An Efficient Algorithm for Hydrodynamical Interaction of Many Deformable Drops

Alexander Z. Zinchenko and Robert H. Davis

Department of Chemical Engineering, University of Colorado, Boulder, Colorado 80309-0424
E-mail: zinchenk@rastro.colorado.edu, robert.davis@colorado.edu

An efficient and accurate 3D algorithm for dynamical simulations of many deformable drops with strong hydrodynamical interactions at zero Reynolds numbers is developed. The drop-to-medium viscosity ratio, λ, and the Bond number are arbitrary, and the drops are subject to gravity with stationary triply periodic boundary conditions. The algorithm, at each step, is a hybrid of boundary-integral and economical multipole techniques, with extensive use of rotational transformations and economical truncation of multipole expansions to optimize near-field interactions. A significant part of the code is the new, "best paraboloid-spline" technique for calculating the normal vectors and curvatures on drop surfaces, which greatly improves the quality of long-time simulations. Examples show the phenomenon of clustering in a concentrated sedimenting emulsion for $\lambda = 0.25$ and 1, which leads to an increase in the average sedimentation velocity with time. A high efficiency of the method is demonstrated, with two orders-of-magnitude gains over the standard $O(N^2 N_\triangle^2)$ boundary-integral technique for $N \sim 10^2$ drops in a periodic cell with $N_\triangle \sim 10^3$ triangular boundary elements per drop, so that typical long-time dynamical simulations can be performed in a few days or weeks on a standard workstation (as compared to the several years which would be required for the same simulations using standard boundary-integral techniques). The effects of drop triangulation and truncation of multipole expansions on dynamical simulations are assessed. © 2000 Academic Press

Key Words: deformable drops; hydrodynamical interactions; boundary integrals; Stokes flow; sedimentation; multipole methods.

1. INTRODUCTION

The motion of deformable drops at small Reynolds number is relevant to blood rheology and many processes in chemical engineering and biotechnology. So far, computational progress has been primarily restricted to simulations of one or two drops by boundary-integral methods [1] (see [2, 3], for example, for an extended list of recent studies). Limited

3D simulations exist for more (10–12) drops falling under gravity [4] or subject to shear with periodic boundaries [5]; the methods used in these studies are direct, with the computational requirements scaling as $O(N^2 N_\triangle^2)$ per time step (where N is the number of drops and $N_\triangle$ is the number of boundary elements per drop), thus not allowing calculations to proceed to $N \geq O(10^2)$ with adequate resolutions $N_\triangle \geq O(10^3)$, especially for an arbitrary drop-to-medium viscosity ratio, λ. In 2D, larger systems ($N = 25$–49) have been considered [6, 7], still by an $O(N^2 N_\triangle^2)$ method, and the simplest use of Taylor expansions to optimize remote interactions was tested [6], although with only a modest gain. Greater progress has been made in simulating large hydrodynamical systems of solid spheres. Ladd [8] was able to dynamically simulate up to 32000 spheres in sedimentation by a lattice-Boltzmann method [9, 10]. However, it would be a non-trivial task to adapt this approach to boundary conditions other than no slip, and we are not aware if this method has been applied to drops, with or without deformations. Sangani and Mo [11] developed the first hydrodynamical version of the traditional fast multipole method (FMM) [12] with periodic boundaries for spherical particles and applied it to study sedimentation, effective viscosity, and permeability of random static configurations. By taking a few multipoles per particle (and adding lubrication analytically [13], when necessary), they circumvented the difficulty with the traditional FMM becoming inefficient for high-order multipoles and could consider large systems, up to $N = 8000$; potentially, their code should be applicable to spherical drops as well. Unfortunately, this potential is not realizable for drops with deformation; each of them is necessarily an object with a large number of parameters (collocation nodes on the surface), thus limiting dynamical simulations to much smaller N.

In the present paper, an efficient and accurate 3D algorithm for dynamical simulations of many deformable drops with strong hydrodynamical interactions at zero Reynolds number is constructed. For simplicity, a stationary, cubic periodic cell is assumed, and the drops are subject to gravity, which includes applications to emulsion sedimentation; a generalization for rheology simulations will be possible in future work. The algorithm, at each time step, is a hybrid of the boundary-integral and economical multipole techniques; in the multipole part, however, our code does not follow the line of FMM, but rather develops the approach initiated in conductivity simulations by Zinchenko for 2D [14, 15] and 3D [16, 17] problems. Derivation of the boundary-integral equation for sedimenting systems requires some caution (because an additive constant in the velocity affects calculation of the sedimentation rate), and we devote some space to it in Section 2. In Subsection 3.1, a general scheme for fast summation of interactions between collocation nodes on drop surfaces is described. Drops are sliced gridlessly into compact blocks. The near-field interactions between blocks, when possible, are calculated by multipole reexpansions from Lamb's singular to regular forms, using a rotation-based scheme to reduce the cost of this operation from $O(n^4)$ to $O(n^3)$, where n is the order of multipoles retained for the pair (Subsection 3.3). Although rotational transformations of spherical harmonics by Wigner functions have long been known in quantum mechanics [18], it was not until more recent work [16] that their relevance to fast summation of interactions was recognized and exploited (surprisingly, the FMM papers prior to 1996 have all used the less efficient, direct $O(n^4)$ scheme for their cell-to-cell reexpansions). Lamb's singular moments for individual blocks are also calculated by rotations, which gives a fixed, but almost several-fold, gain compared to direct moment evaluation (Subsection 3.2). "Far-field" interactions, associated with periodic images, are treated by Taylor expansions of arbitrary order and Ewald-like forms for Green's functions (Subsection 3.4). Another feature borrowed from [16] is the "economical truncation" of

multipole expansions, i.e., a broad spectrum of truncation bounds strongly dependent on mutual geometry of the blocks for optimized performance; unlike in [16], however, we have to construct these bounds in a more empirical manner for a given precision (Subsection 3.5). Although loaded with multipole details, necessarily cumbersome in hydrodynamics, our scheme for summing interactions seeks to minimize the use of costly direct summations in a logically simple way. Fast summation is only a part of the problem, and Section 4 describes a new, "best paraboloid-spline" (BPS) method for calculating the curvatures and normals, which greatly improves the quality of long-time dynamical simulations. Subsection 5.1 outlines the passive mesh stabilization technique [3] used to maintain the quality of our unstructured drop triangulations with fixed topology.

Examples of long-time dynamical simulations in Section 6 demonstrate the phenomenon of clustering in a concentrated sedimenting emulsion of many deformable drops at $\lambda = 0.25$ and 1, and also show a high efficiency of the code, with two orders-of-magnitude gains over the $O(N^2 N_\triangle^2)$ method at sufficient accuracy for $N = O(10^2)$ and $N_\triangle = O(10^3)$. The effects of multipole truncation and triangulation on long-time simulations are also analyzed. Section 7 discusses the prospects and limitations of the proposed approach.

Some other recent methods for fast summation of interactions include particle-particle/particle-mesh (P^3M) (see [19–23] and references therein), FFT-accelerated FMM [24], rotation-accelerated FMM [25], and finally, the new 3D FMM [26]. These methods are general purpose for an arbitrary distribution of charges, while our approach substantially uses natural grouping of collocation nodes into interfacial surfaces (which still includes a large number of applications, with deformable drops, arbitrary-shaped suspended particles, granular materials, etc.). However, these previous developments and implementations, to our knowledge, are for Coulombic interactions only, thus not allowing for comparisons with the present approach in the solution of the same hydrodynamic (or similar) problems. Even with rotations incorporated in the latest FMM versions [25, 26], the approach initiated in [16] and further developed herein is deeply different from FMM, in treating both close and remote interactions. Very high gains achieved by our code over the $O(N^2 N_\triangle^2)$ method for practical N and $N_\triangle$, combined with a simple logic (compared to P^3M and FMM), can make our approach attractive for disperse media simulations. For $N \sim 10^3$, alternatives should be considered. However, regardless of the method, it may not be possible to dynamically simulate such large concentrated systems of deformable 3D drops with adequate resolution $N_\triangle \sim 10^3$ on present-day workstations. All timings below are for a DEC 500au, a single processor 500 MHz UNIX workstation with FORTRAN 77 optimizing compiler.

2. BOUNDARY-INTEGRAL FORMULATION

Consider an infinite set of deformable drops of the same density ρ_{int} and viscosity μ_{int} slowly settling under gravity in a medium of density ρ_{ext} and viscosity μ_{ext}. An equivalent radius of non-deformed drops is a_o (assumed, for simplicity, to be the same for all drops) and the Bond number is $\mathcal{B} = (\rho_{\text{int}} - \rho_{\text{ext}})g a_o^2/\sigma$, where σ is the constant surface tension. The drop system is obtained from the basic configuration of N drops with surface centroids $\mathbf{x}_1^c \ldots \mathbf{x}_N^c$ in the cell V by triply periodic continuation into the whole space. No external flow is imposed, and so the basic cell V does not change with time and can be taken as the unit cube $[0, 1) \times [0, 1) \times [0, 1)$, if the cell side L is chosen as the characteristic length for nondimensionalization; the x_3-axis is antiparallel to the gravity acceleration vector $\mathbf{g}$. The

fluid velocity $v(x)$ is triply periodic, while the pressure has a linear part plus a periodic part. A system of boundary-integral equations for $v(x)$ on drop surfaces $S_1, \ldots S_N$ is facilitated through the use of the periodic nondimensional Green functions $G^{(k)}(x)$, $k = 1, 2, 3$ [27–29] and corresponding stress tensors $\tau^{(k)}(x)$ (these functions are discussed more in detail in Subsection 3.4). With the normalization used in the present work, the vectors $G^{(k)}(x)$ and the corresponding pressures $q^{(k)}(x)$ satisfy

$$\nabla^2 G^{(k)}(x) - \nabla q^{(k)}(x) = \nabla \cdot \tau^{(k)}(x) = \sum_m \delta(x - m)\, e_k, \tag{2.1}$$

where the summation is over all latice points $m = (m_1, m_2, m_3)$ with integer m_1, m_2, m_3, and e_k ($k = 1, 2, 3$) are basis vectors. Unlike $G^{(k)}(x)$, the pressure $q^{(k)}(x)$ and stress tensor

$$\tau^{(k)} = -q^{(k)} I + (\nabla + \nabla^T)\, G^{(k)} \tag{2.2}$$

are linear plus periodic functions. The additive constants in $G^{(k)}$ are chosen so that the average of $G^{(k)}$ over V is zero (which is equivalent to the requirement that the flux of $G^{(k)}$ through any face of the periodic cell is zero). Besides, we choose the reference frame so that the average fluid velocity $\langle v \rangle$ over V is also zero.

Green's theorem gives, for a point $y \in V$ lying outside the drops,

$$v_k^{\text{ext}}(y) = -\sum_{\beta=1}^{N} \int_{S_\beta} \left[v(x) \cdot \tau^{(k)}(x - y) \cdot n(x) - G^{(k)}(x - y) \cdot T^{\text{ext}}(x) \cdot n(x) \right] dS_x. \tag{2.3}$$

Here, $n(x)$ is the outward unit normal at $x \in S_\beta$. The velocity v and the corresponding stress tensor T (without the hydrostatic term) have been made nondimensional by choosing the scales $L^2(\rho_{\text{int}} - \rho_{\text{ext}})g/\mu_{\text{ext}}$ and $(\rho_{\text{int}} - \rho_{\text{ext}})gL$ for v and T, respectively; the indices *ext* and *int* mark the values related to the continuous and drop phases, respectively. To rigorously derive (2.3), Green's theorem is applied to the volume D_f of the cell V lying outside the drops (shaded in Fig. 1). Periodic parts of the integrand (2.3) do not contribute to the integrals over the cell boundaries. Linear parts make nonzero contributions (if some drops intersect ∂V). However, by the relations

$$\int_{\partial V_{\text{ext}} + \partial V_{\text{int}}} (v \cdot n) x \, dS = \int_V v \, dV = 0, \tag{2.4a}$$

$$\int_{\partial V_{\text{ext}} + \partial V_{\text{int}}} \left[G^{(k)}(x - y) \cdot n(x) \right] x \, dS_x = \int_V G^{(k)}(x - y) \, dV_x = 0 \tag{2.4b}$$

(where ∂V_{ext} and ∂V_{int} are parts of the boundary ∂V lying outside and inside the drops, respectively), the integrals of the linear terms over ∂V_{ext} can be combined with those over drop surface portions lying inside V to produce full surface integrals, which rigorously gives (2.3).

Similar to (2.3),

$$0 = \sum_{\beta=1}^{N} \int_{S_\beta} \left[\lambda v(x) \cdot \tau^{(k)}(x - y) - G^{(k)}(x - y) \cdot T^{\text{int}}(x) \right] \cdot n(x) \, dS_x \tag{2.5}$$
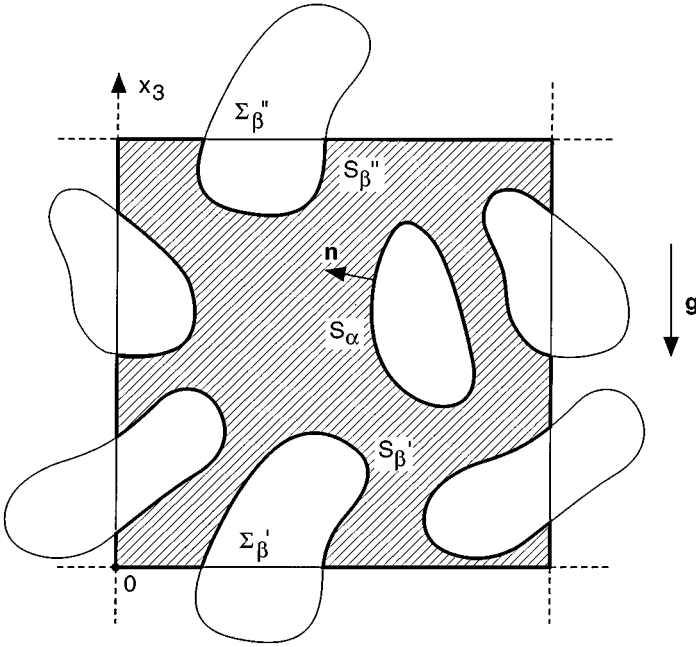
FIG. 1. The derivation of the boundary-integral equation for a system of sedimenting deformable drops (a 2D sketch, not to scale). The boundary of D_f is marked bold. For drop S_β and its periodic image, the integrals of $(v \cdot n)x$ and $(G^{(k)} \cdot n)x$ over the surface portions S'_β, S''_β and the cell boundary portions $\sum'_\beta$, $\sum''_\beta$ combine to boundary integrals of $(v \cdot n)x$ and $(G^{(k)} \cdot n)x$ over the entire S_β.

for any $y \in V$ lying outside the drops, where $\lambda = \mu_{\text{int}}/\mu_{\text{ext}}$ is the drop-to-medium viscosity ratio. Using the boundary conditions and taking the limit $y \rightarrow S_\alpha$ now gives a system of nondimensional boundary-integral equations for $v(x)$ in a standard way:

$$v_k(y) = 2\kappa \sum_{\beta=1}^{N} \int_{S_\beta} v(x) \cdot \tau^{(k)}(x - y) \cdot n(x) \, dS_x + F_k(y). \tag{2.6}$$

Here, $\kappa = (\lambda - 1)/(\lambda + 1)$, the inhomogeneous term is

$$F(y) = \frac{2}{\lambda + 1} \sum_{\beta=1}^{N} \int_{S_\beta} f(x)G(x - y) \cdot n(x) \, dS_x, \tag{2.7a}$$

with

$$f(x)|_{S_\beta} = a^2(k_1 + k_2)/\mathcal{B} + (x - x^c_\beta)_3, \tag{2.7b}$$

$G = (G^{(1)}, G^{(2)}, G^{(3)})$ is the symmetric Green tensor, $a = a_o/L$ is the nondimensional non-deformed radius, $k_1(x)$ and $k_2(x)$ are the dimensionless principal surface curvatures, and x^c_β is the surface centroid

$$x^c_\beta = \frac{1}{S_\beta} \int_{S_\beta} x \, dS. \tag{2.8}$$

It is inconvenient that the kernel $\boldsymbol{\tau}^{(k)}(\boldsymbol{x} - \boldsymbol{y})$, as follows from the derivation of (2.6), is not triply periodic, but contains a linearly growing part from the pressure $q^{(k)}$. The kernel $\boldsymbol{\tau}^{(k)}$ in (2.6) could be changed to the periodic kernel

$$\tilde{\boldsymbol{\tau}}^{(k)}(\boldsymbol{x} - \boldsymbol{y}) = \boldsymbol{\tau}^{(k)}(\boldsymbol{x} - \boldsymbol{y}) - (x - y)_k \boldsymbol{I} \tag{2.9}$$

by simply adding a suitable vector constant to $\boldsymbol{v}(\boldsymbol{x})$, using the identities

$$2 \int_{S_\beta} \tilde{\boldsymbol{\tau}}^{(k)}(\boldsymbol{x} - \boldsymbol{y}) \cdot \boldsymbol{n}(\boldsymbol{x}) \, dS_x = \begin{cases} (1 - 2V_\beta) \, \boldsymbol{e}_k, & \boldsymbol{y} \in S_\beta^T \\ -2V_\beta \boldsymbol{e}_k, & \boldsymbol{y} \text{ outside } S_\beta^T, \end{cases} \tag{2.10}$$

where V_β is the volume of drop β, and S_β^T stands for all periodic images of S_β. As for the standard, free-space case [1], the eigenvalues of the integral operator with the kernel $2\tilde{\boldsymbol{\tau}}^{(k)}$ can be shown to lie within $[-1, 1]$; however, the eigenfunctions are different (see (2.10)). To avoid complications with Wielandt's deflation of the extreme eigenvalues ± 1 (necessary to speed up the convergence of the iterative solution), one can still consider (2.6) in the subspace of $\boldsymbol{v}(\boldsymbol{x})$ with zero fluxes through every S_α. In this subspace, the RHS of (2.6) is a periodic operator, the eigenfunctions for $\kappa = 1$ are standard arbitrary rigid-body motions, and -1 is no longer an eigenvalue for the adjoint. After the standard Wielandt deflation [1, 48] in this space, $\boldsymbol{\tau}^{(k)}(\boldsymbol{x} - \boldsymbol{y})$ can be replaced by the more convenient $\tilde{\boldsymbol{\tau}}^{(k)}(\boldsymbol{x} - \boldsymbol{y})$ using (2.9). Besides the deflation, it is important to reduce the change of the integrands on drop surfaces, like we did in (2.7b), in order to accelerate convergence in the multipole part (Section 3) of the algorithm; for the velocities, this goal is achieved by considering fluctuations from the average values. Thus, we proceed from (2.6) to an equivalent, deflated system of equations for

$$\boldsymbol{w}(\boldsymbol{y}) = \boldsymbol{v}(\boldsymbol{y}) - \kappa \hat{\boldsymbol{v}}(\boldsymbol{y}), \tag{2.11}$$

where the hat stands for the rigid-body projection (see below),

$$\boldsymbol{w}(\boldsymbol{y}) = \boldsymbol{u}(\boldsymbol{y}) - \frac{\boldsymbol{n}(\boldsymbol{y})}{S_\alpha} \int_{S_\alpha} \boldsymbol{u} \cdot \boldsymbol{n} \, dS, \qquad \boldsymbol{y} \in S_\alpha, \tag{2.12a}$$

$$\boldsymbol{u}(\boldsymbol{y}) = \kappa \left[2 \sum_{\beta=1}^N \int_{S_\beta} \boldsymbol{Q}(\boldsymbol{x}) \cdot \tilde{\boldsymbol{\tau}}(\boldsymbol{x} - \boldsymbol{y}) \cdot \boldsymbol{n}(\boldsymbol{x}) \, dS_x - \hat{\boldsymbol{w}}(\boldsymbol{y}) + \langle \boldsymbol{w} \rangle_\alpha \right.$$

$$\left. - 2 \sum_{\beta=1}^N \langle \boldsymbol{w} \rangle_\beta V_\beta + 2 \sum_{\beta=1}^N \int_{S_\beta} (\boldsymbol{w} \cdot \boldsymbol{n}) \left(\boldsymbol{x} - \boldsymbol{x}_\beta^c \right) dS \right] + \boldsymbol{F}(\boldsymbol{y}). \tag{2.12b}$$

Here,

$$\langle \boldsymbol{w} \rangle_\beta = \frac{1}{S_\beta} \int_{S_\beta} \boldsymbol{w} \, dS, \qquad \boldsymbol{Q}(\boldsymbol{x})|_{S_\beta} = \boldsymbol{w}(\boldsymbol{x}) - \langle \boldsymbol{w} \rangle_\beta, \tag{2.13}$$

and the tensor $\tilde{\boldsymbol{\tau}} = \{\tilde{\tau}_{ij}^{(k)}\}$ is now symmetric in all three indices (see Subsection 3.4). A convenient expression [2] for the rigid-body projections $\hat{\boldsymbol{v}}$ and $\hat{\boldsymbol{w}}$ can be used, for example,

$$\hat{\boldsymbol{w}}(\boldsymbol{y})|_{S_\alpha} = \langle \boldsymbol{w} \rangle_\alpha + \boldsymbol{B} \times \left(\boldsymbol{y} - \boldsymbol{x}_\alpha^c \right), \tag{2.14}$$

where the vector $\boldsymbol{B} = (B_1, B_2, B_3)$ is calculated from the solution of a 3×3 system:

$$D_{ij} B_j = \int_{S_\alpha} \boldsymbol{w}(\boldsymbol{x}) \cdot \left[\boldsymbol{e}_i \times \left(\boldsymbol{x} - \boldsymbol{x}_\alpha^c \right) \right] dS, \qquad i = 1, 2, 3, \tag{2.15}$$

with the positive-definite matrix

$$D_{ij} = \int_{S_\alpha} \left[\delta_{ij} \left(\boldsymbol{x} - \boldsymbol{x}_\alpha^c \right)^2 - \left(\boldsymbol{x} - \boldsymbol{x}_\alpha^c \right)_i \left(\boldsymbol{x} - \boldsymbol{x}_\alpha^c \right)_j \right] dS. \tag{2.16}$$

Once $\boldsymbol{w}$ is determined from (2.12), the velocity $\boldsymbol{v}$ can be recovered:

$$\boldsymbol{v} = \boldsymbol{w} + \kappa \hat{\boldsymbol{w}} / (1 - \kappa). \tag{2.17}$$

In addition to (2.12), we have developed a more straightforward deflation scheme, using directly the eigenfunctions for the kernel $2\hat{\tau}(\boldsymbol{x} - \boldsymbol{y})$ with eigenvalues ± 1 which can be derived from (2.10). The two schemes gave practically identical results in tests (which is expected, since they are equivalent in the limit of fine triangulations), but (2.12) is preferred for its simplicity.

Equation (2.12) is solved at each time step by the simplest method of successive substitutions, and the drop surfaces are updated by the second-order Runge–Kutta time integration scheme, with an artificial tangential velocity added to prevent mesh distortion (Subsection 5.1). For extreme ($\lambda \ll 1$ or $\lambda \gg 1$) viscosity ratios and high volume fractions, when drops are close, successive substitutions are poorly convergent, and alternative, biconjugate gradient iterations (as discussed at some length by Zinchenko *et al.* [2]) would be much preferable. This alternative requires, however, the adjoint of the discrete form of the boundary-integral operator (2.12), which is straightforward to calculate for standard point-to-point summations, but more difficult when the combined boundary-integral-multipole scheme (Section 3) of the present work is used. We have chosen the simplest iterative method not to overcomplicate the code logic at this stage, leaving the most difficult case of extreme λ at high volume fractions for further investigation. For the same reason, only the conventional form (2.7) of the inhomogeneous term is used in the present work. A new curvatureless form [3] may be a preferable option for many drops at larger deformations.

Since the emulsion, on average, is at rest ($\langle v \rangle = 0$), the instantaneous sedimentation rate $\boldsymbol{v}_S$ is the average velocity over the drop phase. By the Gauss theorem, calculation of $\boldsymbol{v}_S$ is reduced to surface integrals,

$$\boldsymbol{v}_S = \frac{1}{c} \sum_{\alpha=1}^{N} \int_{S_\alpha} (\boldsymbol{v} \cdot \boldsymbol{n}) \left(\boldsymbol{x} - \boldsymbol{x}_\alpha^c \right) dS, \tag{2.18}$$

where c is the volume fraction of the drop phase.

