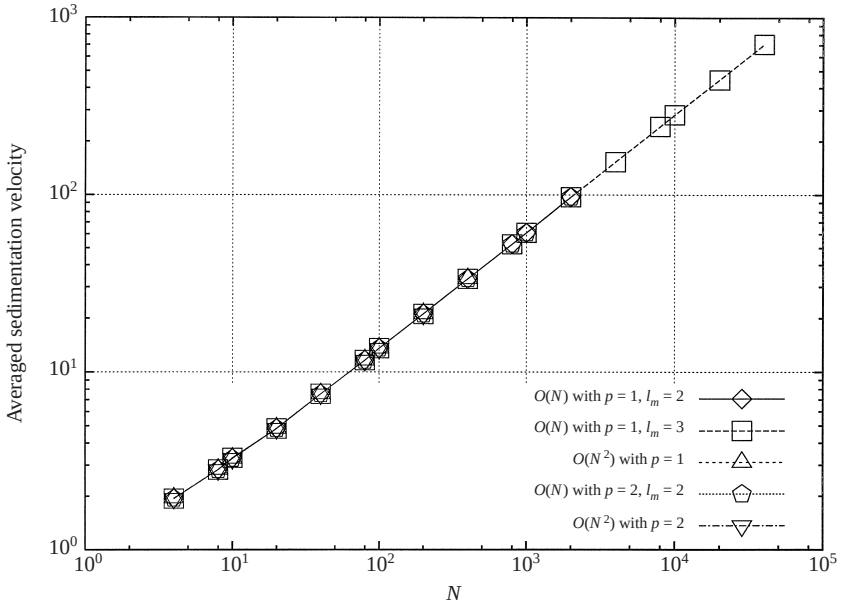

FIGURE 8. Averaged sedimentation velocities of particles placed on a simple cubic lattice with the spacing $r = 3$. The velocity is scaled by the single-particle sedimentation velocity.

Problem:	Mobility				Resistance			
Configuration:	Simple cubic		Random		Simple cubic		Random	
Separation r:	3.0	2.2	3.0	2.2	3.0	2.2	3.0	2.2
$N = 4$	3	7	4	11	5	6	19	42
$N = 8$	3	6	5	10	5	6	30	59
$N = 10$	5	12	5	14	27	65	27	67
$N = 20$	5	15	6	11	44	259	40	116
$N = 40$	5	7	6	9	43	115	56	158
$N = 80$	6	11	6	7	72	427	69	101
$N = 100$	6	12	6	7	79	795	76	131
$N = 200$	5	8	6	7	90	214	114	176
$N = 400$	5	8	6		118	319	133	
$N = 800$	5	8	6		133	380	171	
$N = 1000$	5	8	6		186	680	202	
$N = 2000$	5				180		200	
$N = 4000$	5				378		289	
$N = 8000$	5				482		412	

TABLE 2. Numbers of iterations for mobility and resistance problems of simple cubic and random configurations with separations $r = 3.0$ and 2.2 in FMM code with $p = 1$, $q = 6$, and $n_s = 1$ by the GPBi-CG method under the accuracy $\epsilon = 10^{-6}$. (Blank entries are not calculated.)

iterations, we calculate mobility and resistance problems for a simple cubic lattice and a random configuration with separations $r = 3.0$ and 2.2. The random configurations are obtained by a uniform random distribution in the cubic region with length $rN^{1/3}$ excluding the overlaps. Figure 9 shows the residual with the number of iterations in the GPBi-CG method for $N = 200$. The convergence for mobility problems are much

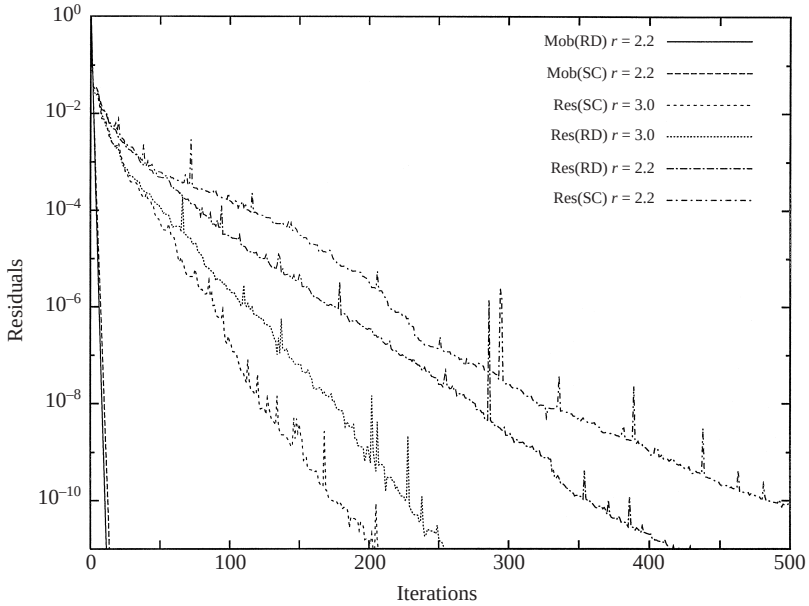

FIGURE 9. Residuals in the GPBi-CG method for $N = 200$. 'Mob' and 'Res' denote the mobility and the resistance problems, and 'SC' and 'RD' denote a simple cubic lattice and random configuration respectively.