The central part of our method is an efficient calculation of the boundary-integrals for the inhomogenous (2.7a) and double-layer (2.12b) terms. A standard, point-to-point boundary-integral method has an $O(N^2 M^2)$ computational cost, where M is the number of collocation points per drop, thus heavily restricting dynamical simulations to small N, even with the fastest calculation of Green's functions by suitable interpolations. The method described below is a certain hybrid of the boundary-integral and economical multipole techniques and follows in many ways the procedure developed for thermal/electrostatical interaction

of many spheres by Zinchenko [16], with extensive use of Wigner functions and economical truncation of multipole expansions to optimize near-field summations.

## 3. FAST ALGORITHM FOR THE BOUNDARY-INTEGRAL OPERATORS

3.1. *General Scheme*

Let every drop surface S_α ($\alpha = 1, \ldots N$) be represented by an unstructured grid of flat triangles (Fig. 2) with M vertices x_j (called thereafter the collocation nodes). For any smooth integrand $\varphi(x)$ on S_α, a simple trapezoidal rule can be used, with reassigning triangle contributions to vertices (a procedure due to Rallison [30], who used it in one-drop calculations),

$$\int_{S_\alpha} \varphi(x)\, dS \approx \sum_{x_j \in S_\alpha} \varphi(x_j) \triangle S_j, \tag{3.1}$$

where

$$\triangle S_j = \frac{1}{3} \sum \triangle S \tag{3.2}$$

and the summation is over all triangle areas $\triangle S$ with vertex x_j. Pozrikidis and co-workers [49] developed an alternative approach to surface discretization in terms of quadratic boundary elements with six nodes. Using quadratic elements is preferable for smooth solutions due to a higher rate of convergence, although the number of operations per element is increased several fold compared to (3.1). Flat triangle formulation, however, is expected to be more robust in extreme cases when high-curvature zones or near contacts between neighboring drops are formed (which are typical of our simulations in Section 6). We are not aware of
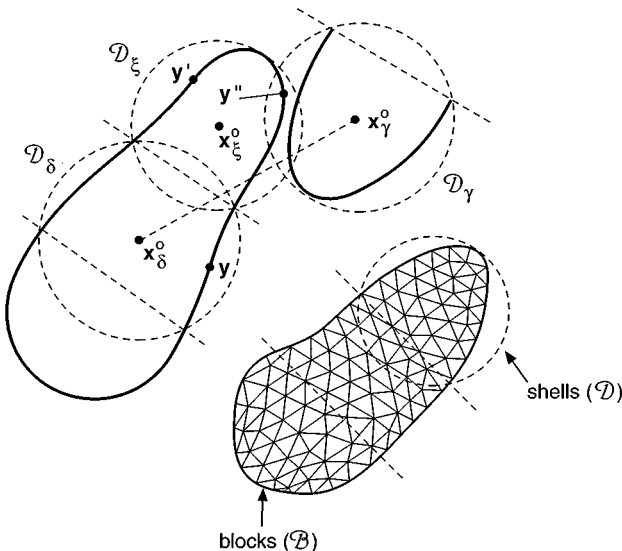
FIG. 2. The calcualtion of near-field interactions. Assuming $\mathcal{B}_\gamma \mid \mathcal{B}_\delta$ and $y' \mid \mathcal{B}_\gamma$, the contributions of block $\mathcal{B}_\gamma$ to the boundary integrals for y, y' and y'' are evaluated, respectively, by (i) reexpansion of Lamb's singular series from x_γ^o to x_δ^o, (ii) pointwise calculation of Lamb's singular series, and (iii) direct point-to-point summation.

3D boundary-integral solutions with quadratic elements for such extreme cases, and so a detailed comparison between the two approaches cannot be made at present.

To calculate "self-interactions," i.e., the contributions of the surface $S_\alpha \ni y$ to single (2.7a) and double-layer (2.12b) boundary integrals for collocation nodes y, we subtract out the free-space contributions from the periodic Stokeslet and stresslet (cf. [16]),

$$\boldsymbol{G}(\boldsymbol{r}) = \boldsymbol{G}_0(\boldsymbol{r}) + \boldsymbol{G}_1(\boldsymbol{r}), \qquad \tilde{\boldsymbol{\tau}}(\boldsymbol{r}) = \boldsymbol{\tau}_0(\boldsymbol{r}) + \boldsymbol{\tau}_1(\boldsymbol{r}), \tag{3.3a}$$

$$\boldsymbol{G}_0(\boldsymbol{r}) = -\frac{1}{8\pi}\left(\frac{\boldsymbol{I}}{r} + \frac{\boldsymbol{rr}}{r^3}\right), \qquad \boldsymbol{\tau}_0(\boldsymbol{r}) = \frac{3}{4\pi}\frac{\boldsymbol{rrr}}{r^5}. \tag{3.3b}$$

Upon substitutions of $\boldsymbol{G}_0$ and $\boldsymbol{\tau}_0$ into self-integrals, singularity subtraction is made, so that

$$\int_{S_\alpha} f(\boldsymbol{x})\boldsymbol{G}(\boldsymbol{x} - \boldsymbol{y}) \cdot \boldsymbol{n}(\boldsymbol{x})\, dS_{\boldsymbol{x}} \approx \sum_{\boldsymbol{x}_j \in S_\alpha} \boldsymbol{G}_1(\boldsymbol{x}_j - \boldsymbol{y}) \cdot \boldsymbol{W}(\boldsymbol{x}_j) \tag{3.4a}$$

$$+ \sum_{\boldsymbol{x}_j \in S_\alpha, \boldsymbol{x}_j \neq \boldsymbol{y}} [f(\boldsymbol{x}_j) - f(\boldsymbol{y})]\boldsymbol{G}_0(\boldsymbol{x}_j - \boldsymbol{y}) \cdot \boldsymbol{n}(\boldsymbol{x}_j)\, \triangle S_j \tag{3.4b}$$

and

$$2\int_{S_\alpha} \boldsymbol{Q}(\boldsymbol{x}) \cdot \tilde{\boldsymbol{\tau}}(\boldsymbol{x} - \boldsymbol{y}) \cdot \boldsymbol{n}(\boldsymbol{x})\, dS_{\boldsymbol{x}} \approx \boldsymbol{Q}(\boldsymbol{y}) \tag{3.5a}$$

$$+ \sum_{\boldsymbol{x}_j \in S_\alpha} \boldsymbol{Q}(\boldsymbol{x}_j) \cdot \boldsymbol{\tau}_1(\boldsymbol{x}_j - \boldsymbol{y}) \cdot \boldsymbol{W}(\boldsymbol{x}_j) \tag{3.5b}$$

$$+ \sum_{\boldsymbol{x}_j \in S_\alpha, \boldsymbol{x}_j \neq \boldsymbol{y}} [\boldsymbol{Q}(\boldsymbol{x}_j) - \boldsymbol{Q}(\boldsymbol{y})] \cdot \boldsymbol{\tau}_0(\boldsymbol{x}_j - \boldsymbol{y}) \cdot \boldsymbol{W}(\boldsymbol{x}_j). \tag{3.5c}$$

In (3.4)–(3.5) and in what follows, the weights $\boldsymbol{W}(\boldsymbol{x}_j)$ are

$$\boldsymbol{W}(\boldsymbol{x}_j) = f(\boldsymbol{x}_j)\boldsymbol{n}(\boldsymbol{x}_j)\, \triangle S_j \tag{3.6}$$

for the single-layer calculations and

$$\boldsymbol{W}(\boldsymbol{x}_j) = 2\boldsymbol{n}(\boldsymbol{x}_j)\, \triangle S_j \tag{3.7}$$

for the double-layer calculations. Terms (3.4b) and (3.5c) are handled by direct summations (obviously, an economical way being to consider pairs $(\boldsymbol{x}_i, \boldsymbol{x}_j)$ with $i < j$ only and accumulate contributions to the integrals for $\boldsymbol{y} = \boldsymbol{x}_i$ and $\boldsymbol{y} = \boldsymbol{x}_j$ simultaneously). The contributions of $\boldsymbol{G}_1$ and $\boldsymbol{\tau}_1$ to (3.4) and (3.5), called "far-field" contributions, are economically calculated by Taylor double series in powers of $(x_j - x_\alpha^c)_k$ and $(y - x_\alpha^c)_l$. These expansions are generated to an arbitrary order (Subsection 3.4). The number N of drops with centroids $\boldsymbol{x}_\alpha^c \in [0, 1)^3$ is assumed to be not too small, so that the minimal spherical shell around S_α centered at $\boldsymbol{x}_\alpha^c$ has the radius $d_\alpha < 1/2$, and the far-field expansion is convergent. For moderate N, many terms may be required, while for large N only a few terms suffice. The algorithm contains an additional parameter ε, which controls the truncation of this and other Taylor or multipole expansions (Subsection 3.5). The precision parameter ε is not a deviation from the standard $O(N^2 M^2)$ non-multipole solution in a rigorous sense, but it does correlate with this deviation (Section 6). The relation between ε and M is found by trials, to make the

error due to truncation typically negligible compared to the triangulation error (Section 6); as $\varepsilon \to 0$, all terms of the Taylor expansions are eventually included.

A far more involved scheme is used to calculate the boundary-integral contributions of surfaces $S_\beta \not\ni y$. When x and y in (2.7a) and (2.12b) are close, the kernels $G(x-y)$ and $\tilde{\tau}(x-y)$ are nearly singular, invalidating a straightforward use of the trapezoidal rule (3.1). Near-singularity subtraction [5] would suggest subtracting $f(x^*)$ and $Q(x^*)$ from $f(x)$ and $Q(x)$, respectively, where x^* is the collocation node on S_β or its periodic image, which is closest to y. This simple procedure is suitable and greatly improves the integrands for the free-space case [31], but it overlooks an additional term for the double-layer in a periodic system (important at large volume fractions), because the integral identities (2.10) are different from the free-space case. (This difference was not taken into account in [5], which made (2.3) therein not equivalent to the original boundary-integral formulation for $\lambda \neq 1$. This might have a small effect on the calculations in [5], since $\lambda \neq 1$ was considered only for $c = 10\%$.) Besides, it is not easy to reconcile near-singularity subtractions, made in a point-to-point manner in our algorithm, with multipole expansions. Obviously, these subtractions are meaningful only for close pairs (x, y), and using them for all pairs would considerably slow down the algorithm without necessity.

With these considerations in mind, the single-layer (2.7a) and double-layer (2.12b) integrals over $S_\beta \neq S_\alpha \ni y$ are approximated as

$$\int_{S_\beta} f(x)G(x-y) \cdot n(x)\, dS_x \approx \sum_{x_j \in S_\beta} G(x_j - y) \cdot W(x_j) \tag{3.8a}$$

$$- \sum_{k_{\alpha\beta}} \Theta(k_{\alpha\beta}, y) f(x^*) \sum_{x_j \in S_\beta} G_0(x_j + k_{\alpha\beta} - y) \cdot n(x_j) \triangle S_j \tag{3.8b}$$

and

$$2 \int_{S_\beta} Q(x) \cdot \tilde{\tau}(x-y) \cdot n(x)\, dS_x \approx \sum_{x_j \in S_\beta} Q(x_j) \cdot \tilde{\tau}(x_j - y) \cdot W(x_j) \tag{3.9a}$$

$$- \sum_{k_{\alpha\beta}} \Theta(k_{\alpha\beta}, y) Q(x^*) \sum_{x_j \in S_\beta} \tau_0(x_j + k_{\alpha\beta} - y) \cdot W(x_j). \tag{3.9b}$$

The summations in (3.8b) and (3.9b) are over all integer vectors $k_{\alpha\beta}$ with $\|x_\beta^c + k_{\alpha\beta} - x_\alpha^c\| < d_\alpha + d_\beta + h_o$, where h_o is the threshold parameter normally set to $0.3a$. The node $x^* \in S_\beta$ minimizes $\|x_j + k_{\alpha\beta} - y\|$, and

$$\Theta(k_{\alpha\beta}, y) = \max\left\{1 - \frac{\|y - x^* - k_{\alpha\beta}\|^2}{h_o^2}, 0\right\}. \tag{3.10}$$

According to (3.10), near-singularity subtractions are in effect only when the distance from y to a periodic image of S_β (calculated as the node-to-node minimum) is less than h_o. For $\|y - x^* - k_{\alpha\beta}\| \ll h_o$, Θ is close to unity, and the addends in (3.8b) and (3.9b) effectively cancel near-singular behavior of the addends in (3.8a) and (3.9a). At the same time, according to the integral identities for G_0 and τ_0, (3.8b) and (3.9b) disappear for fine triangulations. Gradual transition of Θ to zero, as $\|y - x^* - k_{\alpha\beta}\| \to h_o$, serves the smoothness of the near-singularity subtraction. Our most recent experiments, not included in Section 6, show that an additional smoothing of Θ, to make $\Theta \in \mathbb{C}^1$, slightly improves

the convergence of the velocity iterations. In a large system, only a few values of β can contribute to (3.8b) and (3.9b), and only one image of S_β, namely the one which minimizes the centroid-to-centroid distance from S_α, has to be considered. For small N (but still large enough to provide at least several-fold advantage of our code over standard boundary-integral techniques), more than one image of S_β can contribute to (3.8b) and (3.9b). The simplest search of β and $\mathbf{k}_{\alpha\beta}$ contributing to (3.8b) and (3.9b) is fast, and the nodes $\mathbf{y}$ are limited by $\|\mathbf{y} - \mathbf{x}_\beta^c - \mathbf{k}_{\alpha\beta}\| < d_\beta + h_o$, and so the near-singularity subtractions have a small $O(M^2 N)$ computational cost in the present applications, even with large N (Section 6).

Multipole expansions are used to handle (3.8a) and (3.9a), which would be most efficient for nearly spherical drops, since such an expansion at the drop center converges everywhere outside the minimal spherical shell around the drop. For strongly deformable drops at high volume fractions, however, these minimal shells considerably overlap, thus limiting the use of multipole expansions (see below). To increase the robustness of calculating (3.8a) and (3.9a), we first slice all drops into compact blocks $\mathcal{B}_1, \ldots \mathcal{B}_{N_B}$ ($N_B \geq N$). The simplest way to do it is to cut every drop S_α into

$$\left[l_\alpha \left(\frac{2\pi l_\alpha}{3V_\alpha} \right)^{1/2} - 1 \right] \tag{3.11}$$

pieces by planes orthogonal to the line of maximum elongation with equal spacing (Fig. 2), where l_α is the drop diameter ($=\max(\|\mathbf{x}_i - \mathbf{x}_j\|, \mathbf{x}_i, \mathbf{x}_j \in S_\alpha)$ and the brackets in (3.11) represent the greatest integer function; if a drop is compact, it coincides with its only block. We identify each block $\mathcal{B}_\gamma$ with the set of collocation nodes in it and construct a minimal spherical shell $\mathcal{D}_\gamma$ around $\mathcal{B}_\gamma$ with a center $\mathbf{x}_\gamma^o$ and radius d_γ^o. Exact determination of $\mathbf{x}_\gamma^o$ and d_γ^o would be too costly; a simple, stochastic procedure with sufficient accuracy is used instead (Appendix A). If a drop S_β consists of a single block $\mathcal{B}_\gamma$, the block shell center $\mathbf{x}_\gamma^o$ is not necessarily the surface centroid (2.8), and the radius $d_\gamma^o \neq d_\beta$, in general. To avoid confusion, we will reserve letters γ and δ to index the quantities related to blocks, while indices α and β will be used for entire drops.

The calculations of (3.8a) and (3.9a) follow the same logic, and we will only consider the single-layer sum (3.8a) herein; technical details about the differences between (3.8a) and (3.9a) can be found in Subsections 3.2–3.5. First, the free-space contribution of every block $\mathcal{B}_\gamma$ ($\gamma = 1, 2, \ldots N_B$) is expanded in Lamb's singular form [32], i.e.,

$$\sum_{\mathbf{x}_j \in \mathcal{B}_\gamma} \mathbf{G}_0(\mathbf{x}_j - \mathbf{y}) \cdot \mathbf{W}(\mathbf{x}_j) = \sum_{v=1}^{\infty} \left[\nabla \times (\mathbf{R}_\gamma \chi_{-(v+1)}) + \nabla \Phi_{-(v+1)} \right.$$
$$\left. - \frac{(v-2)\mathbf{R}_\gamma^2 \nabla p_{-(v+1)}}{2v(2v-1)} + \frac{(v+1)p_{-(v+1)}\mathbf{R}_\gamma}{v(2v-1)} \right]. \tag{3.12}$$

Here, the differential operations are with respect to $\mathbf{R}_\gamma = \mathbf{y} - \mathbf{x}_\gamma^o$, and the negative-order solid harmonics are

$$p_{-(v+1)}(\mathbf{R}_\gamma) = \sum_{m=-v}^{v} A_{-(v+1),m}^{(\gamma)} \left(\frac{d_\gamma^o}{R_\gamma} \right)^{v+1} Y_{v,m}(\mathbf{R}_\gamma), \tag{3.13a}$$

$$\Phi_{-(v+1)}(\mathbf{R}_\gamma) = \sum_{m=-v}^{v} B_{-(v+1),m}^{(\gamma)} \left(\frac{d_\gamma^o}{R_\gamma} \right)^{v+1} Y_{v,m}(\mathbf{R}_\gamma), \tag{3.13b}$$

$$\chi_{-(\nu+1)}(\mathbf{R}_\gamma) = \sum_{m=-\nu}^{\nu} C^{(\gamma)}_{-(\nu+1),m} \left(\frac{d^o_\gamma}{R_\gamma}\right)^{\nu+1} Y_{\nu,m}(\mathbf{R}_\gamma), \tag{3.13c}$$

$$A^{(\gamma)}_{-(\nu+1),-m} = (-1)^m \bar{A}^{(\gamma)}_{-(\nu+1),m}, \qquad B^{(\gamma)}_{-(\nu+1),-m} = (-1)^m \bar{B}^{(\gamma)}_{-(\nu+1),m} \text{ etc.,} \tag{3.13d}$$

where $Y_{\nu,m}(\mathbf{r})$ is the standard normalized spherical harmonic [33]

$$Y_{\nu,m}(\mathbf{r}) = \left[\frac{(2\nu+1)(\nu-m)!}{4\pi(\nu+m)!}\right]^{1/2} P^m_\nu(\cos\theta)\, e^{im\varphi} \qquad (m \geq 0) \tag{3.14a}$$

$$Y_{\nu,m}(\mathbf{r}) = (-1)^m \bar{Y}_{\nu,-m}(\mathbf{r}) \qquad (m \leq 0) \tag{3.14b}$$

for a vector $\mathbf{r} = (r\sin\theta\cos\varphi, r\sin\theta\sin\varphi, r\cos\theta)$, P^m_ν is the associated Legendre function (in the notation of [33, 34]), and the overbar denotes complex conjugation. The coefficients $A^{(\gamma)}_{-(\nu+1),m}$, $B^{(\gamma)}_{-(\nu+1),m}$, and $C^{(\gamma)}_{-(\nu+1),m}$, (called hereafter "singular near-field moments") are truncated by $\nu \leq \nu_{max} + 1$, with the truncation bound $\nu_{max}(\gamma, \varepsilon)$ determined as discussed below. These moments are precalculated for all blocks by a fast, rotation-based algorithm of Subsection 3.2, before handling the sums (3.8a). Also precalculated are a sufficient number of "far-field moments"

$$D^{(\gamma)}_{\nu,m,k} = (-1)^\nu \sum_{\mathbf{x}_j \in \mathcal{B}_\gamma} Z_{\nu,m}(\mathbf{x}_j - \mathbf{x}^o_\gamma) W_k(\mathbf{x}_j), \qquad |m| \leq \nu, \tag{3.15a}$$

$$E^{(\gamma)}_{\nu,m,k,l} = (-1)^\nu \sum_{\mathbf{x}_j \in \mathcal{B}_\gamma} Z_{\nu,m}(\mathbf{x}_j - \mathbf{x}^o_\gamma) W_k(\mathbf{x}_j)(\mathbf{x}_j - \mathbf{x}^o_\gamma)_l, \qquad |m| \leq \nu \tag{3.15b}$$

for every block $\mathcal{B}_\gamma$ (the truncation bounds for (3.15) are discussed below). The functions $Z_{\nu,m}(\mathbf{r})$ are denormalized solid harmonics,

$$Z_{\nu,m}(\mathbf{r}) = \frac{2\pi^{1/2} r^\nu Y_{\nu,m}(\mathbf{r})}{[(2\nu+1)(\nu-m)!(\nu+m)!]^{1/2}}, \tag{3.16}$$

convenient in the analysis of the far-field (Subsection 3.4). Now, to calculate the sum (3.8a) for $\mathbf{y} \in \mathcal{B}_\delta \subset S_\alpha$, each block $\mathcal{B}_\gamma$ of the surface $S_\beta \neq S_\alpha$ is temporarily shifted periodically to minimize the center-to-center distance $\|\mathbf{x}^o_\gamma - \mathbf{x}^o_\delta\|$ (the necessary integer displacements may be different for different blocks $\mathcal{B}_\gamma \subset S_\beta$). We still denote here the objects associated with the shifted block by $\mathbf{x}_j, \mathbf{x}^o_\gamma, \mathcal{D}_\gamma$ and use the splitting (3.3) of $\mathbf{G}(\mathbf{x}_j - \mathbf{y})$ into the free-space $\mathbf{G}_0(\mathbf{x}_j - \mathbf{y})$ and far-field $\mathbf{G}_1(\mathbf{x}_j - \mathbf{y})$ parts. If $\mathcal{D}_\delta \cap \mathcal{D}_\gamma = \emptyset$, the free-space contribution of block $\mathcal{B}_\gamma$ to the sum (3.8a) can be evaluated at $\mathbf{y} \in \mathcal{B}_\delta$ by first reexpanding (3.12) at $\mathbf{x}^o_\delta$ in Lamb's regular form [32]

$$\sum_{n=1}^{\infty} \left[\nabla \times (\mathbf{R}_\delta \chi_n) + \nabla \Phi_n + \frac{(n+3)\mathbf{R}^2_\delta \nabla p_n}{2(n+1)(2n+3)} - \frac{np_n\mathbf{R}_\delta}{(n+1)(2n+3)}\right], \tag{3.17}$$

where the differentiations are with respect to $\mathbf{R}_\delta = \mathbf{y} - \mathbf{x}^o_\delta$, and the positive-order solid harmonics are of the forms

$$p_n(\mathbf{R}_\delta) = \sum_{m=-n}^{n} A_{n,m} \left(\frac{R_\delta}{d^o_\delta}\right)^n Y_{n,m}(\mathbf{R}_\delta), \qquad A_{n,-m} = (-1)^m \bar{A}_{n,m}, \tag{3.18a}$$

$$\Phi_n(\boldsymbol{R}_\delta) = \sum_{m=-n}^{n} B_{n,m} \left(\frac{R_\delta}{d_\delta^o}\right)^n Y_{n,m}(\boldsymbol{R}_\delta), \qquad B_{n,-m} = (-1)^m \bar{B}_{n,m}, \qquad (3.18b)$$

$$\chi_n(\boldsymbol{R}_\delta) = \sum_{m=-n}^{n} C_{n,m} \left(\frac{R_\delta}{d_\delta^o}\right)^n Y_{n,m}(\boldsymbol{R}_\delta), \qquad C_{n,-m} = (-1)^m \bar{C}_{n,m}. \qquad (3.18c)$$

Generally, the larger the gap between $\mathcal{D}_\delta$ and $\mathcal{D}_\gamma$, the fewer terms both in (3.12) and (3.17) suffice for this reexpansion. For a given precision ε, close-to-optimum truncation bounds $v_{nf}(\delta, \gamma)$ and $n_{nf}(\delta, \gamma)$ are constructed (Subsection 3.5), to limit the reexpansion from (3.12) to (3.17) by $v \leq v_{nf} + 1$ and $n \leq n_{nf} + 1$. When $\|\boldsymbol{x}_\delta^o - \boldsymbol{x}_\gamma^o\| \gg \max(d_\delta^o, d_\gamma^o)$, the bounds v_{nf} and n_{nf} approach 3 (this limitation being imposed to ensure absolute convergence of remote block contributions in the "thermodynamical" limit $N \to \infty$, as discussed in Subsection 3.5), while $v_{nf}, n_{nf} \to \infty$ for $\|\boldsymbol{x}_\delta^o - \boldsymbol{x}_\gamma^o\| \to d_\delta^o + d_\gamma^o$ or $\varepsilon \to 0$. The reexpansion from (3.12) to (3.17) is divergent for overlapping shells $\mathcal{D}_\delta$ and $\mathcal{D}_\gamma$ and is also avoided for a small clearance between $\mathcal{D}_\delta$ and $\mathcal{D}_\gamma$ because of slow convergence. Our code has a threshold k_o (optimal values being $\leq O(M^{1/2})$, as found experimentally in Section 6). The shifted block $\mathcal{B}_\gamma$ is called "sufficiently separated" from block $\mathcal{B}_\delta$ (which is denoted by $\mathcal{B}_\gamma \,|\, \mathcal{B}_\delta$), if their shells $\mathcal{D}_\delta$ and $\mathcal{D}_\gamma$ have enough clearance, so that $v_{nf}(\delta, \gamma) \leq k_o$ and $n_{nf}(\delta, \gamma) \leq k_o$. Note that the relation $\mathcal{B}_\gamma \,|\, \mathcal{B}_\delta$ may be asymmetric and also depends on k_o and ε; as $k_o \to \infty$, every $\mathcal{B}_\gamma$ with $\mathcal{D}_\delta \cap \mathcal{D}_\gamma = \emptyset$ becomes sufficiently separated from $\mathcal{B}_\delta$. Reexpansion from (3.12) to (3.17) is performed only for $\mathcal{B}_\gamma \,|\, \mathcal{B}_\delta$ using a fast $O(v_{nf}^3, n_{nf}^3)$-algorithm of Subsection 3.3 with Wigner functions, and the coefficients $A_{n,m}$, $B_{n,m}$ and $C_{n,m}$ in (3.18) are accumulated as the contributions from all blocks $\mathcal{B}_\gamma$ with $\mathcal{B}_\gamma \,|\, \mathcal{B}_\delta$. It is advantageous that the bounds v_{nf} and n_{nf} are small, except for close pairs $(\mathcal{B}_\delta, \mathcal{B}_\gamma)$.

Before pointwise calculation of the cumulative series (3.17) for all collocation nodes $\boldsymbol{y} \in \mathcal{B}_\delta$, (3.17) is transformed to a more efficient form

$$\frac{1}{2}\boldsymbol{R}_\delta \sum_{n=1}^{\infty} p_n(\boldsymbol{R}_\delta) + \sum_{n=0}^{\infty} \sum_{m=-n}^{n} \boldsymbol{H}_{n,m} \left(\frac{R_\delta}{d_\delta^o}\right)^n Y_{n,m}(\boldsymbol{R}_\delta). \qquad (3.19)$$

The vector coefficients $\boldsymbol{H}_{n,m}$ are expressed in terms of cumulative harmonic coefficients $A_{k,\mu}$, $B_{k,\mu}$, and $C_{k,\mu}$ in (3.18) by recurrent relations for spherical harmonics (Appendix B). Using (3.19) in place of (3.17) greatly reduces the number of operations. The truncation bound for (3.19) is

$$n \leq \max_{\gamma : \mathcal{B}_\gamma | \mathcal{B}_\delta} n_{nf}(\delta, \gamma) + 1, \qquad (3.20)$$

which is typically $n \leq k_o + 1$, except for low-concentrated systems.

If a shifted block $\mathcal{B}_\gamma$ is not sufficiently separated from $\mathcal{B}_\delta$, the LHS of (3.12) is calculated for $\boldsymbol{y} \in \mathcal{B}_\delta$ either as Lamb's singular series (3.12), or by standard point-to-point summations. To this end, another economical bound $v_{nf}^*(\boldsymbol{y}, \gamma)$ is constructed (Subsection 3.5), to limit the summation in the RHS of (3.12) (and in (3.21) below) by $v \leq v_{nf}^*(\boldsymbol{y}, \gamma) + 1$ for $\|\boldsymbol{y} - \boldsymbol{x}_\gamma^o\| > d_\gamma^o$. If $\boldsymbol{y}$ is "well outside" $\mathcal{D}_\gamma$ for a given precision ε (which is denoted by $\boldsymbol{y} \,|\, \mathcal{B}_\gamma$), so that $v_{nf}^*(\boldsymbol{y}, \gamma) \leq k_o$, Lamb's series (3.12) is used. The latter operation is considerably optimized by transforming the RHS of (3.12) to

$$\frac{1}{2}\boldsymbol{R}_\gamma \sum_{v=1}^{\infty} p_{-(v+1)}(\boldsymbol{R}_\gamma) + \sum_{v=0}^{\infty} \sum_{m=-v}^{v} \boldsymbol{F}_{v,m} \left(\frac{d_\gamma^o}{R_\gamma}\right)^{v+1} Y_{v,m}(\boldsymbol{R}_\gamma). \qquad (3.21)$$

The vector coefficients $\boldsymbol{F}_{v,m}$ for $v \leq v_{\max}^-(\gamma, \varepsilon) + 1$ are calculated via $A_{-(k+1),\mu}^{(\gamma)}$, $B_{-(k+1),\mu}^{(\gamma)}$, and $C_{-(k+1),\mu}^{(\gamma)}$ (Appendix B) for every block $\mathcal{B}_\gamma$, before the search is made for $\mathcal{B}_\delta$ and $\boldsymbol{y} \in \mathcal{B}_\delta$ for which (3.21) is necessary; there is no need, therefore, to store the coefficients $\boldsymbol{F}_{v,m}$ for all blocks $\mathcal{B}_\gamma$ simultaneously. Only in the rare cases, when $\boldsymbol{y}$ is inside $\mathcal{D}_\gamma$, or is outside but too close to $\mathcal{D}_\gamma$, so that $v_{nf}^*(\boldsymbol{y}, \gamma) > k_o$, should we use standard point-to-point summations (3.12).

The above scheme for near-field interactions also sets the limit $v_{\max}^- + 1$ on v in (3.12)–(3.13) for a given precision. In particular,

$$v_{\max}^-(\gamma) = \max\left\{ \max_{\delta:\mathcal{B}_\gamma|\mathcal{B}_\delta} v_{nf}(\delta, \gamma), \quad \max_{y:v_{nf}^*\leq k_o} v_{nf}^*(\boldsymbol{y}, \gamma)\right\}, \tag{3.22}$$

which is typically $v_{\max}^- = k_o$, except for at low concentrations.

The far-field contribution of the shifted block $\mathcal{B}_\gamma$, i.e., the LHS of (3.12) with $\boldsymbol{G}_1(\boldsymbol{x}_j - \boldsymbol{y})$ instead of $\boldsymbol{G}_o(\boldsymbol{x}_j - \boldsymbol{y})$, can be evaluated at $\boldsymbol{y} \in \mathcal{B}_\delta$ using a special form of Taylor double series in powers of $\boldsymbol{x}_j - \boldsymbol{x}_\gamma^o$ and $\boldsymbol{y} - \boldsymbol{x}_\delta^o$ for Stokes flows,

$$\sum_{\boldsymbol{x}_j \in \mathcal{B}_\gamma} W_k(\boldsymbol{x}_j)(G_1)_{kl}(\boldsymbol{x}_j - \boldsymbol{y}) = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} Z_{n,m}\left(\boldsymbol{y} - \boldsymbol{x}_\delta^o\right) \left\{ \sum_{v=0}^{\infty} \sum_{\mu=-v}^{v} D_{v,\mu,k}^{(\gamma)} L_{n+v,k,l}^{m+\mu} \right.$$
$$\left. + \left(\boldsymbol{y} - \boldsymbol{x}_\delta^o + \boldsymbol{x}_\delta^o - \boldsymbol{x}_\gamma^o\right)_l D_{v,\mu,k}^{(\gamma)} M_{n+v,k}^{m+\mu} - E_{v,\mu,k,l}^{(\gamma)} M_{n+v,k}^{m+\mu} \right\}. \tag{3.23}$$

The detailed form of (3.23) is given in Subsection 3.4, the coefficients $L_{n,k,l}^m$ and $M_{n,k}^m$ being related to high-order derivatives of $(G_1)_{kl}(\boldsymbol{r})$ and the associated pressure $q_1^{(k)}(\boldsymbol{r})$ at $\boldsymbol{r} = \boldsymbol{x}_\delta^o - \boldsymbol{x}_\gamma^o$. The number N of drops in the cell is assumed to be not too small, so that

$$\zeta_{\delta\gamma} = \left[1 - 2\max\left\{\left|x_\delta^o - x_\gamma^o\right|_1, \left|x_\delta^o - x_\gamma^o\right|_2, \left|x_\delta^o - x_\gamma^o\right|_3\right\} + \left(\boldsymbol{x}_\delta^o - \boldsymbol{x}_\gamma^o\right)^2\right]^{1/2} > d_\delta^o + d_\gamma^o. \tag{3.24}$$

The condition (3.24) guarantees that the shell $\mathcal{D}_\delta$ does not overlap any periodic images of $\mathcal{D}_\gamma$, except possibly $\mathcal{D}_\gamma$ itself (recall that $|x_\delta^o - x_\gamma^o|_k \leq 1/2$ is assumed), and so the series (3.23) is convergent. Moreover, for large systems ($N \geq O(100)$), the convergence of (3.23) is fast, and typically only low-order terms suffice. Three more truncation bounds are constructed (Subsection 3.5), $v_{ff}(\delta, \gamma)$, $n_{ff}(\delta, \gamma)$, and $v_{ff}^*(\delta, \gamma)$, to limit the summation in (3.23) by

$$v \leq v_{ff}, \qquad n \leq n_{ff}, \qquad v + n \leq v_{ff}^*, \tag{3.25}$$

for a given precision ε. The form (3.23) is used to accumulate contributions from all blocks $\mathcal{B}_\gamma \not\subset S_\alpha \ni \boldsymbol{y}$, with $L_{n+v,k,l}^{m+\mu}$ and $M_{n+v,k}^{m+\mu}$ calculated by the tables of high-order derivatives of $\boldsymbol{G}_1$-related functions (Subsection 3.5). The cumulative expression, as a function of $\boldsymbol{R}_\delta$, has the same structure as (3.19) for the near field, and the two are combined before fast pointwise calculations for all collocation nodes $\boldsymbol{y} \in \mathcal{B}_\delta$ are made. The truncations (3.25) also set the limit on v in precalculating far-field moments (3.15) for every block $\mathcal{B}_\gamma$.

Our far-field scheme is clearly $O(N^2)$-intensive, but the coefficient before N^2 for $N \gg 1$ is very small, since for $N \gg 1$ the bounds v_{ff}, n_{ff}, and v_{ff}^* are $O(1)$ and are practically

independent of M; in the present applications, the total cost is strongly dominated by near-field interactions between close blocks, even for quite large systems. For this reason, more involved schemes [12, 19] for remote interactions designed to eliminate $O(N^2)$ scaling were not incorporated in our algorithm.

Partitioning of elongated drops into compact blocks is particularly helpful in the far-field part for moderately large N. Without this option, the convergence of (3.23) may be much slower, and we had difficulties in tabulating a sufficient number of high-order derivatives of G_1-related functions. For self-interactions, however, partitioning into blocks is not used, since these derivatives are required only at $r = 0$, and can be calculated to high order in case of slow convergence.

The following Subsections 3.2–3.5 elaborate on the multipole details of our code. Additional aspects of the algorithm are discussed in Sections 4 and 5.

3.2. Fast Calculation of Singular Near-Field Moments

A large portion of the total computational load is spent on generating (3.12)–(3.13) and, when $\lambda \neq 1$, similar expansions for the double-layer,

$$\sum_{x_j \in \mathcal{B}_\gamma} Q(x_j) \cdot \tau_0(x_j - y) \cdot W(x_j) = \sum_{\nu=1}^{\infty} \left[\nabla \times \left(R_\gamma \tilde{\chi}_{-(\nu+1)} \right) + \nabla \tilde{\Phi}_{-(\nu+1)} \right.$$
$$\left. - \frac{(\nu-2) R_\gamma^2 \nabla \tilde{p}_{-(\nu+1)}}{2\nu(2\nu-1)} + \frac{(\nu+1) \tilde{p}_{-(\nu+1)} R_\gamma}{\nu(2\nu-1)} \right] + \nabla \tilde{\Phi}_{-1},$$

$$(3.26)$$

on every iteration. The harmonics $\tilde{p}_{-(\nu+1)}$, $\tilde{\Phi}_{-(\nu+1)}$, and $\tilde{\chi}_{-(\nu+1)}$ have the form (3.13a)–(3.13c), with $\tilde{A}^{(\gamma)}_{-(\nu+1),m}$, etc., instead of $\tilde{A}^{(\gamma)}_{-(\nu+1),m}$, etc.; the source term $\nabla \tilde{\Phi}_{-1}$ is absent for the single layer (3.12). This task is a particular case of translation of Stokes singularities, and the general relations of Sangani and Mo [11] could be used for this purpose. However, an alternative, rotation-based technique is used herein, since it considerably reduces the number of operations in generating (3.12) and (3.26), and greatly simplifies the algebra. The idea of our approach is to generate the expansions

$$G_0(x_j - y) \cdot W(x_j)$$
$$= \sum_{\nu=1}^{\infty} \left[\nabla \times \left(R_\gamma \chi^{(j)}_{-(\nu+1)} \right) + \nabla \Phi^{(j)}_{-(\nu+1)} - \frac{(\nu-2) R_\gamma^2 \nabla p^{(j)}_{-(\nu+1)}}{2\nu(2\nu-1)} + \frac{(\nu+1) p^{(j)}_{-(\nu+1)} R_\gamma}{\nu(2\nu-1)} \right]$$

$$(3.27)$$

or

$$Q(x_j) \cdot \tau_0(x_j - y) \cdot W(x_j) = \sum_{\nu=1}^{\infty} \left[\nabla \times \left(R_\gamma \tilde{\chi}^{(j)}_{-(\nu+1)} \right) + \nabla \tilde{\Phi}^{(j)}_{-(\nu+1)} \right.$$
$$\left. - \frac{(\nu-2) R_\gamma^2 \nabla \tilde{p}^{(j)}_{-(\nu+1)}}{2\nu(2\nu-1)} + \frac{(\nu+1) \tilde{p}^{(j)}_{-(\nu+1)} R_\gamma}{\nu(2\nu-1)} \right] + \nabla \tilde{\Phi}^{(j)}_{-1}$$

$$(3.28)$$

for a single node x_j, using a preferred, "intrinsic" coordinate system (x'_1, x'_2, x'_3) with the x'_3-axis along $\rho = x_j - x^o_\gamma$ (Fig. 3), then transform harmonics $p^{(j)}_{-(\nu+1)}(R_\gamma)$, $\Phi^{(j)}_{-(\nu+1)}(R_\gamma)$,
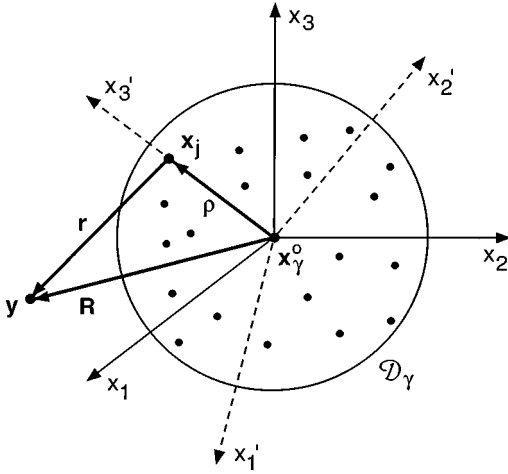
FIG. 3. The rotation-based scheme for calculating near-field moments.

and $\chi^{(j)}_{-(\nu+1)}(\boldsymbol{R}_\gamma)$ (or $\tilde{p}^{(j)}_{-(\nu+1)}$, $\tilde{\Phi}^{(j)}_{-(\nu+1)}$, and $\tilde{\chi}^{(j)}_{-(\nu+1)}$) into the old coordinates (x_1, x_2, x_3) (the same for all $\boldsymbol{x}_j$), and accumulate contributions from all nodes $\boldsymbol{x}_j \in \mathcal{B}_\gamma$ in this manner. This procedure is reminiscent of using rotational transformations in conductivity simulations [16, 17]. An intrinsic coordinate system is not unique, but the freedom of rotation about the x_3'-axis does not affect the result.

In deriving (3.27)–(3.28) in the intrinsic coordinates, we omit primes and identify $\boldsymbol{W}(\boldsymbol{x}_j)$, $\boldsymbol{Q}(\boldsymbol{x}_j)$, and $\boldsymbol{R}_\gamma = \boldsymbol{y} - \boldsymbol{x}_\gamma^o$ with $\boldsymbol{W}, \boldsymbol{Q}$, and $\boldsymbol{R}$, respectively, for brevity. The pressure $p^{(j)}(\boldsymbol{y})$ associated with the Stokeslet $\boldsymbol{\omega}(\boldsymbol{y}) = \boldsymbol{G}_0(\boldsymbol{x}_j - \boldsymbol{y}) \cdot \boldsymbol{W}$ is

$$p^{(j)}(\boldsymbol{y}) = \frac{1}{4\pi} \boldsymbol{W} \cdot \nabla_y \left(\frac{1}{r}\right), \qquad \boldsymbol{r} = \boldsymbol{y} - \boldsymbol{x}_j. \tag{3.29}$$

The addition theorem for spherical harmonics greatly simplifies in the intrinsic coordinates:

$$\frac{1}{r} = \sum_{n=0}^{\infty} \left(\frac{4\pi}{2n+1}\right)^{1/2} \frac{Y_{n,0}(\boldsymbol{R})}{R^{n+1}} \rho^n. \tag{3.30}$$

Using the well-known relations

$$(D \pm i D_2)\frac{Y_{n,m}(\boldsymbol{R})}{R^{n+1}} = \mp \left[\frac{(2n+1)(n \pm m+1)(n \pm m+2)}{2n+3}\right]^{1/2} \frac{Y_{n+1,m\pm 1}(\boldsymbol{R})}{R^{n+2}},$$

$$D_3 \frac{Y_{n,m}(\boldsymbol{R})}{R^{n+1}} = -\left[\frac{(2n+1)(n-m+1)(n+m+1)}{2n+3}\right]^{1/2} \frac{Y_{n+1,m}(\boldsymbol{R})}{R^{n+2}} \tag{3.31}$$

(where $D_i = \partial/\partial R_i$ is the Cartesian partial derivative) at $m = 0$ and differentiating (3.30), the pressure (3.29) can be expanded into spherical harmonics,

$$p^{(j)}(\boldsymbol{y}) = \sum_{\nu=1}^{\infty} p^{(j)}_{-(\nu+1)}(\boldsymbol{R}) = \sum_{n=1}^{\infty} \frac{\rho^{n-1}}{4[\pi(2n+1)]^{1/2}R^{n+1}} \{[n(n+1)]^{1/2}[(W_1 + i W_2)Y_{n,-1}(\boldsymbol{R})$$

$$- (W_1 - i W_2)Y_{n,1}(\boldsymbol{R})] - 2n W_3 Y_{n,0}(\boldsymbol{R})\}, \tag{3.32}$$

which immediately gives the coefficients $A^{(j)}_{-(v+1),m}$ in

$$p^{(j)}_{-(v+1)}(\boldsymbol{R}) = \sum_{m=-1}^{1} A^{(j)}_{-(v+1),m} \left(\frac{d_\gamma^o}{R} \right)^{v+1} Y_{v,m}(\boldsymbol{R}). \tag{3.33}$$

To find $\Phi^{(j)}_{-(v+1)}(\boldsymbol{R})$, we use the general relation [32]

$$\boldsymbol{\omega}(\boldsymbol{y}) \cdot \boldsymbol{R} = \sum_{v=1}^{\infty} \left[\frac{(v+1)}{2(2v-1)} R^2 p^{(j)}_{-(v+1)} - (v+1)\Phi^{(j)}_{-(v+1)} \right]. \tag{3.34}$$

By (3.3b), the LHS of (3.34) can be written as

$$-\frac{1}{8\pi} \left[(\boldsymbol{W} \cdot \boldsymbol{R}) \left(\frac{1}{r} - \boldsymbol{R} \cdot \nabla_y \frac{1}{r} \right) + (\boldsymbol{W} \cdot \boldsymbol{\rho}) \, \boldsymbol{R} \cdot \nabla_y \frac{1}{r} \right]. \tag{3.35}$$

To expand (3.35) into spherical harmonics, additional recurrent relations are needed:

$$\frac{(R_1 \pm i R_2) Y_{n,m}(\boldsymbol{R})}{R} = \pm \left[\frac{(n \pm m + 1)(n \pm m + 2)}{(2n+1)(2n+3)} \right]^{1/2} Y_{n+1,m\pm1}(\boldsymbol{R})$$

$$\mp \left[\frac{(n \mp m - 1)(n \mp m)}{(2n+1)(2n-1)} \right]^{1/2} Y_{n-1,m\pm1}(\boldsymbol{R}), \tag{3.36a}$$

$$\frac{R_3}{R} Y_{n,m}(\boldsymbol{R}) = \left[\frac{(n-m+1)(n+m+1)}{(2n+1)(2n+3)} \right]^{1/2} Y_{n+1,m}(\boldsymbol{R})$$

$$+ \left[\frac{(n-m)(n+m)}{(2n-1)(2n+1)} \right]^{1/2} Y_{n-1,m}(\boldsymbol{R}). \tag{3.36b}$$

Substituting (3.30) into (3.35), and using the homogenity of solid harmonics [32] and (3.36) at $m=0$, one obtains

$$\boldsymbol{\omega}(\boldsymbol{y}) \cdot \boldsymbol{R} = \sum_{n=1}^{\infty} \frac{(n+3)\rho^{n+1}}{8(2n+3)[\pi(2n+1)]^{1/2} R^{n+1}}$$

$$\times \left\{ [n(n+1)]^{1/2}[(W_1 - i W_2) Y_{n,1}(\boldsymbol{R}) - (W_1 + i W_2) Y_{n,-1}(\boldsymbol{R})] \right.$$

$$\left. - 2(n+1) W_3 Y_{n,0}(\boldsymbol{R}) \right\} + \sum_{n=1}^{\infty} \frac{(n+1)(\boldsymbol{W} \cdot \boldsymbol{\rho}) \rho^n}{4[\pi(2n+1)]^{1/2}} \frac{Y_{n,0}(\boldsymbol{R})}{R^{n+1}} + \cdots, \tag{3.37}$$

where we have omitted the terms of the type R^2 times a solid harmonic of negative order (those terms match $R^2 p^{(j)}_{-(v+1)}$-terms in (3.34) and would only be needed to determine $p^{(j)}_{-(v+1)}$ already provided by (3.32)). Comparing the Φ-part of (3.34) with (3.37) gives the coefficients $B^{(j)}_{-(v+1),m}$ in

$$\Phi^{(j)}_{-(v+1)}(\boldsymbol{R}) = \sum_{m=-1}^{1} B^{(j)}_{-(v+1),m} \left(\frac{d_\gamma^o}{R} \right)^{v+1} Y_{v,m}(\boldsymbol{R}). \tag{3.38}$$

The calculation of $\chi_{-(v+1)}^{(j)}$ is even simpler, through the relation [32]

$$[\nabla \times \boldsymbol{\omega}(\boldsymbol{y})] \cdot \boldsymbol{R} = \sum_{n=1}^{\infty} n(n+1) \chi_{-(n+1)}^{(j)}. \tag{3.39}$$

As follows from (3.3b),

$$[\nabla \times \boldsymbol{\omega}(\boldsymbol{y})] \cdot \boldsymbol{R} = -\frac{1}{4\pi} \boldsymbol{W} \cdot \left(\boldsymbol{R} \times \nabla_y \frac{1}{r} \right). \tag{3.40}$$

Substituting (3.30) into (3.40) and using the relations

$$(R_2 D_3 - R_3 D_2) \frac{Y_{n,m}(\boldsymbol{R})}{R^{n+1}} = -\frac{i}{2R^{n+1}} \{ [(n-m)(n+m+1)]^{1/2} Y_{n,m+1}(\boldsymbol{R})$$
$$+ [(n-m+1)(n+m)]^{1/2} Y_{n,m-1}(\boldsymbol{R}) \},$$

$$(R_3 D_1 - R_1 D_3) \frac{Y_{n,m}(\boldsymbol{R})}{R^{n+1}} = \frac{1}{2R^{n+1}} \{ -[(n-m)(n+m+1)]^{1/2} Y_{n,m+1}(\boldsymbol{R}) \tag{3.41}$$
$$+ [(n-m+1)(n+m)]^{1/2} Y_{n,m-1}(\boldsymbol{R}) \},$$

$$(R_1 D_2 - R_2 D_1) \frac{Y_{n,m}(\boldsymbol{R})}{R^{n+1}} = \frac{im}{R^{n+1}} Y_{n,m}(\boldsymbol{R})$$

at $m = 0$ allows us to represent $[\nabla \times \boldsymbol{\omega}(\boldsymbol{y})] \cdot \boldsymbol{R}$ as

$$\sum_{n=1}^{\infty} \left[\frac{n(n+1)}{\pi(2n+1)} \right]^{1/2} \frac{\rho^n}{4R^{n+1}} [(iW_1 + W_2) Y_{n,1}(\boldsymbol{R}) + (iW_1 - W_2) Y_{n,-1}(\boldsymbol{R})]. \tag{3.42}$$