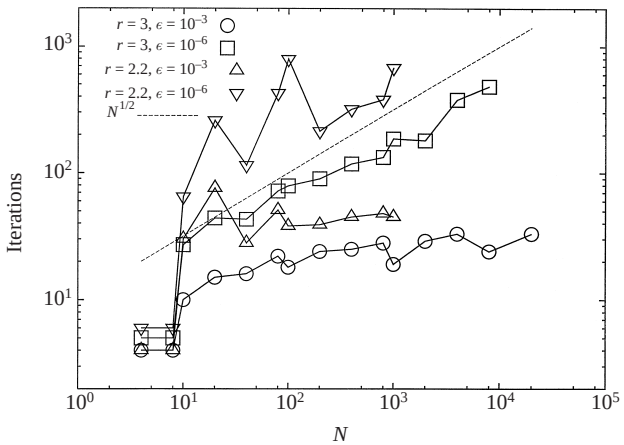

FIGURE 10. Numbers of iterations for resistance problems for simple cubic lattices with $r = 3.0$ and 2.2 for accuracies $\epsilon = 10^{-3}$ and 10^{-6}. The broken line shows $N^{1/2}$ as a reference.

faster than that for resistance problems. Table 2 shows the numbers of iterations for an accuracy of 10^{-6}. For mobility problems, the number of iterations is independent of N and the problems are solved with an $O(N)$ cost. On the other hand, for resistance problems, more iterations are necessary for larger N. Figure 10 shows the numbers of iterations for resistance problems. For high accuracy $\epsilon = 10^{-6}$ the number of iterations increases almost as $N^{1/2}$, and for low accuracy $\epsilon = 10^{-3}$ it increases like $\log N$ (or, at lease, slower than $N^{1/2}$). Therefore, the total cost of the calculation for resistance problems would be $O(N^{3/2})$.

5. Conclusions

In this paper, we have given a formulation of the hydrodynamic interactions for rigid spherical particles in an unbounded fluid under the Stokes approximation by a multipole expansion in real space. We have derived the generalized mobility problem which relates the force moments to the velocity moments with arbitrary orders. In our formulation, we calculate the velocity moments through the velocity derivatives. Faxén's law was used in the original Stokesian Dynamics method and is one of the reasons for the limitation at the FTS version which is equivalent to the truncation at the first order in our formulation. We do not utilize Faxén's law explicitly. However, the present formulation contains it implicitly; we integrate the disturbance field of the fluid velocity on the surface of particles for velocity moments, and Batchelor (1972) did the same in the classical derivation of Faxén's law. To obtain well-defined problems, the proper reduction of force and velocity moments is essential; the reduction is another barrier of the extension of the original Stokesian Dynamics method. To overcome this barrier, we formulate the scheme not directly with the physical variables such as forces, torques, stresslets, translational and rotational velocities, and rate of strains, but with the mathematical variables such as force moments and velocity moments. Because of the systematic formulation with the proper reduction, there is no limitation to the FTS, and extensions to higher orders are straightforward. By this formulation, one of the difficulties in the original Stokesian Dynamics method has been overcome and the results have been shown up to order $p = 7$ explicitly.

We have also obtained an improvement in the calculation speed, which is the other difficulty in the original Stokesian Dynamics method. By the application of an iterative method to solve the linear set of equations, we have formulated an $O(N^2)$ scheme. Because we calculate the velocity moments through the velocity derivatives, we can formulate the fast multipole method (FMM) in a simple way as a natural extension of the conventional multipole expansion. We have obtained an $O(N)$ scheme by the non-adaptive FMM. Our formulation of the FMM has no harmonics nor trigonometric functions; this is the difference (and would be the advantage) from the original formulations by Greengard & Rokhlin (1987) and the application to Stokes flows by Sangani & Mo (1996).

The performance of these schemes has been tested for a single calculation of the generalized mobility problem, which appears in each iteration to solve the mobility and resistance problems. The CPU time for this calculation is either $O(N^2)$ or $O(N)$ where we choose the levels of the non-adaptive cell structure as $\log N$. Real problems have been calculated by these schemes for particles $N = 400\,000$ at most. We have found that mobility problems need small and constant iterations and the total calculation costs are $O(N)$. On the other hand, resistance problems need more iterations and total calculation costs seem to be scaled by $N^{3/2}$ with a high accuracy $\epsilon = 10^{-6}$; using the $O(N^2)$ scheme, the total cost would be $O(N^{5/2})$ and the original Stokesian Dynamics method requires a cost of $O(N^3)$. It should be noted that we did not apply any preconditioning techniques on the iterative method and that would be helpful for this situation.

Using the $O(N)$ scheme, we can examine the detailed hydrodynamic interactions for a huge agglomerate of particles in a fluid. Fortunately, many interesting phenomena such as the breakup of falling clusters by Nitsch & Batchelor (1997) and dispersion by shear flows by Kao & Mason (1975) are mobility problems, and the plain $O(N)$ scheme is sufficient.

We avoid discussing the lubrication correction in this paper because of lack of theoretical justification. Recently, Cichocki *et al.* (1999) showed a failure of the original lubrication correction on the three-body problems and gave a modification without theoretical justification. The present scheme with higher moments would give the correct interactions not only for two particles but also for three and more particles, and could give the theoretical background and the correct treatments.

The adaptive version of the FMM was also not discussed in this paper. However, this would be important for practical studies of the dynamics, because we usually observe structures or patterns in the systems and the non-adaptive FMM does not handle such situations well.

The current scheme would be easily applicable to other problems: for the problems with rigid but non-spherical objects, we need to construct proper moment-reduction procedures for the geometries; for non-rigid objects, we need to include the double-layer potential in the integral equation (2.1); and for systems with the periodic boundary condition, we need to replace the Green function from the Oseen tensor by the tensor with the Ewald summation (Beenakker 1986 and Brady *et al.* 1988). In addition to these hydrodynamic problems, we could extend this formulation in a straightforward way to problems governed by linear equations such as Laplace problems, linear elastic problems, gravitational systems, and vortex dynamics.

The author wishes to thank Professor John F. Brady for guiding him to this work and helpful discussions and comments on the manuscript. This work was supported by the Japan Society for the Promotion of Science.

Appendix A. Explicit form of the self-part of the mobility matrix

We discuss the properties of the self-part of the generalized mobility matrix defined by (2.22). First we prove two properties discussed in § 2.2:

(i) $\mathscr{M}^{s(n,m)}$ is non-zero only when n and m are both odd or both even;

(ii) $\mathscr{M}^{s(n,m)}$ is zero for $m \geqslant n+2$.

The derivatives of the Oseen tensor $\mathscr{J}^{(m)}$ have the scalar part proportional to $1/r^{m+1}$ and the tensor part a linear combination of $\hat{\boldsymbol{r}}^n$ with $n = m+2, m, m-2, \ldots, 1$ or 0. The surface integral of unit tensor $\hat{\boldsymbol{r}}^n$ has the following non-zero value:

$$\frac{1}{4\pi}\int_{|\hat{\boldsymbol{r}}|=1} \mathrm{d}S(\hat{\boldsymbol{r}})\hat{r}^n_{k\ldots} = \frac{(n_x-1)!!(n_y-1)!!(n_z-1)!!}{(n+1)!!} \tag{A 1}$$

only if n, n_x, n_y, and n_z are all even, where n_x, n_y, n_z are the number of indices of x, y, and z in $k\ldots$ respectively. This is a simple extension of Kronecker's delta to higher rank. In fact,

$$\frac{1}{4\pi}\int_{|\hat{\boldsymbol{r}}|=1} \mathrm{d}S(\hat{\boldsymbol{r}})\hat{r}_i\hat{r}_j = \frac{1}{3!!}\delta_{ij}, \tag{A 2}$$

$$\frac{1}{4\pi}\int_{|\hat{\boldsymbol{r}}|=1} \mathrm{d}S(\hat{\boldsymbol{r}})\hat{r}_i\hat{r}_j\hat{r}_k\hat{r}_l = \frac{1}{5!!}(\delta_{ij}\delta_{kl}+\delta_{ik}\delta_{jl}+\delta_{il}\delta_{jk}). \tag{A 3}$$

From these properties, it is found that $\mathscr{M}^{s(n,m)}$ becomes zero if m is odd and n is even or vice versa; the property (i) has been proven.

The Oseen tensor J_{ij} defined by (2.2) has the following properties:

$$\nabla_i J_{ij} = 0, \tag{A 4}$$

$$\nabla^2 J_{ii} = 0, \tag{A 5}$$

$$\nabla^2 \nabla^2 J_{ij} = 0. \tag{A 6}$$

Let us consider property (ii) on $\mathscr{M}^{s(n,m)}$ with $m \geqslant n+2$. From the properties used to prove property (i), all indices on $\mathscr{M}^{s(n,m)}_{i,l\ldots;j,k\ldots}$ must match each other. We could consider three matching cases: i and j are matched, i or j is matched with one of $k\ldots$, and i and j are both matched with $l\ldots$. In the first case, Laplacian ∇^2 would appear from the matching among $k\ldots$, and the matrix becomes zero by (A 5). In the second case, the matrix also becomes zero by (A 4). In the third case, there are at least four indices in $k\ldots$ to match each other; therefore, the operator $\nabla^2\nabla^2$ would appear and the matrix again becomes zero by (A 6). Because the possible indices are one of the above three cases, $\mathscr{M}^{s(n,m)}$ with $m \geqslant n+2$ are always zero; the property (ii) has been proven.

As a result, the self-part of the mobility matrix would have the following form:

$$\mathscr{M}^s = \begin{bmatrix} \mathscr{M}^{s(0,0)} & 0 & 0 & 0 & \cdots \\ 0 & \mathscr{M}^{s(1,1)} & 0 & 0 & \cdots \\ \mathscr{M}^{s(2,0)} & 0 & \mathscr{M}^{s(2,2)} & 0 & \cdots \\ 0 & \mathscr{M}^{s(3,1)} & 0 & \mathscr{M}^{s(3,3)} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}. \tag{A 7}$$

The explicit forms of $\mathscr{M}^{s(0,0)}$ and $\mathscr{M}^{s(1,1)}$ are given by (2.23) and (2.24) respectively.

Appendix B. Explicit form of the reduction and recovery operators

We show the explicit form of $\mathscr{P}$ and $\mathscr{Q}$ up to the third order. Although we use the force moments in the following, the results are also applicable for the velocity moments.

B.1. *Reduction operator* $\mathscr{P}$

From the properties of $\mathscr{P}$, the operator is decomposed into small parts as

$$\begin{bmatrix} \hat{\mathscr{F}}^{(0)} \\ \hat{\mathscr{F}}^{(1)} \\ \hat{\mathscr{F}}^{(2)} \\ \hat{\mathscr{F}}^{(3)} \\ \vdots \end{bmatrix} = \begin{bmatrix} \mathscr{P}^{(0,0)} & 0 & 0 & 0 & \cdots \\ 0 & \mathscr{P}^{(1,1)} & 0 & 0 & \cdots \\ \mathscr{P}^{(2,0)} & 0 & \mathscr{P}^{(2,2)} & 0 & \cdots \\ 0 & \mathscr{P}^{(3,1)} & 0 & \mathscr{P}^{(3,3)} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \cdot \begin{bmatrix} \mathscr{F}^{(0)} \\ \mathscr{F}^{(1)} \\ \mathscr{F}^{(2)} \\ \mathscr{F}^{(3)} \\ \vdots \end{bmatrix}. \tag{B 1}$$

To write down the explicit forms of the small matrices, we use a symmetric form for $\mathscr{F}$ for simplicity. Only $\mathscr{P}^{(1,1)}$ of the small matrices in (B 1) has non-diagonal elements which relate $\mathscr{F}_{i,k\ldots}$ with different i. Thus we write $\mathscr{P}^{(1,1)}$ first. The explicit form of

$\mathscr{P}^{(1,1)}$ is given as

$$\begin{bmatrix} \hat{\mathscr{F}}^{(1)}_{x,y} \\ \hat{\mathscr{F}}^{(1)}_{x,z} \\ \hline \hat{\mathscr{F}}^{(1)}_{y,x} \\ \hat{\mathscr{F}}^{(1)}_{y,y} \\ \hat{\mathscr{F}}^{(1)}_{y,z} \\ \hline \hat{\mathscr{F}}^{(1)}_{z,x} \\ \hat{\mathscr{F}}^{(1)}_{z,y} \\ \hat{\mathscr{F}}^{(1)}_{z,z} \end{bmatrix} = \mathscr{P}^{(1,1)} \cdot \mathscr{F}^{(1)} = \left[\begin{array}{ccc|ccc|ccc} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ -1/3 & 0 & 0 & 0 & 2/3 & 0 & 0 & 0 & -1/3 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ -1/3 & 0 & 0 & 0 & -1/3 & 0 & 0 & 0 & 2/3 \end{array}\right] \cdot \begin{bmatrix} \mathscr{F}^{(1)}_{x,x} \\ \mathscr{F}^{(1)}_{x,y} \\ \mathscr{F}^{(1)}_{x,z} \\ \hline \mathscr{F}^{(1)}_{y,x} \\ \mathscr{F}^{(1)}_{y,y} \\ \mathscr{F}^{(1)}_{y,z} \\ \hline \mathscr{F}^{(1)}_{z,x} \\ \mathscr{F}^{(1)}_{z,y} \\ \mathscr{F}^{(1)}_{z,z} \end{bmatrix}. \tag{B 2}$$