Comparing (3.39) and (3.42) gives the coefficients $C_{-(v+1),m}^{(j)}$ in

$$\chi_{-(v+1)}^{(j)}(\boldsymbol{R}) = \sum_{m=-1,1} C_{-(v+1),m}^{(j)} \left(\frac{d_\gamma^o}{R} \right)^{v+1} Y_{v,m}(\boldsymbol{R}). \tag{3.43}$$

To expand the stresslet $\boldsymbol{\Psi}(\boldsymbol{y}) = \boldsymbol{Q} \cdot \boldsymbol{\tau}_0(\boldsymbol{x}_j - \boldsymbol{y}) \cdot \boldsymbol{W}$ in the form (3.28), we switch to the weights (3.7) and generate (3.27) first for the Stokeslet $\boldsymbol{\omega}(\boldsymbol{y}) = \boldsymbol{G}_0(\boldsymbol{x}_j - \boldsymbol{y}) \cdot \boldsymbol{W}$, as described above. Since $\nabla^2 \boldsymbol{\Psi}(\boldsymbol{y}) = -\nabla[2\boldsymbol{Q} \cdot \nabla p^{(j)}(\boldsymbol{y})]$, the pressure $\tilde{p}^{(j)}(\boldsymbol{y}) = -2\boldsymbol{Q} \cdot \nabla p^{(j)}(\boldsymbol{y})$ associated with $\boldsymbol{\Psi}(\boldsymbol{y})$ is easily expanded into spherical harmonics, using (3.33) and (3.31), which gives the coefficients $\tilde{A}_{-(v+1),m}^{(j)}$ in

$$\tilde{p}_{-(v+1)}^{(j)}(\boldsymbol{R}) = \sum_{m=-2}^{2} \tilde{A}_{-(v+1),m}^{(j)} \left(\frac{d_\gamma^o}{R} \right)^{v+1} Y_{v,m}(\boldsymbol{R}). \tag{3.44}$$

To find $\tilde{\Phi}_{-(v+1)}^{(j)}(\boldsymbol{R})$, we use again the property [32]

$$\boldsymbol{\Psi}(\boldsymbol{y}) \cdot \boldsymbol{R} = \sum_{v=0}^{\infty} \left[\frac{(v+1)}{2(2v-1)} R^2 \tilde{p}_{-(v+1)}^{(j)} - (v+1) \tilde{\Phi}_{-(v+1)}^{(j)} \right] \qquad \left(\tilde{p}_{-1}^{(j)} = \tilde{p}_{-2}^{(j)} = 0 \right). \tag{3.45}$$

Note that $\boldsymbol{\Psi}(\boldsymbol{y}) \cdot \boldsymbol{R} = -R\boldsymbol{Q} \cdot \boldsymbol{\tau}_R$, where $\boldsymbol{\tau}_R$ is the stress vector on the sphere of radius R centered at $\boldsymbol{x}_\gamma^o$ for the Stokeslet $\boldsymbol{\omega}(\boldsymbol{y})$. Since $\boldsymbol{\omega}(\boldsymbol{y})$ is expanded in Lamb's form (3.27), a

ready expression for τ_R can be used [32], resulting in

$$\Psi(\boldsymbol{y}) \cdot \boldsymbol{R} = \boldsymbol{Q} \cdot \sum_{n=1}^{\infty} \left[(n+2)\nabla\chi^{(j)}_{-(n+1)} \times \boldsymbol{R} + 2(n+2)\,\nabla\Phi^{(j)}_{-(n+1)} \right.$$
$$\left. + \frac{(2n^2+1)\boldsymbol{R}p^{(j)}_{-(n+1)}}{n(2n-1)} - \frac{(n+1)(n-1)\boldsymbol{R}^2\nabla p^{(j)}_{-(n+1)}}{n(2n-1)} \right]. \qquad (3.46)$$

Substituting (3.33), (3.38), and (3.43) into (3.46), and using the relations (3.41), (3.31), and (3.36) allows us to represent (3.46) in the form (3.45) and calculate the coefficients $\tilde{B}^{(j)}_{-(\nu+1),m}$ in

$$\tilde{\Phi}^{(j)}_{-(\nu+1)}(\boldsymbol{R}) = \sum_{m=-2}^{2} \tilde{B}^{(j)}_{-(\nu+1),m} \left(\frac{d^o_\gamma}{R} \right)^{\nu+1} Y_{\nu,m}(\boldsymbol{R}). \qquad (3.47)$$

The coefficients $\tilde{C}^{(j)}_{-(\nu+1),m}$ are determined from (3.39), with $\Psi(\boldsymbol{y})$ and $\tilde{\chi}^{(j)}_{-(n+1)}$ in place of $\boldsymbol{\omega}(\boldsymbol{y})$ and $\chi^{(j)}_{-(n+1)}$, respectively. Using

$$\Psi(\boldsymbol{y}) = p^{(j)}(\boldsymbol{y})\boldsymbol{Q} - \nabla[\boldsymbol{\omega}(\boldsymbol{y}) \cdot \boldsymbol{Q}] - (\boldsymbol{Q}\nabla)\,\boldsymbol{\omega}(\boldsymbol{y}), \qquad (3.48)$$

one can derive

$$(\nabla \times \boldsymbol{\Psi}) \cdot \boldsymbol{R} = -\boldsymbol{Q} \cdot \left(\nabla p^{(j)} \times \boldsymbol{R} \right) - \boldsymbol{Q} \cdot \nabla[(\nabla \times \boldsymbol{\omega}) \cdot \boldsymbol{R}] + \boldsymbol{Q} \cdot (\nabla \times \boldsymbol{\omega}). \qquad (3.49)$$

Since $\boldsymbol{\omega}(\boldsymbol{y})$ is expanded as Lamb's series (3.27), a convenient expression [11] can be used:

$$\nabla \times \boldsymbol{\omega}(\boldsymbol{y}) = \sum_{n=1}^{\infty} \left[-n\nabla\chi^{(j)}_{-(n+1)} - \frac{1}{n}\boldsymbol{R} \times \nabla p^{(j)}_{-(n+1)} \right]. \qquad (3.50)$$

Substituting (3.50) into (3.49) and using (3.39) gives a compact relation:

$$\sum_{\nu=1}^{\infty} \nu(\nu+1)\tilde{\chi}^{(j)}_{-(\nu+1)} = -\sum_{n=1}^{\infty} \left[\frac{(n-1)}{n}\boldsymbol{Q} \cdot \left(\nabla p^{(j)}_{-(n+1)} \times \boldsymbol{R} \right) + n(n+2)\boldsymbol{Q} \cdot \nabla\chi^{(j)}_{-(n+1)} \right]. \qquad (3.51)$$

Finally, by (3.33), (3.43), and the recurrent relations (3.31), (3.41), the RHS of (3.51) is expanded into spherical harmonics, which gives the coefficients $\tilde{C}^{(j)}_{-(\nu+1),m}$ in

$$\tilde{\chi}^{(j)}_{-(\nu+1)}(\boldsymbol{R}) = \sum_{m=-2}^{2} \tilde{C}^{(j)}_{-(\nu+1),m} \left(\frac{d^o_\gamma}{R} \right)^{\nu+1} Y_{\nu,m}(\boldsymbol{R}). \qquad (3.52)$$

It remains to subject the sparse matrices $A^{(j)}_{-(\nu+1),m}$, $\tilde{A}^{(j)}_{-(\nu+1),m}$, $B^{(j)}_{-(\nu+1),m}$, $\tilde{B}^{(j)}_{-(\nu+1),m}$, $C^{(j)}_{-(\nu+1),m}$, and $\tilde{C}^{(j)}_{-(\nu+1),m}$ to rotational transformation into the original coordinate system (more detail on this operation is given in Subsection 3.3) and accumulate contributions from all nodes $\boldsymbol{x}_j \in \mathcal{B}_\gamma$ in this manner, to obtain singular near-field moments $A^{(\gamma)}_{-(\nu+1),m}$, $\tilde{A}^{(\gamma)}_{-(\nu+1),m}$, etc. Although this rotation-based technique still has an $O[(\nu^-_{\max})^2|\mathcal{B}_\gamma|]$ computational cost (where $|\mathcal{B}_\gamma|$ is the number of nodes $\boldsymbol{x}_j \in \mathcal{B}_\gamma$), it reduces the number of operations compared to direct singularity translations, besides greatly simplifying the algebra. Our fully optimized rotation-based routine takes $2.9 \times 10^{-7}(\nu^-_{\max})^2|\mathcal{B}_\gamma|$ and $4.3 \times 10^{-7}(\nu^-_{\max})^2|\mathcal{B}_\gamma|$ seconds of CPU time to generate the single-layer (3.12) and double-layer (3.26) expansions, respectively, for $\nu^-_{\max} \gg 1$ on a DEC 500au workstation. These times are 2.3-fold and

2.8-fold faster than for the optimized routine without rotations which we used for comparisons. On the other hand, when solving the Laplace equation in a multiparticle system by a similar hybrid of multipole and boundary-integral techniques, it would not be advantageous to use rotations for calculating singular near-field moments.

3.3. *Fast Algorithm for Regular Near-Field Moments*

Another principal task, as outlined in Subsection 3.1, is to accumulate Lamb's singular contributions (3.12) or (3.26) from all blocks $\mathcal{B}_\gamma \mid \mathcal{B}_\delta$ and represent the result as Lamb's regular expansion (3.17)–(3.18) centered at x_δ^o. Our code in this part closely follows the idea of the "rotational algorithm for the near-field operator" in conductivity simulations [16], with substantial use of Wigner functions. Namely, a temporary, "axial" basis (x_1', x_2', x_3') is introduced for blocks $\mathcal{B}_\delta$ and $\mathcal{B}_\gamma$, with the x_3'-axis along the center-to-center vector $\boldsymbol{R}_{\delta\gamma} = x_\gamma^o - x_\delta^o$ (Fig. 4). The harmonics $p_{-(\nu+1)}$, $\Phi_{-(\nu+1)}$, and $\chi_{-(\nu+1)}$, or $\tilde{p}_{-(\nu+1)}$, $\tilde{\Phi}_{-(\nu+1)}$, and $\tilde{\chi}_{-(\nu+1)}$ in Lamb's singular expansions (3.12) or (3.26) for block $\mathcal{B}_\gamma$ are first transformed into the coordinates (x_1', x_2', x_3') by Wigner functions (see below), then (3.12) or (3.26) is reexpanded at x_δ^o as Lamb's regular series (3.17), the harmonics p_n, Φ_n, and χ_n are transformed back to the original basis (x_1, x_2, x_3) and the contributions from all blocks $\mathcal{B}_\gamma$ with $\mathcal{B}_\gamma \mid \mathcal{B}_\delta$ are added, to produce the cumulative near-field moments $A_{n,m}$, $B_{n,m}$, and $C_{n,m}$. Most importantly, the computational cost of each $\mathcal{B}_\gamma \to \mathcal{B}_\delta$ reexpansion in this scheme is $O(\nu_{nf}^3, n_{nf}^3)$, compared to $O(\nu_{nf}^4, n_{nf}^4)$ for direct reexpansion without rotations. For Laplace interactions, Zinchenko [16] found 3-fold and 6-fold advantages of this scheme over the direct reexpansions at $\nu_{nf} = n_{nf} = 10$ and $\nu_{nf} = n_{nf} = 20$, respectively. For the Stokes problems, even much greater gains are expected.
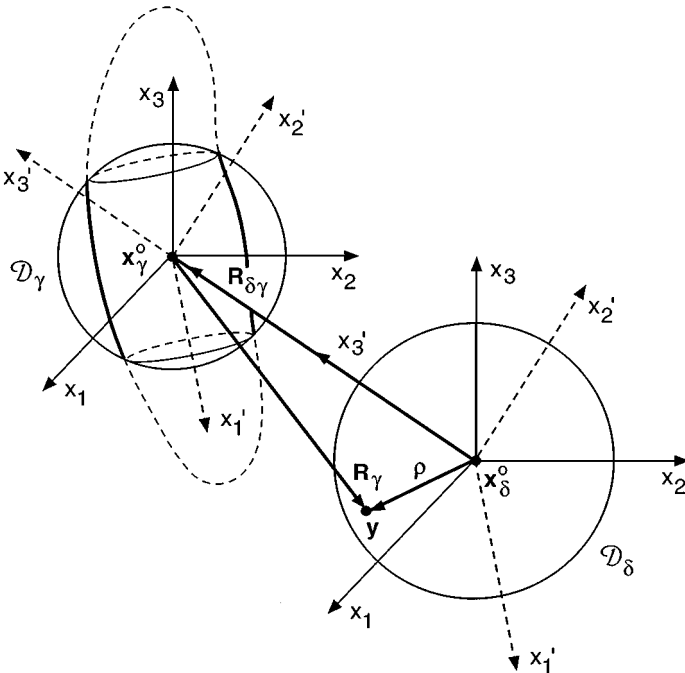

FIG. 4.   The rotation-based scheme for block-to-block reexpansions.

The rotation from the original (x_1, x_2, x_3) to the new (x'_1, x'_2, x'_3) basis is a sequence of two transformations: (i) rotation about the x_3-axis of angle $\chi_{\delta\gamma}$ and (ii) rotation about the new x_1-axis of angle $\Psi_{\delta\gamma}$, where

$$\exp(i\chi_{\delta\gamma}) = -(X_2 + iX_1)/(X_1^2 + X_2^2)^{1/2}, \qquad 0 \leq \chi_{\delta\gamma} < 2\pi,$$
$$\eta = \cos\Psi_{\delta\gamma} = X_3/R_{\delta\gamma}, \qquad 0 \leq \Psi_{\delta\gamma} < \pi, \tag{3.53}$$

and $\boldsymbol{R}_{\delta\gamma} = (X_1, X_2, X_3)$ in the original coordinates (in the degenerate case $X_1 = X_2 = 0$, the value of $\chi_{\delta\gamma}$ is arbitrary). The third Euler angle can be set to zero. Let (r', θ', φ') be spherical coordinates associated with (x'_1, x'_2, x'_3). It follows from the theory of Wigner functions [33] that

$$Y_{\nu,m}(\theta, \varphi) = \sum_{|m'| \leq \nu} \exp(im\chi_{\delta\gamma}) P^\nu_{m,m'}(\Psi_{\delta\gamma}) Y_{\nu,m'}(\theta', \varphi'), \tag{3.54a}$$

$$Y_{\nu,m'}(\theta', \varphi') = \sum_{|m| \leq \nu} (-1)^{m+m'} \exp(-im\chi_{\delta\gamma}) P^\nu_{m',m}(\Psi_{\delta\gamma}) Y_{\nu,m}(\theta, \varphi). \tag{3.54b}$$

The relations (3.54) allow us to transform harmonic coefficients from (x_1, x_2, x_3) to (x'_1, x'_2, x'_3) and back, if the complex coefficients $P^\nu_{m,m'}$ are known. These coefficients obey

$$P^\nu_{m,m'} = P^\nu_{m',m}, \qquad P^\nu_{m,m'} = P^\nu_{-m,-m'} \tag{3.55}$$

and are related to Jacobi polynomials [33], which give different recurrent schemes. Memory limitations do not allow all necessary P-coefficients to be stored, and they are instead calculated on every iteration, with a small additional cost. The fastest way, as noted by Zinchenko [16], is to introduce real X-coefficients,

$$P^\nu_{m,m'} = i^{|m'|-m}(c_{\nu,m'}/c_{\nu,m}) X^\nu_{m,m'}, \qquad |m'| \leq m \leq \nu, \tag{3.56}$$

where

$$c_{\nu,m} = \frac{1}{2^\nu} \left[\frac{(2\nu)!}{(\nu - m)!(\nu + m)!} \right]^{1/2}, \tag{3.57}$$

and use the recurrent scheme

$$X^0_{0,0} = 1, \quad X^\nu_{\nu,\pm\nu} = \frac{\eta \pm 1}{2} X^{\nu-1}_{\nu-1,\pm(\nu-1)} \qquad \text{(for } \nu \geq 1\text{)}, \tag{3.58a}$$

$$X^\nu_{\nu,m'} = \frac{1}{2} \sin\Psi_{\delta\gamma} X^{\nu-1}_{\nu-1,m'} \qquad \text{(for } |m'| \leq \nu - 1\text{)}, \tag{3.58b}$$

$$X^\nu_{m,m'} = \frac{1}{\nu - m} \left[(\nu\eta - m') X^{\nu-1}_{m,m'} - \nu\sin\Psi_{\delta\gamma} X^{\nu-1}_{m+1,m'} \right] \qquad \text{(for } |m'| \leq m \leq \nu - 1\text{)} \tag{3.58c}$$

(omitting the last term in (3.58c) for $m = \nu - 1$). Relation (3.58c) was observed to lose stability [17], but not until ν exceeds 100–115 in double-precision calculations. An alternative, slightly slower scheme was found to be absolutely stable [17]. However, since the maximum values of ν in the present work were about 20 (Section 6), the fastest scheme

(3.58c) could be used safely. Rotational transformations of harmonic coefficients between (x_1, x_2, x_3) and (x'_1, x'_2, x'_3) are further expedited by (28)–(29) of Zinchenko [16].

It remains to discuss how to reexpand Lamb's singular form (3.12) or (3.26) for block $\mathcal{B}_\gamma$ as a regular series (3.17) at x^o_δ, with the harmonics written in the axial basis (x'_1, x'_2, x'_3). It appears that reexpansion relations from Lamb's singular to regular forms have been independently rederived by several authors. Happel and Brenner [32] considered this transformation, but used coordinate systems with polar axes $\theta = 0$ normal to the translation vector, which made the relations too cumbersome and noneconomical. Mo and Sangani [35] developed a more attractive formalism, not limited to a particular choice of coordinate systems, but without using the advantages of axial coordinates. Earlier, Zinchenko [36, 37] gave compact reexpansion formulae in the axial coordinates, but only for azimuthal numbers $m = 1$ and 2. Below, we generalize the relations [36, 37] for arbitrary azimuthal numbers, with a short and simple derivation following the logic of [36]; alternatively, the general, but not simple relations of Mo and Sangani (Appendix I of [35]) could be applied in the axial coordinates for a free-space case to reach the same goal. The transitions from (3.12) and (3.26) to (3.17) are the same, and we only consider (3.12), with the summation range $\nu \geq 0$ and $p_{-1} = \chi_{-1} = 0$, to cover both the single-layer and double-layer cases.

Assuming that $A^{(\gamma)}_{-(\nu+1),m}$, $B^{(\gamma)}_{-(\nu+1),m}$, and $C^{(\gamma)}_{-(\nu+1),m}$ have been transformed to the axial basis (x'_1, x'_2, x'_3), primes are omitted in the following calculations using these coordinates. The generalized addition theorem for spherical harmonics (e.g., [38]) greatly simplifies in the axial coordinates,

$$\left(\frac{d^o_\gamma}{R_\gamma}\right)^{\nu+1} Y_{\nu,m}(\boldsymbol{R}_\gamma) = \sum_{n=|m|}^{\infty} \Gamma^m_{\nu,n} \left(\frac{\rho}{d^o_\delta}\right)^n Y_{n,m}(\boldsymbol{\rho}), \qquad (3.59)$$

where

$$\Gamma^m_{\nu,n} = (-1)^{\nu+m} \left[\frac{2\nu+1}{2n+1}\right]^{1/2} \frac{c_{\nu,m} c_{n,m} c_{n+\nu,n-\nu}}{2c_{n+\nu,0}} \left(\frac{2}{R_{\delta\gamma}}\right)^{\nu+n+1} (d^o_\delta)^n (d^o_\gamma)^{\nu+1} \qquad (3.60)$$

and, for brevity, $\boldsymbol{R}_\delta = \boldsymbol{y} - \boldsymbol{x}^o_\delta = \boldsymbol{\rho}$. The relation (3.59) immediately gives $A_{n,m}$ for the $\mathcal{B}_\gamma \to \mathcal{B}_\delta$ contribution:

$$A_{n,m} = \sum_{\nu=|m|}^{\infty} \Gamma^m_{\nu,n} A^{(\gamma)}_{-(\nu+1),m}. \qquad (3.61)$$

To find $C_{n,m}$, we dot the curl of (3.12) with $\boldsymbol{\rho}$, yielding

$$\sum_{n=1}^{\infty} n(n+1)\chi_n(\boldsymbol{\rho})$$

$$= \sum_{\nu=1}^{\infty} \left\{ -\frac{1}{\nu} R_{\delta\gamma} \frac{\partial p_{-(\nu+1)}}{\partial \varphi} + \rho \frac{\partial}{\partial \rho} \left[\chi_{-(\nu+1)} + \rho \frac{\partial \chi_{-(\nu+1)}}{\partial \rho} - \boldsymbol{R}_{\delta\gamma} \cdot \nabla \chi_{-(\nu+1)} \right] \right\}. \qquad (3.62)$$

Here, the partial derivatives are taken in the spherical coordinates (ρ, θ, φ) associated with the axial coordinate system centered at x^o_δ. The harmonics (3.13a) $p_{-(\nu+1)}(\boldsymbol{R}_\gamma)$ and (3.13c) $\chi_{-(\nu+1)}(\boldsymbol{R}_\gamma)$ are expanded near x^o_δ by (3.59), and the $\boldsymbol{R}_{\delta\gamma} \cdot \nabla$-term in (3.62) is then handled by the relation

$$D_3 \rho^n Y_{n,m}(\boldsymbol{\rho}) = \left[\frac{(2n+1)(n-m)(n+m)}{2n-1}\right]^{1/2} \rho^{n-1} Y_{n-1,m}(\boldsymbol{\rho}), \qquad (3.63)$$

where $D_3 = \partial/\partial\rho_3$ is the Cartesian partial derivative. This procedure gives

$$
\begin{aligned}
C_{n,m} = {} & -\frac{imR_{\delta\gamma}}{n(n+1)} \sum_{v=|m|}^{\infty} \frac{1}{v} \Gamma_{v,n}^{m} A_{-(v+1),m}^{(\gamma)} + \sum_{v=|m|}^{\infty} \Gamma_{v,n}^{m} C_{-(v+1),m}^{(\gamma)} \\
& - \frac{R_{\delta\gamma}}{(n+1)d_{\delta}^{o}} \left[\frac{(2n+3)(n-m+1)(n+m+1)}{2n+1} \right]^{1/2} \sum_{v=|m|}^{\infty} \Gamma_{v,n+1}^{m} C_{-(v+1),m}^{(\gamma)} \quad (3.64)
\end{aligned}
$$

(omitting $v = 0$ for $m = 0$). For calculating $B_{n,m}$, we dot the velocity (3.12) with $\boldsymbol{\rho}$, to obtain

$$
\begin{aligned}
\sum_{n=1}^{\infty} & \left[\frac{n\rho^2 p_n}{2(2n+3)} + n\Phi_n \right] \\
= {} & \sum_{v=0}^{\infty} \rho \frac{\partial \Phi_{-(v+1)}}{\partial \rho} - R_{\delta\gamma} \sum_{v=1}^{\infty} \frac{\partial \chi_{-(v+1)}}{\partial \varphi} + \sum_{v=1}^{\infty} \left[\frac{(v+1)}{v(2v-1)} \left(\rho^2 - R_{\delta\gamma}\rho_3 \right) p_{-(v+1)} \right. \\
& \left. - \frac{(v-2)\rho}{2v(2v-1)} \left(\rho^2 + R_{\delta\gamma}^2 - 2R_{\delta\gamma}\rho_3 \right) \frac{\partial p_{-(v+1)}}{\partial \rho} \right], \quad (3.65)
\end{aligned}
$$

where ρ_3 is the Cartesian component in the axial basis and, again, the partial derivatives are taken in the associated spherical coordinates (ρ, θ, φ) centered at $\boldsymbol{x}_{\delta}^{o}$. Expanding the harmonics (3.13) by (3.59) in terms of $\boldsymbol{\rho}$, and using the recurrent relation (3.36b), one can obtain from (3.65)

$$
\begin{aligned}
B_{n,m} = {} & -\frac{im}{n} R_{\delta\gamma} \sum_{v=|m|}^{\infty} \Gamma_{v,n}^{m} C_{-(v+1),m}^{(\gamma)} + \sum_{v=|m|}^{\infty} \Gamma_{v,n}^{m} B_{-(v+1),m}^{(\gamma)} \\
& - R_{\delta\gamma}^2 \sum_{v=|m|}^{\infty} \frac{(v-2)}{2v(2v-1)} \Gamma_{v,n}^{m} A_{-(v+1),m}^{(\gamma)} + \frac{R_{\delta\gamma}d_{\delta}^{o}}{n} \left[\frac{(n-m)(n+m)}{(2n-1)(2n+1)} \right]^{1/2} \\
& \times \sum_{v=|m|}^{\infty} \frac{[(n-1)(v-2)-(v+1)]}{v(2v-1)} \Gamma_{v,n-1}^{m} A_{-(v+1),m}^{(\gamma)} \quad (3.66)
\end{aligned}
$$

(omitting $v = 0$ in the last two sums for $m = 0$). The transformation relations from $\tilde{A}_{-(v+1),m}^{(\gamma)}$, etc., to $A_{n,m}$, etc., are identical to (3.61), (3.64), and (3.66).