Because the other matrices are essentially independent of the indices i of $\mathscr{F}_{i,k\ldots}$, we write matrices only for $\mathscr{F}_{x,k\ldots}$. The diagonal matrices $\mathscr{P}^{(m,m)}$ are just the extraction of the independent elements from the $\mathscr{F}$. The explicit forms are given as

$$\mathscr{P}^{(0,0)}_{x,x} = 1, \tag{B 3}$$

$$\begin{bmatrix} \hat{\mathscr{F}}^{(2)}_{x,xy} \\ \hat{\mathscr{F}}^{(2)}_{x,xz} \\ \hline \hat{\mathscr{F}}^{(2)}_{x,yy} \\ \hat{\mathscr{F}}^{(2)}_{x,yz} \\ \hline \hat{\mathscr{F}}^{(2)}_{x,zz} \end{bmatrix} = \mathscr{P}^{(2,2)}_{x,x} \cdot \mathscr{F}^{(2)}_{x} = \left[\begin{array}{ccc|ccc} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 1 \end{array}\right] \cdot \begin{bmatrix} \mathscr{F}^{(2)}_{x,xx} \\ \mathscr{F}^{(2)}_{x,xy} \\ \mathscr{F}^{(2)}_{x,xz} \\ \hline \mathscr{F}^{(2)}_{x,yy} \\ \mathscr{F}^{(2)}_{x,yz} \\ \hline \mathscr{F}^{(2)}_{x,zz} \end{bmatrix}, \tag{B 4}$$

and

$$\begin{bmatrix} \hat{\mathscr{F}}^{(3)}_{x,xyy} \\ \hat{\mathscr{F}}^{(3)}_{x,xyz} \\ \hat{\mathscr{F}}^{(3)}_{x,xzz} \\ \hline \hat{\mathscr{F}}^{(3)}_{x,yyy} \\ \hat{\mathscr{F}}^{(3)}_{x,yyz} \\ \hat{\mathscr{F}}^{(3)}_{x,yzz} \\ \hline \hat{\mathscr{F}}^{(3)}_{x,zzz} \end{bmatrix} = \mathscr{P}^{(3,3)}_{x,x} \cdot \mathscr{F}^{(3)}_{x} = \left[\begin{array}{ccc|ccc|ccc|c} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{array}\right] \cdot \begin{bmatrix} \mathscr{F}^{(3)}_{x,xxx} \\ \mathscr{F}^{(3)}_{x,xxy} \\ \mathscr{F}^{(3)}_{x,xxz} \\ \hline \mathscr{F}^{(3)}_{x,xyy} \\ \mathscr{F}^{(3)}_{x,xyz} \\ \mathscr{F}^{(3)}_{x,xzz} \\ \hline \mathscr{F}^{(3)}_{x,yyy} \\ \mathscr{F}^{(3)}_{x,yyz} \\ \mathscr{F}^{(3)}_{x,yzz} \\ \hline \mathscr{F}^{(3)}_{x,zzz} \end{bmatrix}. \tag{B 5}$$

The off-diagonal matrices showing the subtraction of the traces appear only in the

lower-half part in (B 1). The explicit forms are given as

$$\begin{bmatrix} \hat{\mathscr{F}}^{(2)}_{x,xy} \\ \hat{\mathscr{F}}^{(2)}_{x,xz} \\ \hline \hat{\mathscr{F}}^{(2)}_{x,yy} \\ \hat{\mathscr{F}}^{(2)}_{x,yz} \\ \hline \hat{\mathscr{F}}^{(2)}_{x,zz} \end{bmatrix} = \mathscr{P}^{(2,0)}_{x,x} \cdot \mathscr{F}^{(0)}_{x} = \begin{bmatrix} 0 \\ 0 \\ \hline -1/3 \\ 0 \\ \hline -1/3 \end{bmatrix} \cdot \mathscr{F}^{(0)}_{x}, \tag{B 6}$$

and

$$\begin{bmatrix} \hat{\mathscr{F}}^{(3)}_{x,xyy} \\ \hat{\mathscr{F}}^{(3)}_{x,xyz} \\ \hat{\mathscr{F}}^{(3)}_{x,xzz} \\ \hline \hat{\mathscr{F}}^{(3)}_{x,yyy} \\ \hat{\mathscr{F}}^{(3)}_{x,yyz} \\ \hat{\mathscr{F}}^{(3)}_{x,yzz} \\ \hline \hat{\mathscr{F}}^{(3)}_{x,zzz} \end{bmatrix} = \mathscr{P}^{(3,1)}_{x,x} \cdot \mathscr{F}^{(1)}_{x} = \begin{bmatrix} -1/5 & 0 & 0 \\ 0 & 0 & 0 \\ -1/5 & 0 & 0 \\ \hline 0 & -3/5 & 0 \\ 0 & 0 & -1/5 \\ 0 & -1/5 & 0 \\ \hline 0 & 0 & -3/5 \end{bmatrix} \cdot \begin{bmatrix} \mathscr{F}^{(1)}_{x,x} \\ \mathscr{F}^{(1)}_{x,y} \\ \mathscr{F}^{(1)}_{x,z} \end{bmatrix}. \tag{B 7}$$