Our optimized routine based on this simple, axial form of reexpansion relations and rotational techniques (see above) is quite fast. For example, with $v_{nf} = n_{nf} \gg 1$, it takes only $1.4 \times 10^{-7} v_{nf}^3$ seconds of CPU time on a DEC 500au workstation to reexpand (3.12) or (3.26) into (3.17) for one $\mathcal{B}_{\gamma}$.

3.4. The Far-Field Part

Our approach to calculating the far-field part of the boundary-integral operators is based on the special form of the Taylor expansion for a harmonic function $f(\boldsymbol{x})$,

$$
f(\boldsymbol{x}) = \sum_{v=0}^{\infty} \sum_{\mu=-v}^{v} \partial_{v,\mu} f(\boldsymbol{x})|_{\boldsymbol{x}=0} Z_{v,\mu}(\boldsymbol{x}), \qquad \partial_{v,\mu} = (D_1 - iD_2)^{\mu} D_3^{v-|\mu|}, \quad (3.67)
$$

where $D_i = \partial/\partial x_i$ is the Cartesian partial derivative, $(D_1 - iD_2)^\mu = (-1)^\mu (D_1 + iD_2)^{-\mu}$ for $\mu < 0$, and $Z_{\nu,\mu}(x)$ is the special solid harmonic (3.16). The expansion (3.67) directly follows from Lemma (B.1) of Zinchenko [17] and is also equivalent to (27) of Sangani and Mo [11].

Another useful observation is that $v(x) - \frac{1}{2}p(x)x$ is a harmonic vector field, if $(v(x), p(x))$ are the Stokes velocity and associated pressure, respectively, with unit viscosity ($v(x)$ may be compressible, only $\nabla^2 v = \nabla p$ and $\nabla^2 p = 0$ must hold). It is convenient to introduce a harmonic field ($g_{kl} \neq g_{lk}$)

$$g_{kl}(r) = (G_1)_{kl}(r) - \frac{1}{2}q_1^{(k)}(r)r_l, \tag{3.68}$$

where $q_1^{(k)}(r)$ is the pressure associated with the Stokes velocity $((G_1)_{k1}, (G_1)_{k2}, (G_1)_{k3})$, and apply (3.67) to both g_{kl} and $q_1^{(k)}$. Using the symmetry $g(-r) = g(r)$, $q_1^{(k)}(-r) = -q_1^{(k)}(r)$, one obtains

$$\sum_{x_j \in \mathcal{B}_\gamma} W_k(x_j)(G_1)_{kl}(x_j - y)$$

$$= \sum_{\nu=0}^{\infty}\sum_{\mu=-\nu}^{\nu} D_{\nu,\mu,k}^{(\gamma)}\partial_{\nu,\mu}g_{kl}(y - x_\gamma^o) - \frac{1}{2}\sum_{\nu=0}^{\infty}\sum_{\mu=-\nu}^{\nu}\Big[E_{\nu,\mu,k,l}^{(\gamma)}$$

$$+ (x_\gamma^o - y)_l D_{\nu,\mu,k}^{(\gamma)}\Big]\partial_{\nu,\mu}q_1^{(k)}(y - x_\gamma^o), \tag{3.69}$$

where $D_{\nu,\mu,k}^{(\gamma)}$ and $E_{\nu,\mu,k,l}^{(\gamma)}$ are the far-field moments (3.15). The high-order derivatives in (3.69), as harmonic functions of y, can be expanded at $y = x_\delta^o$ by (3.67), leading to

$$\sum_{x_j \in \mathcal{B}_\gamma} W_k(x_j)(G_1)_{kl}(x_j - y)$$

$$= \sum_{n=0}^{\infty}\sum_{m=-n}^{n} Z_{n,m}(R_\delta)\Bigg\{\sum_{\nu=0}^{\infty}\sum_{\mu=-\nu}^{\nu}\Big[D_{\nu,\mu,k}^{(\gamma)}\partial_{n+\nu,m+\mu}g_{kl}(R_{\gamma\delta})$$

$$- \frac{1}{2}E_{\nu,\mu,k,l}^{(\gamma)}\partial_{n+\nu,m+\mu}q_1^{(k)}(R_{\gamma\delta}) + \frac{1}{2}(R_{\gamma\delta})_l D_{\nu,\mu,k}^{(\gamma)}\partial_{n+\nu,m+\mu}q_1^{(k)}(R_{\gamma\delta})\Big]\Bigg\}$$

$$+ \frac{1}{2}(R_\delta)_l \sum_{n=0}^{\infty}\sum_{m=-n}^{n} Z_{n,m}(R_\delta)\sum_{\nu=0}^{\infty}\sum_{\mu=-\nu}^{\nu} D_{\nu,\mu,k}^{(\gamma)}\partial_{n+\nu,m+\mu}q_1^{(k)}(R_{\gamma\delta}), \tag{3.70}$$

with $R_{\gamma\delta} = x_\delta^o - x_\gamma^o$. The expansion (3.70) is a detailed form of (3.23) and, as a function of $R_\delta = y - x_\delta^o$, has the same structure as (3.19) (recall (3.16)).

The far-field expansion for the double-layer contribution can be obtained in the same manner:

$$\sum_{x_j \in \mathcal{B}_\gamma} Q_s(x_j)(\tau_1)_{ksl}(x_j - y)W_k(x_j)$$

$$= \sum_{n=0}^{\infty}\sum_{m=-n}^{n} Z_{n,m}(R_\delta)\Bigg\{\sum_{\nu=0}^{\infty}\sum_{\mu=-\nu}^{\nu}\Big[\tilde{D}_{\nu,\mu,k,s}^{(\gamma)}\partial_{n+\nu,m+\mu}t_{ksl}(R_{\gamma\delta})$$

$$+ \frac{1}{2} \tilde{E}_{\nu,\mu,k,s,l}^{(\gamma)} \partial_{n+\nu,m+\mu} \tilde{q}_1^{(ks)}(\boldsymbol{R}_{\gamma\delta}) + \frac{1}{2}(\boldsymbol{R}_{\gamma\delta})_l \tilde{D}_{\nu,\mu,k,s}^{(\gamma)} \partial_{n+\nu,m+\mu} \tilde{q}_1^{(ks)}(\boldsymbol{R}_{\gamma\delta}) \Big] \Big\}$$

$$+ \frac{1}{2}(\boldsymbol{R}_\delta)_l \sum_{n=0}^{\infty} \sum_{m=-n}^{n} Z_{n,m}(\boldsymbol{R}_\delta) \sum_{\nu=0}^{\infty} \sum_{\mu=-\nu}^{\nu} \tilde{D}_{\nu,\mu,k,s}^{(\gamma)} \partial_{n+\nu,m+\mu} \tilde{q}_1^{(ks)}(\boldsymbol{R}_{\gamma\delta}). \tag{3.71}$$

Here, $\tilde{q}_1^{(ks)}(\boldsymbol{r})$ is the pressure associated with the (compressible) Stokesian velocity $((\tau_1)_{ks1}$, $(\tau_1)_{ks2}, (\tau_1)_{ks3})(\boldsymbol{r})$ (so that $\tilde{q}_1^{(sk)}(\boldsymbol{r}) = \tilde{q}_1^{(ks)}(\boldsymbol{r}) = \tilde{q}_1^{(ks)}(-\boldsymbol{r})$), and

$$t_{ksl}(\boldsymbol{r}) = (\tau_1)_{ksl}(\boldsymbol{r}) - \frac{1}{2}\tilde{q}_1^{(ks)}(\boldsymbol{r})r_l \tag{3.72}$$

is an auxiliary harmonic field, with $t_{ksl}(\boldsymbol{r}) = t_{skl}(\boldsymbol{r}) = -t_{ksl}(-\boldsymbol{r})$. The double-layer far-field moments

$$\tilde{D}_{\nu,\mu,k,s}^{(\gamma)} = (-1)^{\nu-1} \sum_{\boldsymbol{x}_j \in \mathcal{B}_\gamma} W_{(s}(\boldsymbol{x}_j) Q_{k)}(\boldsymbol{x}_j) Z_{\nu,\mu}(\boldsymbol{x}_j - \boldsymbol{x}_\gamma^o), \tag{3.73a}$$

$$\tilde{E}_{\nu,\mu,k,s,l}^{(\gamma)} = (-1)^{\nu} \sum_{\boldsymbol{x}_j \in \mathcal{B}_\gamma} (\boldsymbol{x}_j - \boldsymbol{x}_\gamma^o)_l W_{(s}(\boldsymbol{x}_j) Q_{k)}(\boldsymbol{x}_j) Z_{\nu,\mu}(\boldsymbol{x}_j - \boldsymbol{x}_\gamma^o) \tag{3.73b}$$

(where $W_{(s}Q_{k)} = \frac{1}{2}(W_s Q_k + W_k Q_s)$ is the symmetrization in indices s and k) are precalculated for all blocks $\mathcal{B}_\gamma$ on every iteration, before the sums (3.71) are calculated.

Sangani *et al.* [39] and Zinchenko [16] developed different formalisms to calculate necessary high-order derivatives of the periodic Green's function for Laplace interactions. The computation of high-order derivatives in (3.70) and (3.71) is slightly more complex. We start from Ewald-like forms for $\boldsymbol{G}_1 = \{(G_1)_{kl}\}$ and associated pressures $\boldsymbol{q}_1 = (q_1^{(1)}, q_1^{(2)}, q_1^{(3)})$,

$$\boldsymbol{G}_1(\boldsymbol{x}) = \frac{\boldsymbol{I}}{4\pi} + \frac{1}{4\pi^{3/2}} \int_0^{\pi^{1/2}} e^{-t^2\boldsymbol{x}^2}[\boldsymbol{I} + 2t^2\boldsymbol{x}\boldsymbol{x}]\,dt$$

$$- \frac{1}{4\pi^{3/2}} \sum_{\boldsymbol{k}}{}' \int_{\pi^{1/2}}^{\infty} e^{-t^2(\boldsymbol{x}-\boldsymbol{k})^2}[\boldsymbol{I} + 2t^2(\boldsymbol{x}-\boldsymbol{k})(\boldsymbol{x}-\boldsymbol{k})]\,dt$$

$$- \frac{1}{4\pi^2} \sum_{\boldsymbol{k}}{}' \left[\boldsymbol{I} - \pi\boldsymbol{k}\boldsymbol{k}\left(1 + \frac{1}{\pi\boldsymbol{k}^2}\right)\right] \frac{e^{-\pi\boldsymbol{k}^2 - 2\pi i\boldsymbol{k}\cdot\boldsymbol{x}}}{\boldsymbol{k}^2}, \tag{3.74a}$$

$$q_1(\boldsymbol{x}) = -\boldsymbol{x} + \frac{\boldsymbol{x}}{\pi^{3/2}} \int_0^{\pi^{1/2}} e^{-t^2\boldsymbol{x}^2} t^2\,dt - \frac{1}{\pi^{3/2}} \sum_{\boldsymbol{k}}{}'(\boldsymbol{x}-\boldsymbol{k}) \int_{\pi^{1/2}}^{\infty} e^{-t^2(\boldsymbol{x}-\boldsymbol{k})^2} t^2\,dt$$

$$- \frac{i}{2\pi} \sum_{\boldsymbol{k}}{}' \frac{\boldsymbol{k} e^{-\pi\boldsymbol{k}^2 - 2\pi i\boldsymbol{k}\cdot\boldsymbol{x}}}{\boldsymbol{k}^2}, \tag{3.74b}$$

where the summation $\sum'$ is over all integer vectors $\boldsymbol{k} = (k_1, k_2, k_3) \neq \boldsymbol{0}$. The average of $\boldsymbol{G}_1 + \boldsymbol{G}_0$ over the cell $V = [0, 1)^3$ is zero, as required in Section 2. One way to derive (3.74) is to use Hasimoto's expressions [27] for $\boldsymbol{G}(\boldsymbol{x})$ and $\nabla\boldsymbol{q}(\boldsymbol{x})$ for a special lattice considered herein and then subtract the free-space contributions $\boldsymbol{G}_0(\boldsymbol{x})$ and $\boldsymbol{q}_0(\boldsymbol{x})$. Beenakker's approach [28], although popular, gives slower convergent series (by a factor of $O(\boldsymbol{k}^2)$ for each term both in the real and reciprocal spaces); besides, (3.74a), (3.74b) are much simpler and more amenable for calculating high-order derivatives.

According to (3.68) and (3.74),

$$g_{sl}(\boldsymbol{x}) = \frac{\delta_{sl}}{4\pi^{3/2}} \int_0^{\pi^{1/2}} e^{-t^2 x^2}\, dt - \frac{1}{4\pi^{3/2}} \sum_{\boldsymbol{k}}{}' \int_{\pi^{1/2}}^{\infty} e^{-t^2(\boldsymbol{x}-\boldsymbol{k})^2} \left[\delta_{sl} - 2t^2(x_s - k_s)k_l \right] dt + \cdots,$$

(3.75)

where only the integral (real-space) contributions are shown (the other terms are easy to differentiate). Generalizing (58) of Zinchenko [16], we have

$$\left(\frac{\partial}{\partial y_1} - i\frac{\partial}{\partial y_2} \right)^{\mu} \left(\frac{\partial}{\partial y_3} \right)^{\nu-\mu} \left[Z_{n,m}(\boldsymbol{y}) \int_{\pi^{1/2}}^{\infty} t^{2p} e^{-t^2 y^2}\, dt \right]$$

$$= \frac{\pi^{1/2}}{2^{p+1}} \sum_{s,l} \frac{(-1)^{\nu+s+l}(\nu-\mu)!\mu!(y_1 - iy_2)^{\mu-s}}{l!(\nu-\mu-l)!s!(\mu-s)!} J_{\nu-l-s+p}^{\mu-s+p}(\boldsymbol{y}) Z_{n-s-l,m-s}(\boldsymbol{y}) \quad (3.76)$$

for $0 \le \mu \le \nu$, $|m| \le n$, where the summation is over all integers s, l with

$$0 \le s \le \mu, \qquad 0 \le l \le \min(\nu-\mu, n-m), \qquad 2s + l \le n + m, \qquad (3.77)$$

and

$$J_{\nu}^{\mu}(\boldsymbol{y}) = \frac{2^{\mu+1}}{\pi^{1/2}} \int_{\pi^{1/2}}^{\infty} t^{\nu+\mu} H_{\nu-\mu}(y_3 t) e^{-t^2 y^2}\, dt, \qquad (3.78)$$

with $H_n(\xi)$ being the Hermite polynomial [33, 34]. Relations (3.76)–(3.78) also hold for the integrals over $[0, \pi^{1/2}]$, and efficient recurrent schemes [16] can be used for calculating J_{ν}^{μ} in both cases. Since $x_s - k_s$ is a combination of harmonics $Z_{1,m}(\boldsymbol{x} - \boldsymbol{k})$, relations (3.76) for $p = n = 1$ and $p = n = 0$, together with (3.16), (3.74)–(3.75), suffice for a simple computation of $\partial_{\nu,\mu}\boldsymbol{g}(\boldsymbol{x})$ and $\partial_{\nu,\mu}\boldsymbol{q}_1(\boldsymbol{x})$ to an arbitrary order.

Ewald-like forms for the stresslet $(\tau_1)_{ksl}$ and associated pressures $\tilde{q}_1^{(ks)}$ are not required, since $\partial_{\nu,\mu}t_{ksl}(\boldsymbol{x})$ and $\partial_{\nu,\mu}\tilde{q}_1^{(ks)}(\boldsymbol{x})$ are simply expressed via $\partial_{\nu',\mu'}\boldsymbol{g}$ and $\partial_{\nu',\mu'}\boldsymbol{q}_1(\boldsymbol{x})$. Indeed, with a suitable choice of an additive constant, $\tilde{q}_1^{(ks)}(\boldsymbol{x}) = 2\partial q_1^{(s)}(\boldsymbol{x})/\partial x_k$ (cf. with Subsection 3.2), and harmonicity of $q_1^{(s)}$ implies

$$\partial_{\nu,\mu}\tilde{q}_1^{(1,s)} = (\partial_{\nu+1,\mu+1} - \partial_{\nu+1,\mu-1})q_1^{(s)}, \qquad \partial_{\nu,\mu}\tilde{q}_1^{(2,s)} = i(\partial_{\nu+1,\mu+1} + \partial_{\nu+1,\mu-1})q_1^{(s)},$$

(3.79)

$$\partial_{\nu,\mu}\tilde{q}_1^{(3,s)} = 2\partial_{\nu+1,\mu}q_1^{(s)}.$$

Besides, it follows from (2.2), (2.9), (3.68), (3.72) and the symmetry of $\tilde{\boldsymbol{q}}_1$ that

$$t_{ksl}(\boldsymbol{x}) = -q_1^{(l)}\delta_{ks} + \frac{1}{2}q_1^{(s)}\delta_{kl} + \frac{1}{2}q_1^{(k)}\delta_{sl} + D_k g_{sl} + D_s g_{kl} - \delta_{ks}x_l. \qquad (3.80)$$

Since $\boldsymbol{g}(\boldsymbol{x})$ is harmonic, the same algebra as in (3.79) allows us to express $\partial_{\nu,\mu}t_{ksl}$ via $\partial_{\nu+1,\mu'}\boldsymbol{g}$ and $\partial_{\nu,\mu}\boldsymbol{q}_1$.

Our fast calculation of the derivatives in (3.70) and (3.71) at each time step is based on a precalculated table of $\partial_{n,m}g_{kl}(\boldsymbol{x})$ and $\partial_{n,m}q_1^{(k)}(\boldsymbol{x})$ (with $0 \le m \le n$ and $k \le l$) on a mesh $\boldsymbol{x} = (n_1 h, n_2 h, n_3 h)$, where $h = 0.5/N_T$, the integers n_i are in the range $0 \le n_1 \le n_2 \le N_T$, $0 \le n_3 \le N_T$, and N_T is typically about 20; the maximum order depends on the system size. For a given $\boldsymbol{R}_{\gamma\delta} \in [-\frac{1}{2}, \frac{1}{2})^3$, the nearest node $\boldsymbol{x}_o = (n_1^o h, n_2^o h, n_3^o h)$ (with $|n_i^o| \le N_T$) is found, and $\partial_{n,m}g_{kl}$ (for $k \le l$ and $k > l$) and $\partial_{n,m}q_1^{(k)}$ are calculated at this node based on

the table and symmetries of $\boldsymbol{G}_1$ and $\boldsymbol{q}_1$ (see (3.74)). For (3.70), $\partial_{n,m}\boldsymbol{g}(\boldsymbol{R}_{\gamma\delta})$ and $\partial_{n,m}\boldsymbol{q}_1(\boldsymbol{R}_{\gamma\delta})$ are then computed as second-order or third-order Taylor expansions at $\boldsymbol{x}_o$ by (3.67), for example,

$$\partial_{n,m}\boldsymbol{g}(\boldsymbol{R}_{\gamma\delta}) = \sum_{\nu\leq 2 \text{ or } 3} \sum_{\mu=-\nu}^{\nu} \partial_{n+\nu,m+\mu}\boldsymbol{g}(\boldsymbol{x}_o) Z_{\nu,\mu}(\boldsymbol{R}_{\gamma\delta}-\boldsymbol{x}_o) \tag{3.81}$$

(which is more appropriate than the simplest linear interpolation, since $\partial_{n,m}\boldsymbol{g}(\boldsymbol{x})$ and $\partial_{n,m}\boldsymbol{q}_1(\boldsymbol{x})$ are slowly varying in the range $|x_i| \leq 1/2$ only for very small n). For (3.71), $\partial_{n,m}\boldsymbol{t}(\boldsymbol{R}_{\gamma\delta})$ and $\partial_{n,m}\tilde{\boldsymbol{q}}_1(\boldsymbol{R}_{\gamma\delta})$ are expressed via $\partial_{n',m'}\boldsymbol{g}(\boldsymbol{R}_{\gamma\delta})$ and $\partial_{n',m'}\boldsymbol{q}_1(\boldsymbol{R}_{\gamma\delta})$. To cover all the simulations and more demanding tests in Section 6, $\partial_{n,m}\boldsymbol{g}$ and $\partial_{n,m}\boldsymbol{q}_1$ were tabulated up to $n=20$ on a mesh with $N_T = 20$, although medium-precision dynamical simulations for $N > 100$ typically require much smaller values of n. The single-precision format was found sufficient for extensive tables of $\partial_{n,m}\boldsymbol{g}$ and $\partial_{n,m}\boldsymbol{q}_1$, while double precision is used in the rest of the code. For typical truncation bounds $v_{ff}=n_{ff}$, $v_{ff}^*=0.7(v_{ff}+n_{ff})$ in (3.25), it takes our algorithm about $5 \times 10^{-7}(v_{ff}+1)^4$ seconds of CPU time on a DEC 500au workstation to calculate the coefficients before $Z_{n,m}(\boldsymbol{R}_\delta)$ and $\boldsymbol{R}_\delta Z_{n,m}(\boldsymbol{R}_\delta)$ for (3.71), and half as much for the single-layer case (3.70). With the quadratic Taylor approximation (see above), additional operations on calculating the derivatives in (3.70) or (3.71) slow down the far-field part by 22 to 45% for typical v_{ff}, n_{ff} from 7 to 4, respectively.

The far-field part of self-interactions (Subsection 3.1) is a particular case of the considered scheme, when $\mathcal{B}_\gamma$ constitutes a whole drop and $\boldsymbol{R}_{\gamma\delta} = 0$ in (3.70)–(3.71).

3.5. *Economical Truncation Bounds*

Unlike in the other parts of our code, there is a considerable freedom in constructing the truncation bounds v_{nf}, n_{nf}, v_{nf}^*, v_{ff}, n_{ff}, and v_{ff}^* (Subsection 3.1) of multipole expansions for a given precision ε. In principle, any choice of these bounds, such that v_{nf}, n_{nf}, etc. $\to \infty$ (if unrestricted by k_o) for $\varepsilon \to 0$, is allowed, since this condition alone guarantees the convergence to the standard, nonmultipole boundary-integral solution for a given triangulation. However, an unreasonable, *ad hoc* choice of the bounds (especially, uniform v_{nf}, n_{nf}, v_{nf}^*) can greatly reduce the performance. Most importantly, only the interaction of low-order multipoles is long-ranged, and so optimal near-field truncation bounds should be strongly sensitive to mutual geometry of the blocks. Rigorous majorants for multipole coefficients are problematic, especially for $\lambda \neq 1$, and they would probably greatly overestimate the actual truncation errors. Instead, our approach is based, in part, on some plausible arguments about the behavior of multipole coefficients.