B.2. *Recovery operator* $\mathscr{Q}$

The recovery operator $\mathscr{Q}$ is also decomposed as

$$\begin{bmatrix} \mathscr{F}^{(0)} \\ \mathscr{F}^{(1)} \\ \mathscr{F}^{(2)} \\ \mathscr{F}^{(3)} \\ \vdots \end{bmatrix} = \begin{bmatrix} \mathscr{Q}^{(0,0)} & 0 & 0 & 0 & \cdots \\ 0 & \mathscr{Q}^{(1,1)} & 0 & 0 & \cdots \\ \mathscr{Q}^{(2,0)} & 0 & \mathscr{Q}^{(2,2)} & 0 & \cdots \\ 0 & \mathscr{Q}^{(3,1)} & 0 & \mathscr{Q}^{(3,3)} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \cdot \begin{bmatrix} \hat{\mathscr{F}}^{(0)} \\ \hat{\mathscr{F}}^{(1)} \\ \hat{\mathscr{F}}^{(2)} \\ \hat{\mathscr{F}}^{(3)} \\ \vdots \end{bmatrix}. \tag{B 8}$$

As for $\mathscr{P}$, we write $\mathscr{Q}^{(1,1)}$ first. The explicit form is given as

$$\begin{bmatrix} \mathscr{F}^{(1)}_{x,x} \\ \mathscr{F}^{(1)}_{x,y} \\ \mathscr{F}^{(1)}_{x,z} \\ \hline \mathscr{F}^{(1)}_{y,x} \\ \mathscr{F}^{(1)}_{y,y} \\ \mathscr{F}^{(1)}_{y,z} \\ \hline \mathscr{F}^{(1)}_{z,x} \\ \mathscr{F}^{(1)}_{z,y} \\ \mathscr{F}^{(1)}_{z,z} \end{bmatrix} = \mathscr{Q}^{(1,1)} \cdot \hat{\mathscr{F}}^{(1)} = \left[\begin{array}{cc|ccc|ccc} 0 & 0 & 0 & -1 & 0 & 0 & 0 & -1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{array}\right] \cdot \begin{bmatrix} \hat{\mathscr{F}}^{(1)}_{x,y} \\ \hat{\mathscr{F}}^{(1)}_{x,z} \\ \hline \hat{\mathscr{F}}^{(1)}_{y,x} \\ \hat{\mathscr{F}}^{(1)}_{y,y} \\ \hat{\mathscr{F}}^{(1)}_{y,z} \\ \hline \hat{\mathscr{F}}^{(1)}_{z,x} \\ \hat{\mathscr{F}}^{(1)}_{z,y} \\ \hat{\mathscr{F}}^{(1)}_{z,z} \end{bmatrix}. \tag{B 9}$$

The diagonal matrices $\mathcal{Q}^{(m,m)}$ are given as

$$\mathcal{Q}^{(0,0)}_{x,x} = 1, \tag{B 10}$$

$$\begin{bmatrix} \mathscr{F}^{(2)}_{x,xx} \\ \mathscr{F}^{(2)}_{x,xy} \\ \mathscr{F}^{(2)}_{x,xz} \\ \hline \mathscr{F}^{(2)}_{x,yy} \\ \mathscr{F}^{(2)}_{x,yz} \\ \hline \mathscr{F}^{(2)}_{x,zz} \end{bmatrix} = \mathcal{Q}^{(2,2)}_{x,x} \cdot \hat{\mathscr{F}}^{(2)}_{x} = \left[\begin{array}{cc|ccc} 0 & 0 & -1 & 0 & -1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ \hline 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ \hline 0 & 0 & 0 & 0 & 1 \end{array}\right] \cdot \begin{bmatrix} \hat{\mathscr{F}}^{(2)}_{x,xy} \\ \hat{\mathscr{F}}^{(2)}_{x,xz} \\ \hline \hat{\mathscr{F}}^{(2)}_{x,yy} \\ \hat{\mathscr{F}}^{(2)}_{x,yz} \\ \hline \hat{\mathscr{F}}^{(2)}_{x,zz} \end{bmatrix}, \tag{B 11}$$

and

$$\begin{bmatrix} \mathscr{F}^{(3)}_{x,xxx} \\ \mathscr{F}^{(3)}_{x,xxy} \\ \mathscr{F}^{(3)}_{x,xxz} \\ \hline \mathscr{F}^{(3)}_{x,xyy} \\ \mathscr{F}^{(3)}_{x,xyz} \\ \mathscr{F}^{(3)}_{x,xzz} \\ \hline \mathscr{F}^{(3)}_{x,yyy} \\ \mathscr{F}^{(3)}_{x,yyz} \\ \mathscr{F}^{(3)}_{x,yzz} \\ \hline \mathscr{F}^{(3)}_{x,zzz} \end{bmatrix} = \mathcal{Q}^{(3,3)}_{x,x} \cdot \hat{\mathscr{F}}^{(3)}_{x} = \left[\begin{array}{ccc|ccc|c} -1 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & -1 \\ \hline 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{array}\right] \cdot \begin{bmatrix} \hat{\mathscr{F}}^{(3)}_{x,xyy} \\ \hat{\mathscr{F}}^{(3)}_{x,xyz} \\ \hat{\mathscr{F}}^{(3)}_{x,xzz} \\ \hline \hat{\mathscr{F}}^{(3)}_{x,yyy} \\ \hat{\mathscr{F}}^{(3)}_{x,yyz} \\ \hat{\mathscr{F}}^{(3)}_{x,yzz} \\ \hline \hat{\mathscr{F}}^{(3)}_{x,zzz} \end{bmatrix}. \tag{B 12}$$

The off-diagonal matrices are given as

$$\begin{bmatrix} \mathscr{F}^{(2)}_{x,xx} \\ \mathscr{F}^{(2)}_{x,xy} \\ \mathscr{F}^{(2)}_{x,xz} \\ \hline \mathscr{F}^{(2)}_{x,yy} \\ \mathscr{F}^{(2)}_{x,yz} \\ \hline \mathscr{F}^{(2)}_{x,zz} \end{bmatrix} = \mathcal{Q}^{(2,0)}_{x,x} \cdot \hat{\mathscr{F}}^{(0)}_{x} = \begin{bmatrix} 1/3 \\ 0 \\ 0 \\ \hline 1/3 \\ 0 \\ \hline 1/3 \end{bmatrix} \cdot \hat{\mathscr{F}}^{(0)}_{x}, \tag{B 13}$$

and

$$\left[\begin{array}{c} \mathscr{F}^{(3)}_{x,xxx} \\ \mathscr{F}^{(3)}_{x,xxy} \\ \mathscr{F}^{(3)}_{x,xxz} \\ \hline \mathscr{F}^{(3)}_{x,xyy} \\ \mathscr{F}^{(3)}_{x,xyz} \\ \mathscr{F}^{(3)}_{x,xzz} \\ \hline \mathscr{F}^{(3)}_{x,yyy} \\ \mathscr{F}^{(3)}_{x,yyz} \\ \mathscr{F}^{(3)}_{x,yzz} \\ \hline \mathscr{F}^{(3)}_{x,zzz} \end{array}\right] = \mathscr{Q}^{(3,1)}_{x,x} \cdot \hat{\mathscr{F}}^{(1*)}_{x} = \left[\begin{array}{ccc} 3/5 & 0 & 0 \\ 0 & 1/5 & 0 \\ 0 & 0 & 1/5 \\ \hline 1/5 & 0 & 0 \\ 0 & 0 & 0 \\ 1/5 & 0 & 0 \\ \hline 0 & 3/5 & 0 \\ 0 & 0 & 1/5 \\ 0 & 1/5 & 0 \\ \hline 0 & 0 & 3/5 \end{array}\right] \cdot \left[\begin{array}{c} \hat{\mathscr{F}}^{(1*)}_{x,x} \\ \hat{\mathscr{F}}^{(1)}_{x,y} \\ \hat{\mathscr{F}}^{(1)}_{x,z} \end{array}\right], \tag{B 14}$$

where $\hat{\mathscr{F}}^{(1*)}_{x,x} = \hat{\mathscr{F}}^{(1)}_{x,y} - \hat{\mathscr{F}}^{(1)}_{x,z}$ for simplicity.

Appendix C. General remarks on the iterative method

We usually meet the situation for a linear set of equations (3.1) where some elements of the left-hand-side vector $\boldsymbol{b}$ are unknown and some elements of a vector in the right-hand side $\boldsymbol{x}$ are given. To clarify the situation, we explicitly write the equation as

$$\begin{pmatrix} \boldsymbol{b} \\ \boldsymbol{c} \end{pmatrix} = \boldsymbol{A} \cdot \begin{pmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{pmatrix}, \tag{C 1}$$

where $\boldsymbol{b}$ and $\boldsymbol{y}$ are given and $\boldsymbol{x}$ and $\boldsymbol{c}$ are unknown. The dimensions of $\boldsymbol{x}$ and $\boldsymbol{b}$ and those of $\boldsymbol{y}$ and $\boldsymbol{c}$ are the same respectively. We can transform (C 1) to the form (3.1) by moving the given variables to the left-hand side and the unknown variables to the right-hand side as

$$\begin{pmatrix} \boldsymbol{b} \\ \boldsymbol{0} \end{pmatrix} - \boldsymbol{A} \cdot \begin{pmatrix} \boldsymbol{0} \\ \boldsymbol{y} \end{pmatrix} = -\begin{pmatrix} \boldsymbol{0} \\ \boldsymbol{c} \end{pmatrix} + \boldsymbol{A} \cdot \begin{pmatrix} \boldsymbol{x} \\ \boldsymbol{0} \end{pmatrix}. \tag{C 2}$$

To solve this problem, first calculate the left-hand-side vector with the dot-product calculation once, then implement the routine of an iterative method using the subroutine to calculate the right-hand side from the input vector ${}^{t}(\boldsymbol{c}, \boldsymbol{x})$, which would be constructed by the subroutine to calculate $\boldsymbol{A} \cdot {}^{t}(\boldsymbol{x}, \boldsymbol{y})$ giving $\boldsymbol{y} = \boldsymbol{0}$ explicitly.

For the generalized linear set of equations which we meet in the original Stokesian Dynamics method with the lubrication correction, the above treatment is also applicable. Let us consider the problem where matrices appear on both sides of the equation as

$$\boldsymbol{B} \cdot \begin{pmatrix} \boldsymbol{b} \\ \boldsymbol{c} \end{pmatrix} = \boldsymbol{A} \cdot \begin{pmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{pmatrix}, \tag{C 3}$$

where $\boldsymbol{b}$ and $\boldsymbol{y}$ are given and $\boldsymbol{x}$ and $\boldsymbol{c}$ are unknown as in (C 1). Again we can transform (C 3) to

$$\boldsymbol{B} \cdot \begin{pmatrix} \boldsymbol{b} \\ \boldsymbol{0} \end{pmatrix} - \boldsymbol{A} \cdot \begin{pmatrix} \boldsymbol{0} \\ \boldsymbol{y} \end{pmatrix} = -\boldsymbol{B} \cdot \begin{pmatrix} \boldsymbol{0} \\ \boldsymbol{c} \end{pmatrix} + \boldsymbol{A} \cdot \begin{pmatrix} \boldsymbol{x} \\ \boldsymbol{0} \end{pmatrix}. \tag{C 4}$$

This is the same form as (3.1); the left-hand side consists of the given values only and the right-hand side can be summarized as the matrix-vector product with the unknown vector. Although the operation of the dot-product increases, the generalized linear set of equations is also solved without decomposing the matrices $\boldsymbol{A}$ and $\boldsymbol{B}$.