We start from determining the bounds $v_{nf}+1$ and $n_{nf}+1$ on ν and n, respectively, in the reexpansion from (3.12) to (3.17) assuming $\mathcal{D}_\delta \cap \mathcal{D}_\gamma = \emptyset$. For a fixed direction of $\boldsymbol{R}_\gamma = \boldsymbol{y} - \boldsymbol{x}_\gamma^o$, (3.12) can be viewed as a Taylor expansion,

$$\sum_{\boldsymbol{x}_j \in \mathcal{B}_\gamma} \boldsymbol{G}_0(\boldsymbol{x}_j - \boldsymbol{y}) \cdot \boldsymbol{W}(\boldsymbol{x}_j) = \sum_{k=0}^{\infty} a_k \left(\frac{d_\gamma^o}{R_\gamma}\right)^{k+1}, \tag{3.82}$$

convergent for $R_\gamma > d_\gamma^o$. The asymptotic behavior of $|a_k|$ for $k \gg 1$ is related to the behavior of the LHS of (3.82) at $\boldsymbol{y} \to \boldsymbol{x}_{j^*}$, where $\boldsymbol{x}_{j^*} \in \mathcal{B}_\gamma$ is the node with $|\boldsymbol{x}_{j^*} - \boldsymbol{x}_\gamma^o| = d_\gamma^o$. For a continuous distribution of Stokeslets, the single-layer gradient is finite at internal points of a surface and only logarithmically singular at the edge, so an approximate behavior

$a_k = O(k^{-3})$ is assumed for moderately large k. At $k \to \infty$, however, $a_k = O(1)$ due to the discrete structure of the LHS of (3.82), with a pole singularity at $\mathbf{y} = \mathbf{x}_{j*}$. The following model of $|a_k|$ is assumed for *all* k:

$$|a_k| = \begin{cases} C_\gamma / (k+1)^3, & k < k_{cr} \\ C_\gamma / (k_{cr}+1)^3, & k \geq k_{cr}. \end{cases} \tag{3.83}$$

Numerical experiments suggest the switch value

$$k_{cr} = 2d_\gamma^o (\triangle S_{j*})^{-1/2}, \tag{3.84}$$

where $\triangle S_j$ is given by (3.2). The constant C_γ is estimated by considering $k = 0$ in (3.82):

$$C_\gamma = \frac{1}{4\pi d_\gamma^o} \left\| \sum_{\mathbf{x}_j \in \mathcal{B}_\gamma} \mathbf{W}(\mathbf{x}_j) \right\|. \tag{3.85}$$

Since $R_\gamma \geq R_{\delta\gamma} - d_\delta^o$ for $\mathbf{y} \in \mathcal{B}_\delta$, the series (3.82) is majorated by a double series:

$$\sum_{k=0}^{\infty} \sum_{n=0}^{\infty} \Psi_{n,k}, \qquad \Psi_{n,k} = |a_k| \frac{(k+n)!}{k!n!} \left(\frac{d_\delta^o}{R_{\delta\gamma}} \right)^n \left(\frac{d_\gamma^o}{R_{\delta\gamma}} \right)^{k+1}. \tag{3.86}$$

Obviously, (3.86) mimics the reexpansion from Lamb's singular (3.12) to regular (3.17) forms. The bounds $\nu_{nf}(\delta, \gamma)$ and $n_{nf}(\delta, \gamma)$ are such that the remainder of the sum (3.86) is within a prescribed tolerance:

$$\sum_{n,k: n > n_{nf} \text{ or } k > \nu_{nf}} \Psi_{n,k} < \varepsilon_1, \qquad \varepsilon_1 = \frac{e_{nf} R_{\delta\gamma}^{-4} \varepsilon}{\sum_\gamma R_{\delta\gamma}^{-4}}. \tag{3.87}$$

The summation in the RHS of (3.87) is over all blocks $\mathcal{B}_\gamma \not\subset S_\alpha$ (where S_α is the drop containing $\mathcal{B}_\delta$) and $e_{nf} = O(1)$ is a numerical factor found experimentally; $R_{\delta\gamma}^{-4}$ is chosen as the simplest function integrable at infinity in $\Re^3$ (cf. [16]). The form (3.87) for ε_1 guarantees that the sum of the omitted contributions from *all* $\mathcal{B}_\gamma$ to (3.86) is less than $e_{nf}\varepsilon$.

We first find ν_{nf} from

$$\sum_{n=0}^{\infty} \sum_{k=\nu_{nf}+1}^{\infty} \Psi_{n,k} = \left(\sum_{k=0}^{\infty} - \sum_{k=0}^{\nu_{nf}} \right) |a_k| \left(\frac{d_\gamma^o}{R_{\delta\gamma} - d_\delta^o} \right)^{k+1} < \frac{\varepsilon_1}{2} \tag{3.88}$$

(see region A in Fig. 5.) This task is easy, since the first sum in the RHS of (3.88) is known explicitly through (3.83). The bound n_{nf} is then determined from

$$\sum_{n=n_{nf}+1}^{\infty} \sum_{k=0}^{\nu_{nf}} \Psi_{n,k} = \sum_{k=0}^{\nu_{nf}} |a_k| \left(\frac{d_\gamma^o}{R_{\delta\gamma} - d_\delta^o} \right)^{k+1} - \sum_{n=0}^{n_{nf}} \sum_{k=0}^{\nu_{nf}} \Psi_{n,k} < \frac{\varepsilon_1}{2} \tag{3.89}$$

(see region B in Fig. 5). Inequalities (3.88)–(3.89) imply (3.87). If $\nu_{nf} > k_o$, or $n_{nf} > k_o$, or $R_{\delta\gamma} < d_\gamma^o + d_\delta^o$, the bound $\nu_{nf}^* + 1$ on ν is used instead for pointwise calculation of (3.12) for $\mathbf{y} \in \mathcal{B}_\delta$, $R_\gamma > d_\gamma^o$. This bound is simply determined from

$$\sum_{k=0}^{\infty} |a_k| \left(\frac{d_\gamma^o}{R_\gamma} \right)^{k+1} - \sum_{k=0}^{\nu_{nf}^*} |a_k| \left(\frac{d_\gamma^o}{R_\gamma} \right)^{k+1} < \varepsilon_1. \tag{3.90}$$
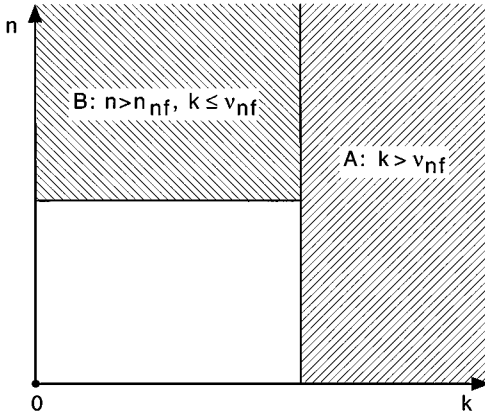
FIG. 5. The determination of near-field truncation bounds.

A very similar technique is used to limit the summations by $\nu \leq \tilde{\nu}_{nf} + 1$ and $n \leq \tilde{n}_{nf} + 1$ in the double-layer reexpansion from (3.26) to (3.17), and also to find the bound $\tilde{\nu}_{nf}^*(y, \gamma) + 1$ on ν for pointwise calculation of (3.26) in case $\tilde{\nu}_{nf} > k_o$, or $\tilde{n}_{nf} > k_o$, or $\mathcal{D}_\delta \cap \mathcal{D}_\gamma \neq \emptyset$. The RHS of (3.26) is estimated as

$$\sum_{k=1}^{\infty} \tilde{a}_k \left(\frac{d_\gamma^o}{R_\gamma} \right)^{k+1}, \qquad |\tilde{a}_k| = \begin{cases} \tilde{C}_\gamma / k^2, & k < k_{\mathrm{cr}} \\ \tilde{C}_\gamma k / k_{\mathrm{cr}}^3, & k \geq k_{\mathrm{cr}}. \end{cases} \tag{3.91}$$

The switch value k_{cr} is still given by (3.84). The form (3.91) vs (3.83) reflects a higher double-layer singularity. The constant $\tilde{C}_\gamma$ is estimated as

$$\tilde{C}_\gamma = \frac{3}{4\pi \left(d_\gamma^o \right)^2} \left\| \sum_{x_j \in \mathcal{B}_\gamma} Q(x_j) W(x_j) \right\| \tag{3.92}$$

(Euclidean norm is used for the matrix (3.92)). All the relations (3.86)–(3.90) hold for determining $\tilde{\nu}_{nf}$, $\tilde{n}_{nf}$, and $\tilde{\nu}_{nf}^*$; only $k = 0$ is excluded from the summations. Besides, ε_1 differs from (3.87) by a factor of $2\tilde{e}_{nf}/|\lambda - 1|$ instead of e_{nf}, where $\tilde{e}_{nf} = O(1)$ is another coefficient determined experimentally; the presence of $2/|\lambda - 1|$ serves to balance the truncation errors for the inhomogeneous and double-layer terms in (2.12b).

For the far-field, we note that the terms in the RHS series (3.23) with $n + \nu \geq 3$ could be evaluated as if $G_1(x_j - y)$ was a formal sum of free-space contributions $G_0(x_j - y)$ from all periodic replicas of $\mathcal{B}_\gamma$, excluding $\mathcal{B}_\gamma$ itself (cf. [16]). Although this partition is not used in our calculations, it gives a way of constructing the far-field truncation bounds ν_{ff}, n_{ff}, ν_{ff}^* (see (3.25)) and similar bounds $\tilde{\nu}_{ff}$, $\tilde{n}_{ff}$, $\tilde{\nu}_{ff}^*$ for (3.71) by slightly modifying the above arguments. Namely, $\nu_{ff} + 1$ is found as the first value of $k \geq 3$ satisfying

$$\sum_{n=0}^{\infty} \Psi_{n,k} = |a_k| \left(\frac{d_\gamma^o}{\zeta_{\delta\gamma} - d_\delta^o} \right)^{k+1} < \frac{\varepsilon_2}{(k+1)^2}, \tag{3.93}$$

where $\zeta_{\delta\gamma} = \min \| R_{\delta\gamma} + m \|$ ($m \neq 0$ is an arbitrary integer vector) given by (3.24) is the minimum of center-to-center distances from $\mathcal{D}_\delta$ to periodic images of $\mathcal{D}_\gamma$ excluding $\mathcal{D}_\gamma$

itself, $\Psi_{n,k}$ for the far-field bounds differs from (3.86) in $\zeta_{\delta\gamma}$ instead of $R_{\delta\gamma}$, and only the upper form (3.83) for $|a_k|$ is used due to fast convergence of far-field expansions. The tolerance ε_2 is

$$\varepsilon_2 = \frac{6}{\pi^2} \frac{e_{ff} \zeta_{\delta\gamma}^{-4} \varepsilon}{\sum_\gamma \sum_{m \neq 0} \|R_{\delta\gamma} + m\|^{-4}}, \tag{3.94}$$

where $e_{ff} = O(1)$ is another numerical factor. As in (3.87), the summation in (3.94) is over all blocks $\mathcal{B}_\gamma \not\subset S_\alpha \supset \mathcal{B}_\delta$, and the estimate 23.5 ± 7 for the inner sum (3.94) suffices. For every $0 \leq k \leq \nu_{ff}$, the first value of n^* with $k + n^* \geq 2$ and

$$\sum_{n=n^*+1}^{\infty} \Psi_{n,k} < \frac{\varepsilon_2}{(k+1)^2} \tag{3.95}$$

is found, and n_{ff} and ν_{ff}^* are calculated as $\max(n^*(k))$ and $\max(n^*(k) + k)$, respectively; the limitation ν_{ff}^* on $\nu + n$ expedites (3.70), but a similar limitation for the near-field part would be inconvenient. The form (3.94) is designed to make the sum of the omitted contributions from all $\mathcal{B}_\gamma$ to the RHS of (3.70) less than $e_{ff}\varepsilon$, with our estimate (3.83), (3.85) for the moments. Only minor changes to (3.93)–(3.95) are used to limit the far-field double-layer expansion (3.71) by $\nu \leq \tilde{\nu}_{ff}$, $n \leq \tilde{n}_{ff}$, and $\nu + n \leq \tilde{\nu}_{ff}^*$. Namely, e_{ff} in (3.94) is replaced by $2\tilde{e}_{ff}/|\lambda - 1|$, where $\tilde{e}_{ff} = O(1)$ is one more numerical factor, and, using the upper form (3.91) for $|\tilde{a}_k|$, the value of $\tilde{\nu}_{ff}$ is determined from

$$\tilde{C}_\gamma \left[d_\gamma^o / (\zeta_{\delta\gamma} - d_\delta^o) \right]^{\tilde{\nu}_{ff}+3} < \varepsilon_2, \qquad \tilde{\nu}_{ff} \geq 1. \tag{3.96}$$

Next, for every $1 \leq k \leq \tilde{\nu}_{ff} + 1$, the first value of n^* with $k + n^* \geq 2$ and

$$\tilde{C}_\gamma \sum_{n=n^*+1}^{\infty} \frac{(n+k)!}{k!n!} \left(\frac{d_\delta^o}{\zeta_{\delta\gamma}} \right)^n \left(\frac{d_\gamma^o}{\zeta_{\delta\gamma}} \right)^{k+1} < \varepsilon_2 \tag{3.97}$$

is found, and $\tilde{n}_{ff}$ and $\tilde{\nu}_{ff}^*$ are calculated as $\max(n^*(k))$ and $\max(n^*(k) + k - 1)$.

Finally, it was convenient to consider the far-field truncation bounds $\nu_{ff}, \ldots \tilde{\nu}_{ff}^*$ for self-interactions (Subsection 3.1) as a particular case of the above schemes, when the summations (3.82), (3.85), and (3.92) are over the entire drop surface S_α, d_γ^o and d_δ^o are set to the radius d_α of the minimal shell around S_α centered at x_α^c, $\zeta_{\delta\gamma} = 1$, and the double sum in (3.94) is replaced by 33, twice the value of the inner sum (3.94) at $R_{\delta\gamma} = 0$.

The double-layer truncation bounds are calculated on every iteration at the initial moment $t = 0$, and only on the first iteration at $t > 0$, when the preceding time step gives an initial approximation for $Q(x_j)$. In all cases, the cost of truncation-bound calculations is negligible. After some experimenting, we fixed $e_{nf} = 1$, $e_{ff} = 10$, $\tilde{e}_{nf} = 0.2$, $\tilde{e}_{ff} = 2$. With chosen e_{nf}, e_{ff}, $\tilde{e}_{nf}$, $\tilde{e}_{ff}$, our truncation scheme depends on a single parameter ε, and all multipoles are eventually included, as $\varepsilon \to 0$, if unrestricted (for the near-field) by the threshold k_o.

4. BEST PARABOLOID-SPLINE METHOD FOR NORMALS AND CURVATURES

To calculate the mean curvature $k(x_i) = \frac{1}{2}(k_1 + k_2)$ and outward unit normal $n(x_i) = n_i$ at a node x_i of a triangulated drop surface S_α, we have developed a novel method called BPS ("best paraboloid-spline") herein. For an arbitrary vector n, consider a local coordinate
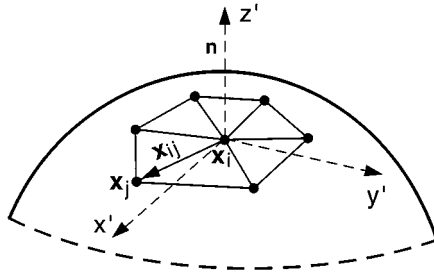
FIG. 6. Schematic for the calculation of the normal vectors and mean curvatures by the best paraboloid-spline method (BPS).

system (x', y', z') centered at $\boldsymbol{x}_i$ and with the z'-axis along $\boldsymbol{n}$ (Fig. 6). Let $(x'_{ij}, y'_{ij}, z'_{ij})$ be the (x', y', z') coordinates of vectors $\boldsymbol{x}_{ij} = \boldsymbol{x}_j - \boldsymbol{x}_i$ from $\boldsymbol{x}_i$ to adjacent nodes $\boldsymbol{x}_j$, and

$$\Pi_i(\boldsymbol{n}) = \min_{C,D,E} \sum_{\boldsymbol{x}_j \in \mathcal{A}_i} \frac{\left(Cx'^2_{ij} + Dx'_{ij}y'_{ij} + Ey'^2_{ij} - z'_{ij}\right)^2}{\|\boldsymbol{x}_{ij}\|^2}, \tag{4.1}$$

where the summation is over the set $\mathcal{A}_i$ of nodes $\boldsymbol{x}_j$ adjacent to $\boldsymbol{x}_i$. The minimum (4.1) is found by a simple solution of a linear system for C, D, and E, and is independent of an arbitrary rotation of (x', y', z') about the z'-axis. Obviously, $\Pi_i(\boldsymbol{n})$ is a measure of local deviation of S_α from the best paraboloid $z' = Cx'^2 + Dx'y' + Ez'^2$ with the given z'-axis, and $\Pi_i(\boldsymbol{n}) = O(h^4)$, if $\boldsymbol{n}$ is the exact normal at $\boldsymbol{x}_i$ and h is the characteristic mesh size. Also, $[\boldsymbol{x}_{ij} \cdot (\boldsymbol{n}_i + \boldsymbol{n}_j)]^2/\|\boldsymbol{x}_{ij}\|^2 = O(h^4)$ for mesh edges $\boldsymbol{x}_{ij}$ and exact normals $\boldsymbol{n}_i$, $\boldsymbol{n}_j$. Based on these observations, in the BPS method the set of normals $\{\boldsymbol{n}_i\}$ on S_α is found to provide a global minimum to the function

$$\Phi = \sum_{\boldsymbol{x}_i \in S_\alpha} \Pi_i(\boldsymbol{n}_i) + c_s \sum_{\boldsymbol{x}_{ij}} \frac{[\boldsymbol{x}_{ij} \cdot (\boldsymbol{n}_i + \boldsymbol{n}_j)]^2}{\|\boldsymbol{x}_{ij}\|^2} \tag{4.2}$$

under the constraints $\boldsymbol{n}_i^2 = 1$, where the first sum is over all mesh vertices $\boldsymbol{x}_i \in S_\alpha$, the second sum is over all mesh edges $\boldsymbol{x}_{ij}$ on S_α (with $i < j$ to avoid double counting), and $c_s = O(1)$ is a numerical coefficient. For $c_s = 0$, the normals $\boldsymbol{n}_i$ would be decoupled and BPS equivalent [3] to the local best-paraboloid technique [2]. The second term (4.2) of spline character is prompted by the curvatureless formulation [2, 3] and interlinks the normals $\boldsymbol{n}_i$ over the entire surface. Given an initial approximation to $\{\boldsymbol{n}_i\}$, all the normals $\boldsymbol{n}_i$ on S_α are successively updated by local minimizations of (4.2) with respect to one $\boldsymbol{n}_i$; these cycles (outer iterations) are repeated until all $\boldsymbol{n}_i$ stabilize to within $\delta_1 = 10^{-4}$–10^{-5}.

Inner iterations are required to minimize (4.2) with respect to one $\boldsymbol{n}_i$. A gradient method is the simplest,

$$\boldsymbol{n}_i^{(\nu+1)} = \boldsymbol{P}/\|\boldsymbol{P}\|, \qquad \boldsymbol{P} = \boldsymbol{n}_i^{(\nu)} - \delta \nabla^\| \Phi,$$

$$\nabla \Phi = \nabla \Pi_i\left(\boldsymbol{n}_i^{(\nu)}\right) + 2c_s \sum_{\boldsymbol{x}_j \in \mathcal{A}_i} \frac{[\boldsymbol{x}_{ij} \cdot (\boldsymbol{n}_i^{(\nu)} + \boldsymbol{n}_j)]\boldsymbol{x}_{ij}}{\|\boldsymbol{x}_{ij}\|^2}, \tag{4.3}$$

where $\boldsymbol{n}_i^{(\nu)}$ is the νth inner iteration and $\|$ means the projection on the plane orthogonal to $\boldsymbol{n}_i^\nu$; the small parameter δ is fixed at 0.05. The calculation of $\nabla^\| \Pi_i$ necessary for (4.3) is

described in [3] as a part of another method. Iterations (4.3) terminate once $\|\nabla^{\|}\Phi\| < \delta_2 \ll \delta_1$ (with $\delta_2 = 10^{-5}$–10^{-6}). An alternative, more efficient for a poor initial approximation, is to make one iteration (4.3) from $\boldsymbol{n}_i^{(0)}$ to $\boldsymbol{n}_i^{(1)}$ and generate two additional points $\boldsymbol{n}_i^{(2)}$ and $\boldsymbol{n}_i^{(3)}$ on the unit sphere, so that $\boldsymbol{n}_i^{(0)}, \boldsymbol{n}_i^{(2)}, \boldsymbol{n}_i^{(3)}$ form an equilateral spherical triangle centered at $\boldsymbol{n}_i^{(1)}$. Stereographic coordinates of $\boldsymbol{n}_i^{(0)}, \boldsymbol{n}_i^{(2)}$, and $\boldsymbol{n}_i^{(3)}$ (zero for $\boldsymbol{n}_i^{(1)}$) in the plane tangential to the unit sphere at $\boldsymbol{n}_i^{(1)}$ are obtained by projecting from the sphere center onto this plane, and Φ is approximated as linear plus quadratic functions of the stereographic coordinates. The linear part is known exactly through Φ and $\nabla^{\|}\Phi$ at $\boldsymbol{n}_i^{(1)}$, and the quadratic part is found by fitting Φ to its values at $\boldsymbol{n}_i^{(0)}, \boldsymbol{n}_i^{(2)}$, and $\boldsymbol{n}_i^{(3)}$. Minimization of the approximating function gives a new initial point $\boldsymbol{n}_i^{(0)}$, and the process is repeated until $\|\nabla^{\|}\Phi\| < \delta_2$. Upon convergence, the C and E coefficients in (4.1) also give the mean curvatures $k(\boldsymbol{x}_i) = -(C + E)$.

In dynamical simulations (Section 6), with $\delta_1 = 10^{-4}$, $\delta_2 = 10^{-5}$ and the initial approximation to $\{\boldsymbol{n}_i\}$ from the preceding step, typically 10–20 calculations of Π_i and $\nabla\Pi_i$ per node suffice, and so the BPS-part of our code is relatively inexpensive, even for $\lambda = 1$. "Spline" in the name of the method should not be misleading; from traditional splines, BPS only borrows the idea to interlink all $\boldsymbol{n}_i$ through a sparse matrix.

Figures 7–9 demonstrate the effect of c_s on the accuracy of the normal-vector calculations by BPS for ellipsoids $x^2/a^2 + y^2/b^2 + z^2/c^2 = 1$ and an axisymmetrical spool-like shape obtained by rotating the curve $x^2 = (0.2z^2 + 0.05)^2(1 - z^2)$ (the insert in Fig. 9b) about the z-axis. Unit-sphere triangulations into $N_\triangle = 80, 320, 1280$, and 5120 elements were prepared by a standard refinement procedure (e.g., [2]) from an icosaedron and subject to a random rotation (such a rotation may slow down the convergence but was used for generality). The simplest mappings $(x, y, z) \rightarrow (ax, by, cz)$ and $(x, y, z) \rightarrow ((0.2z^2 + 0.05)x, (0.2z^2 + 0.05)y, z)$ were then used to obtain ellipsoid and spool shape triangulations from unit-sphere triangulations. Intriguingly, Figs. 7–9 and additional calculations for other shapes reveal that $c_s = 1$ is always a much preferable choice (except for very crude $N_\triangle$) and gives a striking advantage over the local best paraboloid method ($c_s = 0$) for fine triangulations. For example, for the ellipsoid $a = 1, b = 0.5, c = 0.3$, and $N_\triangle = 1280$–5120, the average error of BPS is 29–60 times less, and the maximum error is 7–22 times less, compared to the best-paraboloid method (Figs. 7a, 7b).

In contrast, for the curvatures, the new method does not offer any significant improvements over the local best-paraboloid technique. Despite this bottleneck, using BPS with $c_s = 1$ drastically improves the quality of long-time multidrop simulations (Section 6), probably because accurate normals are essential in updating the surfaces. From our experience, BPS is recommended for drops with large deformations, but still away from breakup and cusping. In breakup simulations, not considered herein, the present version of BPS may be unstable, and we have preferred local methods [3]. As our most recent multidrop calculations show, it is reasonable to switch, in exclusive cases, from BPS to the best-paraboloid method for individual drops with abnormal high-curvature zones. Advantages over the contour integration method for determining the mean curvatures are described in [2].