Appendix D. The explicit form of the shift operators

D.1. *Force moments*

We describe the transformations of the origin of moments. We would like to represent the moments with origin $\boldsymbol{x}_2$ by $\mathscr{F}(\boldsymbol{x}_1)$. From the definition, the moments with $\boldsymbol{x}_2$ are given by

$$\mathscr{F}^{(m)}_{i,k\ldots}(\boldsymbol{x}_2) = \int \mathrm{d}S(\boldsymbol{y})\ (\boldsymbol{y}-\boldsymbol{x}_2)^m_{k\ldots} f_i(\boldsymbol{y}). \tag{D 1}$$

Here $(\boldsymbol{y}-\boldsymbol{x}_2)^m$ could be written by the linear combination of $(\boldsymbol{y}-\boldsymbol{x}_1)^i$ and $(\boldsymbol{x}_1-\boldsymbol{x}_2)^j$ where $i+j=m$. By the 'binomial theorem for three-dimensional vectors', we are able to transform $\mathscr{F}(\boldsymbol{x}_1)$ to $\mathscr{F}(\boldsymbol{x}_2)$ uniquely. The importance of the binomial theorem for vectors is their non-commutable property,

$$a_i b_j \neq b_i a_j. \tag{D 2}$$

Let us consider the expansion

$$[(\boldsymbol{a}+\boldsymbol{b})^n]_{k\ldots}\ . \tag{D 3}$$

If we know the n-indices '$k\ldots$', that is, the numbers of x, y, and z are n_x, n_y, and n_z respectively where $n_x+n_y+n_z=n$, using the usual (scalar) binomial theorem

$$(a+b)^n = \sum_{i=0}^{n} {}_nC_i a^i b^{(n-i)}, \tag{D 4}$$

we obtain the expansion as

$$[(\boldsymbol{a}+\boldsymbol{b})^n]_{k\ldots} = \sum_{i_x=0}^{n_x}\sum_{i_y=0}^{n_y}\sum_{i_z=0}^{n_z} {}_{n_x}C_{i_x}\,{}_{n_y}C_{i_y}\,{}_{n_z}C_{i_z}(a_x)^{i_x}(a_y)^{i_y}(a_z)^{i_z}(b_x)^{j_x}(b_y)^{j_y}(b_z)^{j_z}, \tag{D 5}$$

where $j_x=n_x-i_x$, $j_y=n_y-i_y$, and $j_z=n_z-i_z$ respectively. The expansion of the power of the sum of two vectors $\boldsymbol{a}$ and $\boldsymbol{b}$ is straightforward. We just write it down explicitly for lower powers:

$$\mathscr{F}^{(0)}_i(\boldsymbol{x}_2) = \mathscr{F}^{(0)}_i(\boldsymbol{x}_1), \tag{D 6}$$

$$\mathscr{F}^{(1)}_{i,j}(\boldsymbol{x}_2) = r_j\mathscr{F}^{(0)}_i(\boldsymbol{x}_1) + \mathscr{F}^{(1)}_{i,j}(\boldsymbol{x}_1), \tag{D 7}$$

$$\mathscr{F}^{(2)}_{i,jk}(\boldsymbol{x}_2) = r^2_{jk}\mathscr{F}^{(0)}_i(\boldsymbol{x}_1) + r_j\mathscr{F}^{(1)}_{i,k}(\boldsymbol{x}_1) + r_k\mathscr{F}^{(1)}_{i,j}(\boldsymbol{x}_1) + \mathscr{F}^{(2)}_{i,jk}(\boldsymbol{x}_1), \tag{D 8}$$

where $\boldsymbol{r}=\boldsymbol{x}_1-\boldsymbol{x}_2$. If the order of $\mathscr{F}(\boldsymbol{x}_1)$ and $\mathscr{F}(\boldsymbol{x}_2)$ is the same, they are equivalent, that is, $\mathscr{F}(\boldsymbol{x}_2)$ calculated by the definition and by the transformation above are identical. We denote this transformation by a matrix $\mathscr{S}_F$ as

$$\mathscr{F}(\boldsymbol{x}_2) = \mathscr{S}_F(\boldsymbol{x}_2,\boldsymbol{x}_1)\cdot\mathscr{F}(\boldsymbol{x}_1). \tag{D 9}$$

The explicit form of the transformation matrix $\mathscr{S}$ for vector $(\mathscr{F}^{(0)}_i, \mathscr{F}^{(1)}_{x,i}, \mathscr{F}^{(1)}_{y,i}, \mathscr{F}^{(1)}_{z,i})$ is

written as

$$\mathscr{S}_F(\boldsymbol{x}_2, \boldsymbol{x}_1) = \begin{bmatrix} \boldsymbol{I} & 0 & 0 & 0 \\ \boldsymbol{A}_x & \boldsymbol{I} & 0 & 0 \\ \boldsymbol{A}_y & 0 & \boldsymbol{I} & 0 \\ \boldsymbol{A}_z & 0 & 0 & \boldsymbol{I} \end{bmatrix}, \tag{D 10}$$

where

$$\boldsymbol{A}_x = \begin{bmatrix} r_x & 0 & 0 \\ r_y & 0 & 0 \\ r_z & 0 & 0 \end{bmatrix}, \quad \boldsymbol{A}_y = \begin{bmatrix} 0 & r_x & 0 \\ 0 & r_y & 0 \\ 0 & r_z & 0 \end{bmatrix}, \quad \boldsymbol{A}_z = \begin{bmatrix} 0 & 0 & r_x \\ 0 & 0 & r_y \\ 0 & 0 & r_z \end{bmatrix}, \tag{D 11}$$

where $\boldsymbol{r} = \boldsymbol{x}_1 - \boldsymbol{x}_2$.

D.2. *Velocity derivatives*

We describe the transformation of the origin of the velocity derivatives. By the definition

$$\mathscr{V}^{(m)}_{i,k\ldots}(\boldsymbol{x}_1) = \frac{1}{m!}[\nabla^m_{k\ldots} v_i](\boldsymbol{x}_1), \tag{D 12}$$

the velocity disturbance at $\boldsymbol{x}$ around $\boldsymbol{x}_1$ is given by

$$v_i(\boldsymbol{x}) = \sum_{m=0}^{\infty} \mathscr{V}^{(m)}_{i,k\ldots}(\boldsymbol{x}_1)(\boldsymbol{x} - \boldsymbol{x}_1)^m_{k\ldots}. \tag{D 13}$$

From the above equation, we obtain the transformation among the velocity derivatives as

$$\mathscr{V}^{(n)}_{i,l\ldots}(\boldsymbol{x}_2) = \sum_{m=n}^{\infty} {}_mC_n \mathscr{V}^{(m)}_{i,k\ldots l\ldots}(\boldsymbol{x}_1)(\boldsymbol{x}_2 - \boldsymbol{x}_1)^{m-n}_{k\ldots}, \tag{D 14}$$

or, introducing the operator $\mathscr{S}_V$, as

$$\mathscr{V}(\boldsymbol{x}_2) = \mathscr{S}_V(\boldsymbol{x}_2, \boldsymbol{x}_1) \cdot \mathscr{V}(\boldsymbol{x}_1). \tag{D 15}$$

The explicit form of $\mathscr{S}_V$ is given by

$$\begin{bmatrix} \mathscr{V}^{(0)} \\ \mathscr{V}^{(1)} \\ \mathscr{V}^{(2)} \\ \mathscr{V}^{(3)} \\ \vdots \end{bmatrix}(\boldsymbol{x}_2) = \begin{bmatrix} \boldsymbol{I} & \boldsymbol{r}\odot^{(1)} & 2\boldsymbol{r}\boldsymbol{r}\odot^{(2)} & 3\boldsymbol{r}\boldsymbol{r}\boldsymbol{r}\odot^{(3)} & \cdots \\ 0 & \boldsymbol{I} & 2\boldsymbol{r}\odot^{(1)} & 3\boldsymbol{r}\boldsymbol{r}\odot^{(2)} & \cdots \\ 0 & 0 & \boldsymbol{I} & 3\boldsymbol{r}\odot^{(1)} & \cdots \\ 0 & 0 & 0 & \boldsymbol{I} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} \mathscr{V}^{(0)} \\ \mathscr{V}^{(1)} \\ \mathscr{V}^{(2)} \\ \mathscr{V}^{(3)} \\ \vdots \end{bmatrix}(\boldsymbol{x}_1), \tag{D 16}$$

where $\boldsymbol{r} = \boldsymbol{x}_2 - \boldsymbol{x}_1$, and $\odot^{(n)}$ denotes the n-fold contraction such as

$$(\mathscr{A} \odot^{(2)} \mathscr{B})_{k\ldots,l\ldots} = \mathscr{A}_{ijk\ldots}\mathscr{B}_{ijl\ldots}\,. \tag{D 17}$$

REFERENCES

Batchelor, G. K. 1972 Sedimentation in a dilute dispersion of spheres. *J. Fluid Mech.* **52**, 245–268.

Beenakker, C. W. J. 1986 Ewald sum of the Rotne–Prager tensor. *J. Chem. Phys.* **85**, 1581–1582.

Brady, J. F. & Bossis, G. 1988 Stokesian dynamics. *Annu. Rev. Fluid Mech.* **20**, 111–157.

Brady, J. F., Phillips, R. J., Lester, J. C. & Bossis, G. 1988 Dynamic simulation of hydrodynamically interacting suspensions. *J. Fluid Mech.* **195**, 257–280.

Carrier, J., Greengard, L. & Rokhlin, V. 1988 A fast adaptive multipole algorithm for particle simulations. *J. Comput. Phys.* **9**, 669–686.

Cichocki, B., Ekiel-Jeżewska, M. L. & Wajnryb, E. 1999 Lubrication corrections for three-particle contribution to short-time self-diffusion coefficients in colloidal dispersions. *J. Chem. Phys.* **111**, 3265–3273.

Damour, T. & Iyer, B. R. 1991 Multipole analysis for electromagnetism and linearized gravity with irreducible Cartesian tensors. *Phys. Rev.* D **43**, 3259–3272.

Durlofsky, L., Brady, J. F. & Bossis, G. 1987 Dynamic simulation of hydrodynamically interacting particles. *J. Fluid Mech.* **180**, 21–49.

Gluckman, M. J., Pfeffer, R. & Weinbaum, S. 1971 A new technique for treating multiparticle slow viscous flow: axisymmetric flow past spheres and spheroids. *J. Fluid Mech.* **50**, 705–740.

Greengard, L. & Rokhlin, V. 1987 A fast algorithm for particle simulations. *J. Comput. Phys.* **73**, 325–348.

Ichiki, K. & Brady, J. F. 2001 Many-body effects and matrix-inversion in low-Reynolds-number hydrodynamics. *Phys. Fluids* **13**, 350–353.

Jeffrey, D. J. & Onishi, Y. 1984 Calculation of the resistance and mobility functions for two unequal spheres in low-Reynolds-number flow. *J. Fluid Mech.* **139**, 261–290.

Kao, S. & Mason, S. G. 1975 Dispersion of particles by shear. *Nature* **253**, 619–621.

Ladd, A. J. C. 1988 Hydrodynamic interactions in a suspension of spherical particles. *J. Chem. Phys.* **88**, 5051–5063.

Ladyzhenskaya, O. A. 1969 *The Mathematical Theory of Viscous Incompressible Flow*, 2nd Edn. Gordon & Breach.

Mazur, P. & van Saarloos, W. 1982 Many-sphere hydrodynamic interactions and mobilities in a suspension. *Physica* A **115**, 21–57.

Mo, G. & Sangani, A. S. 1994 A method for computing Stokes flow interactions among spherical objects and its application to suspensions of drops and porous particles. *Phys. Fluids* **6**, 1637–1652.

Nitsche, J. M. & Batchelor, G. K. 1997 Break-up of a falling drop containing dispersed particles. *J. Fluid Mech.* **340**, 161–175.

Saad, Y. & Shultz, M. H. 1986 GMRES – a generalized minimal residual algorithm for solving nonsymmetric linear-systems. *SIAM J. Sci. Statist. Comput.* **7**, 856–869.

Sangani, A. S. & Mo, G. 1996 An $O(N)$ algorithm for Stokes and Laplace interactions of particles. *Phys. Fluids* **8**, 1990–2010.

van der Vorst, H. A. 1992 BI-CGSTAB: A fast and smoothly converging variant of BI-CG for the solution of nonsymmetric linear systems. *SIAM J. Sci. Statist. Comput.* **13**, 631–644.

Weiss, R. 1996 *Parameter-free Iterative Linear Solvers.* Berlin: Akademie Verlag.

Zhang, S. L. 1997 GPBi-CG: Generalized product-type methods based on Bi-CG for solving nonsymmetric linear systems. *SIAM J. Sci. Comput.* **18**, 537–551.