5. ADDITIONAL DETAILS

5.1. *Passive Mesh Stabilization*

A familiar difficulty in 3D boundary-integral calculations for deformable drops is dynamical mesh degradation. Namely, if the collocation nodes are simply advected with the
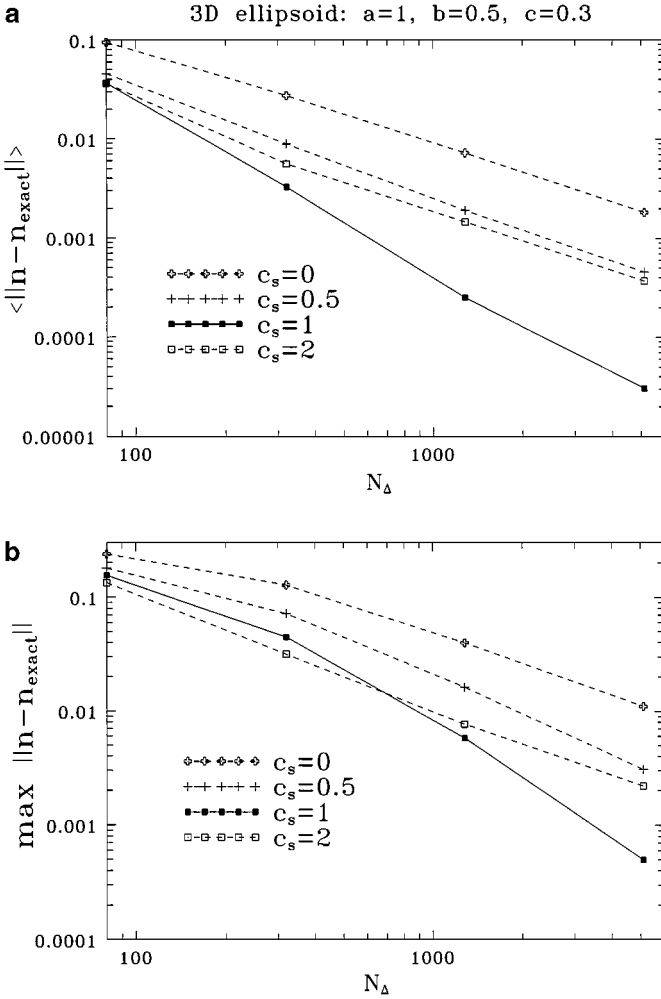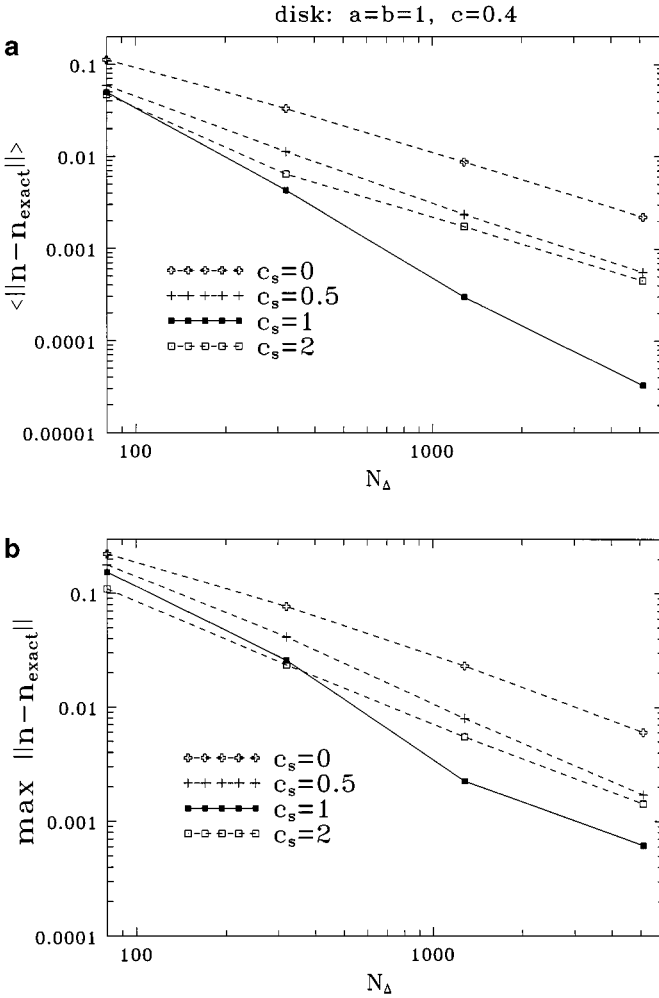
FIG. 7. The average (a) and maximum (b) absolute errors $\|\boldsymbol{n} - \boldsymbol{n}_{\text{exact}}\|$ in the normal vector calculation by BPS for a 3D ellipsoid with different numbers $N_\triangle$ of triangular elements; the limit $c_s = 0$ corresponds to the local best paraboloid method.

fluid velocity or with the surface normal velocity, an initially regular unstructured mesh of triangles on a surface becomes highly irregular and invalid after a short simulation time. We use one of the passive mesh stabilization schemes [3] to construct an additional global tangential field on each surface S_α separately from the solution of a variational problem to prevent mesh degradation. Namely, at any instant of time, the vertex velocities $\boldsymbol{V}_i = d\boldsymbol{x}_i/dt$ to be used in the shape updates are required to minimize

$$F = \sum_{\boldsymbol{x}_{ij}} \frac{1}{\|\boldsymbol{x}_{ij}\|^4} \left\{ \frac{d}{dt} [\boldsymbol{x}_{ij} \cdot (\boldsymbol{n}_j^* - \boldsymbol{n}_i^*)] \right\}^2$$

$$+ c_1 \langle |k_{\max}| \rangle^2 \sum_{\boldsymbol{x}_{ij}} \frac{1}{\|\boldsymbol{x}_{ij}\|^4} \left[\frac{d}{dt} \|\boldsymbol{x}_{ij}\|^2 \right]^2 + c_2 \langle |k_{\max}| \rangle^2 \sum_{\triangle} \frac{1}{S_\triangle^2} \left(\frac{dS_\triangle}{dt} \right)^2 \quad (5.1)$$

FIG. 8. The average (a) and maximum (b) absolute errors $\|\boldsymbol{n} - \boldsymbol{n}_{\text{exact}}\|$ in the normal vector calculation by BPS for a disk.

under the constraints $\boldsymbol{V}_i \cdot \boldsymbol{n}_i = q_i$, where the normal velocities q_i are given by the solution of the boundary-integral equations. The summations in (5.1) are over all mesh edges $\boldsymbol{x}_{ij} = \boldsymbol{x}_j - \boldsymbol{x}_i$ (with $i < j$) on S_α and over all mesh triangles $\triangle$ on S_α with areas $S_\triangle$; $\boldsymbol{n}_i^*$ is the normal at $\boldsymbol{x}_i$ calculated by locally fitting the best plane [3], $|k_{\max}| = \max(|k_1|, |k_2|)$ is the local maximum of two principal curvatures, and $\langle \cdots \rangle$ stands for the surface average. As explained in [3], the first term (5.1) is responsible for anisotropic mesh adaptation to high curvature, the second term (5.1) prevents the edges $\boldsymbol{x}_{ij}$ with low curvature along $\boldsymbol{x}_{ij}$ from excessive elongation, and the last term (5.1) resists triangle degeneration. Excessive adaptivity to high curvature zones may be disadvantageous, especially when these zones are rare and have a small effect on overall dynamics or macroscopic quantities of interest; for weak adaptivity, we set relatively high values for the constants $c_1 = 1$ and $c_2 = 2$. With some effort, F is expressed [3] as a quadratic function of $\{\boldsymbol{V}_i\}$ and minimized by conjugate gradient iterations [2]. Note that this method uses meshes with fixed topology and allows the elements to be highly stretched (which is natural, since the direction along a drop is typically the one of
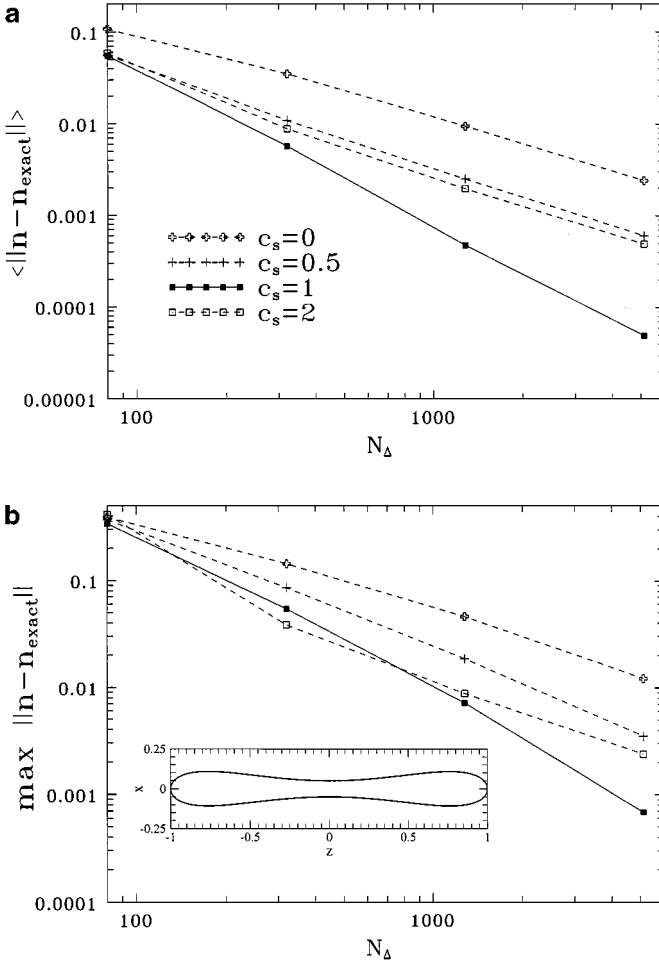
FIG. 9. The average (a) and maximum (b) absolute errors $\|\boldsymbol{n} - \boldsymbol{n}_{exact}\|$ in the normal vector calculation by BPS for the axisymmetrical spool-like shape shown in (b).

slow spatial variation of unknowns), in contrast to essentially isotropic techniques [40–42] of mesh restructuring into compact elements. Besides, our method seeks to minimize, in some sense, the "kinetic energy" of disordered mesh motion, thus avoiding excessive numerical stiffness inherent in the simplest grid tension approach.

5.2. Smoothing and Rescaling

A considerable difficulty of boundary-integral calculations for deformable drops is a possible catastrophic development of shape irregularities. These irregularities may be physical cusps [3] observed experimentally [43] (typically for $\lambda \ll 1$) or numerical artifacts. For our hybrid code, a finite precision ε of multipole truncations is one obvious source of artificial irregularities, due to jumps in the velocity calculations. These artificial irregularities are often amplified by local instabilities due to an insufficiently small time step. While the precision ε can be tightened with a modest effect on the CPU time (Section 6), integrations with small time steps would be too expensive. Instead, we follow the approach of [3] to add an artificial normal velocity to that provided by the boundary-integral solution at each

step to dynamically smooth irregularities. Unlike in [3], we use a weaker, differential form of smoothing which is most suitable for suppressing artificial irregularities. Namely, the normal velocity $q = \boldsymbol{v} \cdot \boldsymbol{n}$ on S_α is modified to

$$q(\boldsymbol{x}) + \varepsilon_s a^5 \langle |q| \rangle_\alpha \nabla_S^2 (k^3(\boldsymbol{x})) \tag{5.2}$$

in collocation nodes $\boldsymbol{x} = \boldsymbol{x}_i$. Here, $\varepsilon_s \ll 1$ is a smoothing parameter, $\langle |q| \rangle_\alpha$ is the average of $|q|$ over S_α, and ∇_S^2 is the surface Laplacian. The additional term (5.2) has a zero mean on S_α and is small, except in the irregularity regions. Strong sensitivity to the curvature and an odd power of k in (5.2) have a restoring effect irrespective of the sign of k and tend to suppress shape irregularities. It is possible to choose ε_s to facilitate long-time calculations with a negligible effect of smoothing on the overall dynamics (Section 6).

With k and $\boldsymbol{n}$ provided by BPS (Section 4), the smoothing term (5.2) is calculated as follows. We approximate $k^3(\boldsymbol{x}) - k^3(\boldsymbol{x}_i)$ at $\boldsymbol{x} \approx \boldsymbol{x}_i$ as a quadratic polynomial of the coordinates of $(\boldsymbol{x} - \boldsymbol{x}_i)$ in the tangential plane at $\boldsymbol{x}_i$. Five coefficients are found by least-square fitting to $k^3(\boldsymbol{x}_j) - k^3(\boldsymbol{x}_i)$ at the adjacent nodes $\boldsymbol{x}_j \in \mathcal{A}_i$, and the linear part gives $\nabla_S k^3(\boldsymbol{x}_i)$. With $\nabla_S k^3(\boldsymbol{x})$ known on the entire surface S_α, the flux of $\nabla_S k^3$ through a small closed contour of mesh triangles with the vertex $\boldsymbol{x}_i$ gives the divergence, $\nabla_S \cdot \nabla_S k^3$, at $\boldsymbol{x}_i$ and the smoothing term (5.2). This procedure has a crude, but adequate, accuracy for $\nabla_S^2 k^3$, since the additional term (5.2) is generally small and used only for smoothing.

We also reset the volumes of all drops to their initial value at each time step by shape rescaling at drop centroids, to avoid a long-time cumulative error. The necessary shape changes at each step are too small to produce artificial overlappings, even at high volume fractions. The effect of rescaling disappears altogether for fine triangulations but is expected to accelerate convergence. Interestingly, this rescaling, of frequent use in boundary-integral calculations, was found undesirable in our critical breakup study [3]. In the present work, however, most drops are far away from breakup conditions.

The choice of the time step is discussed in Appendix C.

6. NUMERICAL RESULTS

As the first test, we compared the sedimentation rate $|(v_S)_3|/U_o$ (where U_o is the settling velocity of an isolated drop) from (2.18) for static random configurations of spherical drops by the present code with the calculations of Mo and Sangani [35]. They used a purely multipole, $O(N^2)$ code for $N = 16$ drops in a cell with averaging over 20 random configurations and an extrapolation $N \to \infty$ by an effective-medium approximation (which is difficult to generalize for nonspherical drops considered below). We used $N = 64$–125, $N_\triangle = 1280$ triangular elements per drop, precision $\varepsilon = 5 \times 10^{-4} a^2$, threshold $k_o = 15$ and averaging over 20–70 configurations prepared by the Monte Carlo method (e.g., [44]) for the same "hard-sphere" distribution. Table I gives, for every λ and the drop volume fraction c, our results at $N = 64$ (the top value) and 125 (the bottom value), along with the calculations of Mo and Sangani [35] for $N = 16$ (the top value) and extrapolated to $N = \infty$ (the bottom value). Our results show a systemic, but very slight increase as N is changed from 64 to 125. For each pair of λ, c and $N = 64, 125$, a few configurations were selected to verify that the increase of $N_\triangle$ from 1280 to 2160 changed the sedimentation rate by less than 0.19%; tightening the precision ε to $10^{-4} a^2$ had even a much smaller

TABLE I

**The Average Sedimentation Rate for Static Random Suspensions of Spherical Drops**

	Present code $N = 64$ (top), $N = 125$ (bottom)			Mo and Sangani [35] $N = 16$ (top), $N \to \infty$ (bottom)	
c	$\lambda = 1$	10	c	$\lambda = 1$	10
0.25	0.299 ± 0.003	0.200 ± 0.002	0.25	0.250	0.190
	0.303 ± 0.002	0.203 ± 0.002		0.30	0.23
0.35	0.191 ± 0.003	0.110 ± 0.002	0.35	0.146	0.098
	0.194 ± 0.002	0.112 ± 0.001		0.17	0.11
0.45	0.117 ± 0.001	0.0577 ± 0.0007	0.45	0.088	0.051
	0.120 ± 0.001	0.0592 ± 0.0005		0.099	0.055

Note. The statistical errors for the present calculations correspond to the 67% confidence level.

influence (a more obscure accumulated effect of ε on long-time *dynamical* simulations is discussed below). For one configuration with $N = 64$, $c = 0.45$, $\lambda = 10$, and $N_\triangle = 1280$, we checked that using the exact curvature and normals for spheres changed our boundary-integral sedimentation rate by less than 0.0001%. Overall, our results are in a good agreement with the extrapolated values of Mo and Sangani [35]; minor differences may be due to larger statistical uncertainties for $N = 16$ than those for $N = 64, 125$ in Table I, and/or an approximate character of the effective-medium extrapolation $N \to \infty$ (as suggested by Fig. 5 of Sangani and Mo [11]). Without an extrapolation, however, small systems ($N = O(10)$) would be inadequate for sedimentation calculations.

Large N are even more important in dynamical simulations for sedimenting deformable drops, because of clustering. Figure 10 presents snapshots of our simulation for $\mathcal{B} = 1.75$, $c = 0.25$, $\lambda = 1$, $N = 125$, $N_\triangle = 1280$, $\varepsilon = 5 \times 10^{-4} a^2$, $k_o = 15$, $\varepsilon_s = 1.5 \times 10^{-5}$, and the time step parameter (Appendix C) $c_{\triangle t} = 1.85$ for different values of the microstructural time $\tilde{t} = ta$; only the drops with centroids in $[0, 1)^3$ are shown. As for Table I, the initial, "well-mixed" state ($\tilde{t} = 0$) of non-overlapping spherical drops was prepared by the Monte Carlo method [44]. Figure 11 shows the shape and mesh evolution for one chosen drop which experiences large deformations. It is seen that the passive mesh stabilization (Subsection 5.1) maintains the mesh quality for long-time simulations; by $\tilde{t} = 180$, a single drop would have fallen 48 radii. The simulation in Fig. 10 demonstrates the phenomenon of clustering of an initially homogeneous emulsion, leading to the formation of holes and a considerable increase in the sedimentation rate (see below).

Because of the large size of the code, it was crucial to test our hybrid scheme of calculating the inhomogeneous term (2.7a) $\boldsymbol{F}(\boldsymbol{y})$ for arbitrary shapes. The run in Fig. 10 was interrupted at $\tilde{t} = 144.6$ to compare our $\boldsymbol{F}(\boldsymbol{x}_i)$ with exact values $\boldsymbol{F}_{\mathrm{ex}}(\boldsymbol{x}_i)$ (for a given triangulation) obtained by standard point-to-point summations in (3.4a) and (3.8a), i.e., without drop partitioning into blocks and multipoles. In the point-to-point scheme, $\boldsymbol{G}(\boldsymbol{x}_j - \boldsymbol{y})$ was split into $(\boldsymbol{G}_0 + \boldsymbol{G}_1)(\boldsymbol{x}_j + \boldsymbol{k} - \boldsymbol{y})$, with $\boldsymbol{x}_j + \boldsymbol{k} - \boldsymbol{y} \in [-\frac{1}{2}, \frac{1}{2})^3$ and the integer $\boldsymbol{k}$, and the smooth part $\boldsymbol{G}_1$ was calculated as the quadratic Taylor approximation at the nearest node of a $161 \times 161 \times 161$ mesh in $[-\frac{1}{2}, \frac{1}{2}]^3$ through tabulated derivatives of $\boldsymbol{G}_1$ to the second order in $[0, \frac{1}{2}]^3$ and symmetry properties; this procedure gives $\boldsymbol{G}(\boldsymbol{x}_j - \boldsymbol{y})$ without any appreciable
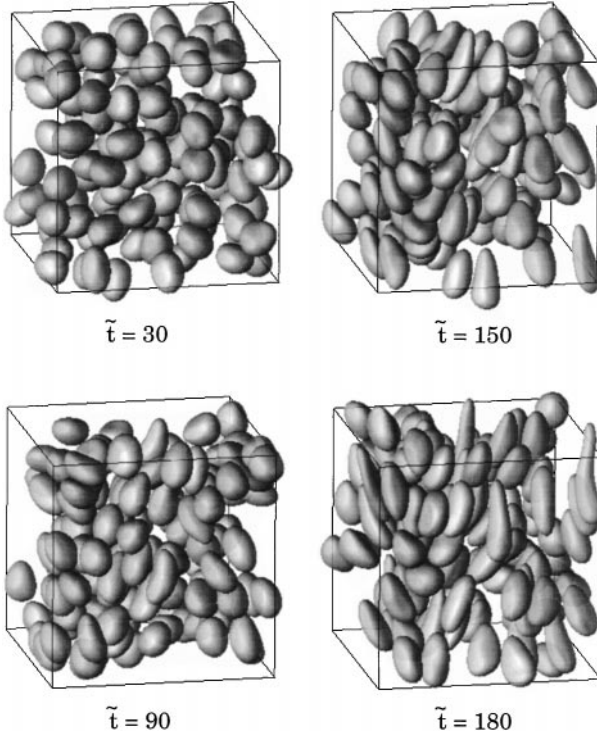
FIG. 10. Snapshots of the dynamical simulation from a homogeneous initial state of spherical drops for $\mathcal{B} = 1.75$, $c = 0.25$, $\lambda = 1$, $N = 125$, $N_{\triangle} = 1280$, $\varepsilon = 0.0005a^2$, $k_o = 15$, $\varepsilon_s = 1.5 \times 10^{-5}$, and $c_{\triangle t} = 1.85$. The drops sediment downwards.

error. We considered three criteria to quantify the difference between F and F_{ex},

$$\delta_1(F, F_{ex}) = \frac{1}{\left\langle F_{ex}^2 \right\rangle^{1/2}} \max_{\alpha, x_i \in S_\alpha} \| F(x_i) - F_{ex}(x_i) \|, \tag{6.1a}$$

$$\delta_2(F, F_{ex}) = \frac{1}{N} \sum_{\alpha=1}^{N} \left\{ \frac{1}{\left\langle F_{ex}^2 \right\rangle_\alpha} \max_{x_i \in S_\alpha} [F(x_i) - F_{ex}(x_i)]^2 \right\}^{1/2}, \tag{6.1b}$$

$$\delta_3(F, F_{ex}) = \left[\left\langle (F - F_{ex})^2 \right\rangle / \left\langle F_{ex}^2 \right\rangle \right]^{1/2}. \tag{6.1c}$$

Here, $\langle \cdots \rangle_\alpha$ and $\langle \cdots \rangle$ denote averaging over S_α and all surfaces, respectively.

Table II demonstrates that all δ_1, δ_2, $\delta_3 \to 0$ as $\varepsilon \to 0$, as a check of the convergence of our results to those by the standard method. It takes our code only 60 and 84 seconds of CPU time on a DEC 500au to calculate all the boundary integrals (2.7a) for $\varepsilon = 5 \times 10^{-4}a^2$ and $5 \times 10^{-5}a^2$, respectively (Table II); these times are much faster than 95 min for the standard optimized point-to-point summation code, even with a linear interpolation for the smooth part G_1 of Green's function. Thus, without any significant loss of accuracy, two-orders-of-magnitude advantage is gained by the present code for $N = O(100)$. The limit F_{ex} is achieved, as $\varepsilon \to 0$, irrespective of k_o, but using high-order multipoles is essential at the advanced stage of drop deformation. For example, when k_o is reduced to 6, the calculation of boundary integrals (2.7a) for $\varepsilon = 5 \times 10^{-4}a^2$ slows down almost three times, because the multipole expansions, when insufficient, are replaced by costly direct summations; large k_o
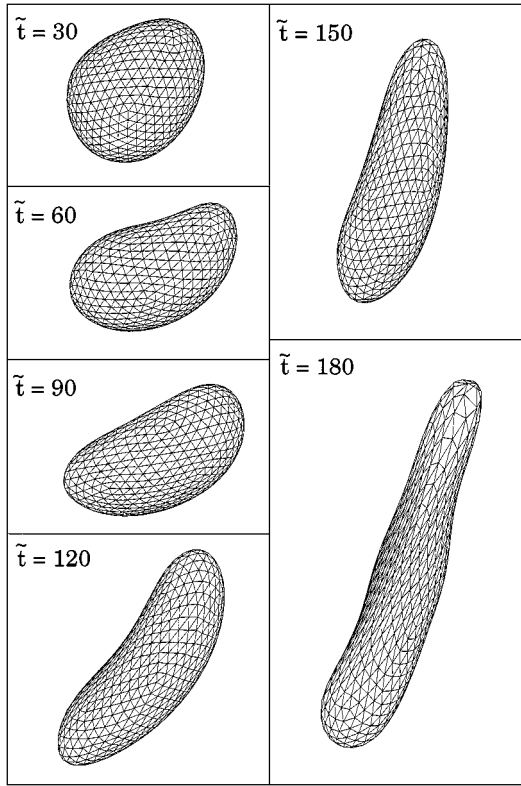
FIG. 11. The mesh and shape evolution for one chosen drop with large deformations from the simulation shown in Fig. 10. The plane of view contains the line of maximum elongation (with $\max\|x_i - x_j\|$) and the vertical.

are even more important for higher precisions. The range $k_o = 15$–20 was found optimal in this and the following simulations. The total CPU time per time step slightly grows (by 30%) from the beginning to the end of the simulation in Fig. 10, mainly due to the increase in the number of blocks $\mathcal{B}_\gamma$ (209 at $\tilde{t} = 144.6$) and clustering. The whole run in Fig. 10 took 2000 second-order Runge–Kutta steps (with two-fold boundary-integral solutions at each step) and about four days on a DEC 500au, with BPS and mesh stabilization parts contributing roughly 20% each; for a standard, point-to-point summation of interactions, this simulation would take nine months.

TABLE II

The Convergence of the Present Solution F to the Standard $O(N^2 N_\triangle^2)$ Solution F_{ex} as $\varepsilon \to 0$, in the Single-Layer Test for $N = 125$, $N_\triangle = 1280$

ε	$\delta_1 (F, F_{ex})$	$\delta_2 (F, F_{ex})$	$\delta_3 (F, F_{ex})$	CPU time (s)
$5 \times 10^{-2} a^2$	3.46×10^{-1}	1.63×10^{-1}	4.52×10^{-2}	—
$5 \times 10^{-3} a^2$	1.44×10^{-1}	4.88×10^{-2}	1.18×10^{-2}	48
$5 \times 10^{-4} a^2$	2.38×10^{-2}	9.79×10^{-3}	2.27×10^{-3}	60
$5 \times 10^{-5} a^2$	7.20×10^{-3}	1.67×10^{-3}	3.70×10^{-4}	81
$5 \times 10^{-6} a^2$	6.54×10^{-4}	2.22×10^{-4}	5.24×10^{-5}	112

*Note.* The CPU times (in seconds) are for the calculation of F by the present method.
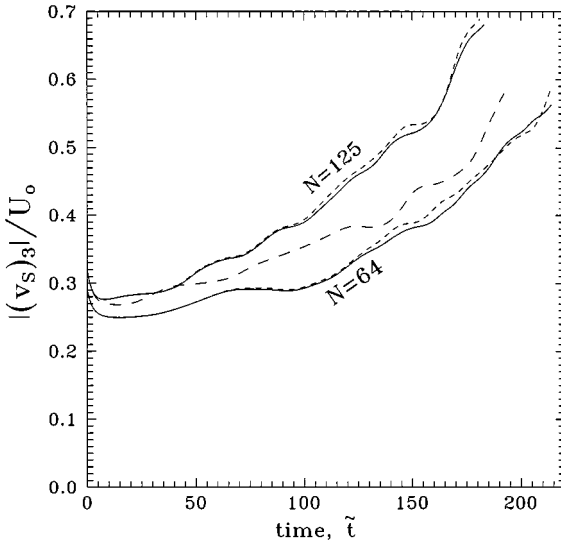
FIG. 12. The effect of the precision parameter ε on the sedimentation rate for individual realizations at $\mathcal{B} = 1.75$, $c = 0.25$, $\lambda = 1$, and $N_\triangle = 1280$. The upper two curves are for a single initial realization of $N = 125$ drops ($c_{\Delta t} = 1.85$); the bottom two curves start from another single initial realization of $N = 64$ drops ($c_{\Delta t} = 1.7$). Solid lines are for $\varepsilon = 5 \times 10^{-5} a^2$, $k_o = 20$; short-dashed lines are for $\varepsilon = 5 \times 10^{-4} a^2$, $k_o = 15$. The short-dashed line for $N = 125$ corresponds to the simulation in Fig. 10. The long-dashed line is for another random initial configuration with $N = 125$ and $\varepsilon = 5 \times 10^{-4} a^2$.

We have studied the effects of precision, smoothing, and triangulation on the sedimentation rate in *dynamical* simulations by repeating the run in Fig. 10 from $\tilde{t} = 0$ with different ε, ε_s, $N_\triangle$, and by taking another initial random realization of spherical drops with $c = 0.25$, $N = 64$ (initial configurations are available from the authors). Increasing the smoothing parameter ε_s from 1.5×10^{-5} to 4.5×10^{-5} did not show any appreciable effects in the studied time range. Small differences between the results for $\varepsilon = 5 \times 10^{-4} a^2$ (short-dashed lines) and $5 \times 10^{-5} a^2$ (solid lines) in Fig. 12 suggest that $\varepsilon \leq O(10^{-4} a^2)$ gives a practically precision-independent sedimentation rate for $\tilde{t} < 180$. With a crude precision $\varepsilon = 5 \times 10^{-3} a^2$, the run for $N = 125$, $N_\triangle = 1280$, $c_{\Delta t} = 1.85$ failed early at $\tilde{t} = 83$ due to instability. Compared to the run in Fig. 10, the simulation for $N = 64$, $\varepsilon = 5 \times 10^{-4} a^2$, $N_\triangle = 1280$, $c_{\Delta t} = 1.7$, with 2000 time steps required to reach $\tilde{t} \approx 191$, took about 44 h on a DEC 500au, showing an $O(N)$, rather than $O(N^2)$ scaling for moderately large $N = 64 - 125$. The runs for $\varepsilon = 5 \times 10^{-5} a^2$ were only slightly (15–20%) slower than those for $\varepsilon = 5 \times 10^{-4} a^2$. The difference between the solid lines for $N = 64$ and 125 in Fig. 12 does not necessarily mean slow convergence of the results, as $N \to \infty$. There is an additional dependence on the initial configuration (as illustrated by a long-dashed line for $N = 125$ in Fig. 12), and the results of many initial configurations should be averaged before taking the limit $N \to \infty$; this task, however, goes beyond the scope of the present paper.

Figures 13a, 13b show the sensitivity of the precision-independent results ($\varepsilon = 5 \times 10^{-5} a^2$) to surface triangulations. When BPS is used for the curvatures and normals with $c_s = 1$ (solid and short-dashed lines in Figs. 13a, 13b), the results for $N_\triangle = 1280$ and 720 are in a remarkable agreement up to large times, both for $N = 125$ and 64, even though the sedimentation rate is a strong and complex function of time for individual realizations. In contrast, for
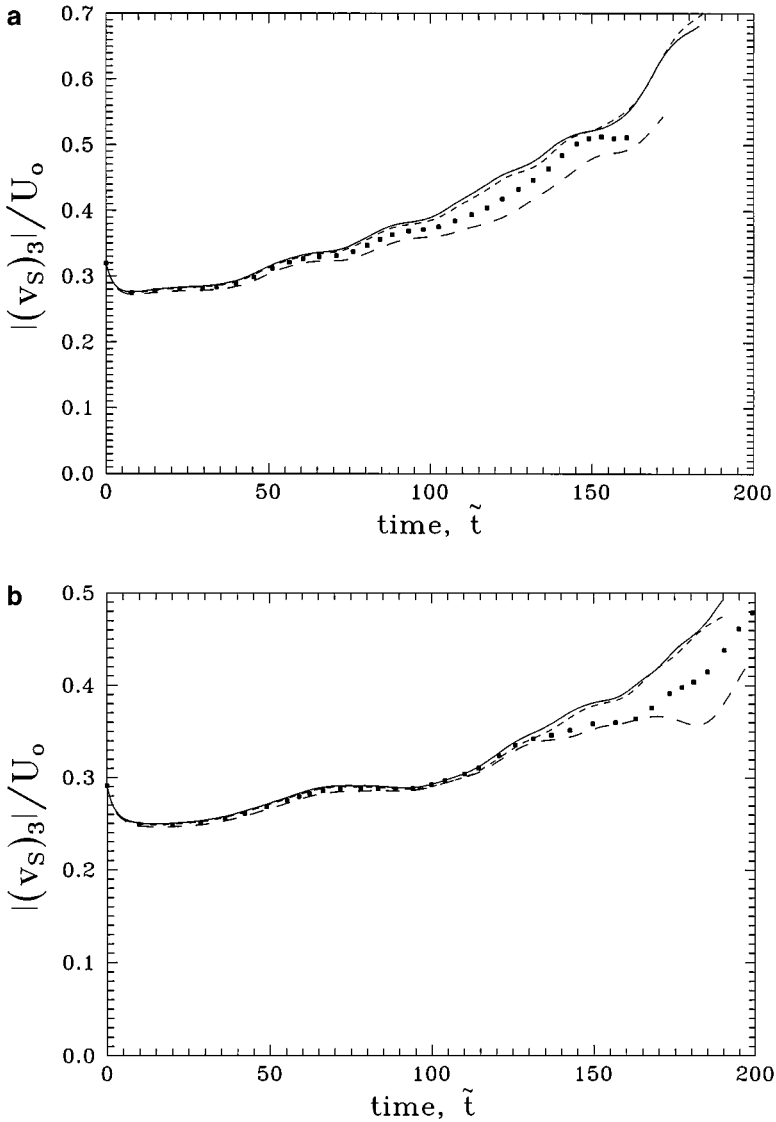
FIG. 13. The effect of surface triangulations on the sedimentation rate for individual realizations at $\mathcal{B} = 1.75$, $c = 0.25$, $\lambda = 1$, and $\varepsilon = 5 \times 10^{-5} a^2$. The initial configurations with $N = 125$ (a) and $N = 64$ (b) are identical to those for the two upper curves ($N = 125$) and the two bottom curves ($N = 64$) in Fig. 12. Solid ($N_\Delta = 1280$) and short-dashed ($N_\Delta = 720$) lines are for the simulations using BPS. Dark squares ($N_\Delta = 1280$) and long-dashed lines ($N_\Delta = 720$) are for the simulations using the local best paraboloid method.

the local best-paraboloid method ($c_s = 0$), the calculations for $N_\Delta = 1280$ and 720 (dark squares and long-dashed lines in Figs. 13a, 13b) are convergent only in a more limited time range.

An initial decrease of $|(v_S)_3|/U_o$ in Figs. 13a, 13b is not accidental. At small times, drop deformation from the initial spherical shapes plays a minor role. On the $O(c^2)$ level of pair interactions, a *flowing* emulsion of spherical drops was shown to be more viscous than in the well-mixed state (without the Brownian contribution to the stress) [45]; accordingly,
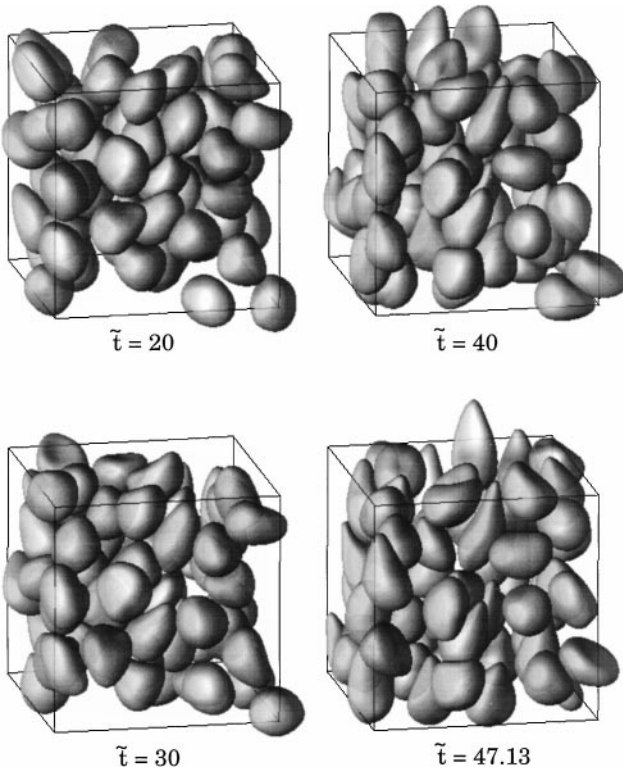
FIG. 14. Snapshots of the dynamical simulation from a homogeneous initial state of spherical drops for $\mathcal{B} = 2.5$, $c = 0.4$, $\lambda = 0.25$, $N = 64$, $N_\triangle = 2160$, $\varepsilon = 5 \times 10^{-5}a^2$, $k_o = 20$, $\varepsilon_s = 1.5 \times 10^{-5}$, and $c_{\triangle t} = 0.5$. The drops sediment downwards.

such an emulsion is expected to sediment slower, which is confirmed by Figs. 13a, 13b. Further drop motion leads to stretching, with a strong preference for almost vertically aligned prolate shapes. However, the deformation itself cannot account for a strong increase of the sedimentation rate. For example, the model of a vertically alligned solid spheroid with the deformation equivalent to the average deformation in Fig. 10 at $\tilde{t} = 180$ gives the settling velocity of an isolated drop as no more than 4% higher than for a solid sphere of the same volume (e.g., [46]). The actual increase of $|(v_S)_3|/U_o$ is much larger and is due to clustering, an indirect consequence of drop deformations.

A considerably more difficult simulation is presented in Fig. 14 for $c = 0.4$, $\mathcal{B} = 2.5$, $\lambda = 0.25$, $N = 64$, $N_\triangle = 2160$, $k_o = 20$, $\varepsilon_s = 1.5 \times 10^{-5}$, and $\varepsilon = 5 \times 10^{-5}a^2$. Unlike for $c = 0.25$ and $\lambda = 1$ (Fig. 10), much smaller time steps had to be used to maintain the stability ($c_{\triangle t} = 0.5$). The iterative solutions of (2.12) were terminated once $\delta_2(w^{(\nu+1)}, w^{(\nu)}) < 10^{-3}$ for two consecutive iterations $w^{(\nu)}$, $w^{(\nu+1)}$, with δ_2 being a compromise between the conservative (6.1a) and loose (6.1c) criteria; typically, 3–4 iterations sufficed. A similar run was performed for $N_\triangle = 1280$ to $\tilde{t} = 45.4$. In these simulations, the average gap between a drop and its neighbors was estimated as $0.03a$ for any $\tilde{t} > 35$; the minimum gap was much smaller, just a few thousandths of a. Despite such small separations, our runs could proceed without any artificial subgrid lubrication or repulsive forces. Instead, it is the physical development of sharp tails (Fig. 15) which caused the calculations to stall. Despite this difficulty, our numerical solutions are capable of describing strong temporal
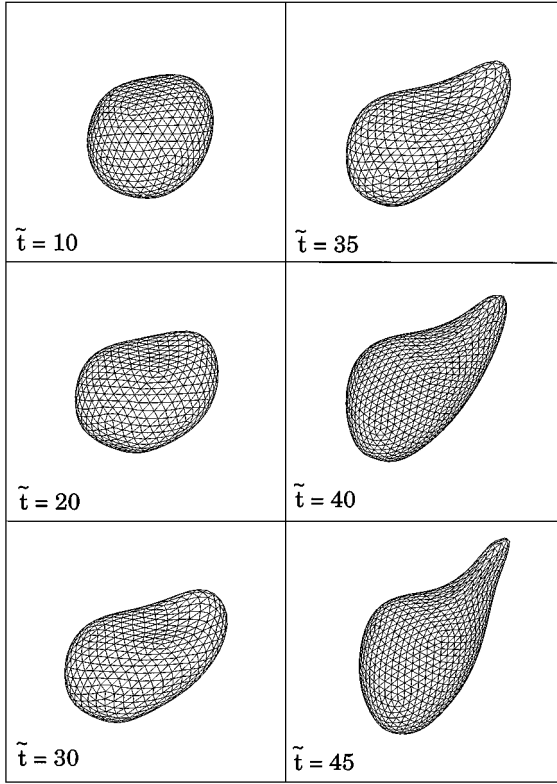
FIG. 15. The mesh and shape evolution for one chosen drop from the simulation shown in Fig. 14 and a similar simulation with $N_\triangle = 1280$. Frames $\tilde{t} = 10$ to 35 are for $N_\triangle = 1280$; frames $\tilde{t} = 40$ and 45 are for $N_\triangle = 2160$. The plane of view contains g and the line of maximum elongation.

changes of the sedimentation rate, about twofold from its minimum (Fig. 16); the results for $N_\triangle = 2160$ (a solid line) and 1280 (a dashed line) are in a good agreement. An additional run was performed for $N_\triangle = 1280$ and a cruder precision $\varepsilon = 2 \times 10^{-4} a^2$ to $\tilde{t} = 43.6$, but showed a much smaller accumulated effect of ε on the sedimentation rate (within 0.25% up to $\tilde{t} = 43.6$) than in the simulations for $c = 0.25$, $\lambda = 1$, $\mathcal{B} = 1.75$ (see above); the difference between the two curves in Fig. 16 is due to triangulation error.

To check our hybrid scheme for the double-layer term (2.12b), the run for $N_\triangle = 1280$ and $\varepsilon = 2 \times 10^{-4} a^2$ in Fig. 16 was interrupted at $\tilde{t} = 38.3$ (with 115 blocks) to compare one iteration of (2.12b) by our code at different precisions ε in the double-layer truncations (Subsection 3.5) with one iteration calculated in a standard point-to-point manner (using, again, a quadratic Taylor approximation from the table for the smooth stresslet part, $\boldsymbol{\tau}_1$). In all these tests, $\boldsymbol{F}(\boldsymbol{y})$ was fixed from our hybrid calculation with $\varepsilon = 2 \times 10^{-4} a^2$, since this part of the code was already verified for $\lambda = 1$; the inhomogeneous term (2.7a) and BPS calculations required 44 and 8–9 s, respectively, on a DEC 500au. The values of $\delta_1(\boldsymbol{v}, \boldsymbol{v}_{\mathrm{ex}})$, $\delta_2(\boldsymbol{v}, \boldsymbol{v}_{\mathrm{ex}})$, and $\delta_3(\boldsymbol{v}, \boldsymbol{v}_{\mathrm{ex}})$ (where $\boldsymbol{v}$ and $\boldsymbol{v}_{\mathrm{ex}}$ are the velocities (2.17) by our code and the standard scheme, respectively) in Table III confirm the convergence $\boldsymbol{v} \to \boldsymbol{v}_{\mathrm{ex}}$ for $\varepsilon \to 0$. In this test, one velocity iteration of our code required 56 and 80 s at $\varepsilon = 5 \times 10^{-4} a^2$ and $5 \times 10^{-5} a^2$, respectively, on a DEC 500au; for $\tilde{t} = 0$, when the drops are spherical, these CPU times are even smaller (27 and 38 s at $\varepsilon = 5 \times 10^{-4} a^2$ and $5 \times 10^{-5} a^2$, respectively). In
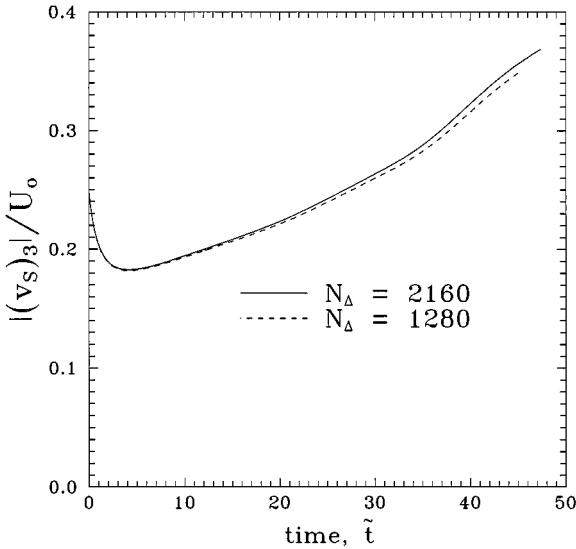
FIG. 16. The sedimentation rate vs time for the simulation shown in Fig. 14 (solid line) and a similar simulation with $N_\triangle = 1280$ (dashed line).

contrast, for these $N = 64$ and $N_\triangle = 1280$, one iteration by the standard $O(N^2 N_\triangle^2)$ method takes 63 min at every $\tilde{t}$, even with the simplest, linear interpolation for τ_1. To reach the same $\tilde{t} = 45$, the runs for $N_\triangle = 1280$, $\varepsilon = 5 \times 10^{-5} a^2$ and $N_\triangle = 2160$, $\varepsilon = 5 \times 10^{-5} a^2$ took 2500 and 3400 second-order Runge–Kutta steps, and 15 and 38 days on a DEC 500au, respectively; the CPU scaling per time step is closer to $O(N_\triangle)$, rather than $O(N_\triangle^2)$, in these simulations. Although these computer expenses appear large, a standard point-to-point method would require about 2.5 and 9 years for the same runs with $N_\triangle = 1280$ and 2160, respectively, to $\tilde{t} = 45$!

Koch and Shaqfeh [47] predicted analytically, on the pairwise level, the instability of a dilute suspension of sedimenting solid spheroids. We attribute the nonexistence of a steady state in our simulations to some qualitative analogy between the two systems. Owing to this unsteady character of the problem, ensemble averaging over many initial realizations is essential before taking the limit $N \to \infty$, and the average sedimentation rate should be studied as a function of time. Our systematic calculations of this kind, relatively easy for $\lambda = O(1)$, will be published elsewhere.

TABLE III
The Convergence of the Present Solution v to the Standard $O(N^2 N_\triangle^2)$ Solution v_{ex}, as $\varepsilon \to 0$, in the Double-Layer Test (One Velocity Iteration) for $N = 64$, $N_\triangle = 1280$

ε	$\delta_1 (v, v_{ex})$	$\delta_2 (v, v_{ex})$	$\delta_3 (v, v_{ex})$	CPU time (s)
$1.25 \times 10^{-1} a^2$	2.86×10^{-1}	1.28×10^{-1}	2.98×10^{-2}	26
$1.25 \times 10^{-2} a^2$	5.57×10^{-2}	3.44×10^{-2}	7.58×10^{-3}	33
$1.25 \times 10^{-3} a^2$	1.18×10^{-2}	6.73×10^{-3}	1.42×10^{-3}	47
$5.00 \times 10^{-4} a^2$	6.21×10^{-3}	3.28×10^{-3}	6.93×10^{-4}	56
$7.50 \times 10^{-5} a^2$	1.52×10^{-3}	6.61×10^{-4}	1.42×10^{-4}	75

Note. The CPU times (in seconds) are for one iteration by the present method ($k_o = 15$–20).

7. CONCLUSIONS

The efficient 3D algorithm has been constructed for hydrodynamical interaction of many deformable drops at zero Reynolds numbers with triply periodic boundaries and subject to gravity. The algorithm is a hybrid of boundary-integral and economical multipole techniques, with extensive use of rotational transformations to optimize near-field summations. Very high, two-order-of-magnitude gains in computational speed over the standard boundary-integral techniques are achieved with the present method, even for a moderately large number of drops $N = O(10^2)$ with surface resolutions of $N_\triangle \sim 10^3$ triangular elements per drop. Also a part of the code is the new, best paraboloid-spline technique (BPS) for the normal vectors and curvatures on unstructured triangulations, greatly improving the quality of long-time dynamical simulations. Our systematic dynamical calculations of the sedimentation rate for viscosity ratios $\lambda = O(1)$ and $N \leq 200$ using this code are in progress. Attempts will be made to expedite calculations in the most difficult case $\lambda \ll 1$, by improving the simplest near-singularity subtraction (3.9a)–(3.9b) (to allow for limited triangulations) and using Lanczos' biconjugate gradient iterations [2]. With minor changes, our code is applicable to moderately polydisperse systems.

Since the present or similar 3D hydrodynamical problems have been unchallenged by other fast multiparticle strategies, it is difficult to make firm judgements about their possible performance in our case. We can only note that the new 3D FMM [26] has shown about 29- and 53-fold advantages over direct summations for 40,000 and 80,000 charges, respectively, in the case of Coulombic, free-space interactions with random uniform distributions in a cube and moderately high accuracy (Table 2 of [26]). For a similar actual accuracy, we have observed, with our code, about 66- and 71-fold advantages over direct summations in the double-layer and single-layer tests for $(N_\triangle/2 + 2)N = 41{,}088$ and 80,250 collocation nodes, respectively, for the more involved case of Stokesian interactions with periodic boundaries at the advanced stage of drop deformation (Section 6). These gains are even higher for nearly spherical drops. Although it is attractive to test different schemes for multidrop interactions, the present technique may be close to optimal for up to several hundred drops with $N_\triangle \sim 10^3$. Larger systems ($N \sim 10^3$) would likely require additional ideas to avoid $O(N^2)$ scaling, namely, the FMM merging of singularities or P^{3}M schemes. Such 3D systems of drops with adequate resolution $N_\triangle \sim 10^3$, however, would be too prohibitive for dynamical simulations on present-day computers. Also, if each drop had to be discretized by a very large number of elements ($N_\triangle \geq O(10^4)$), a faster scheme would be needed for self-interactions; this resolution, however, would limit the present-day dynamical simulations to a few drops only.

Most of our code is applicable to study the rheological behavior of many drops in a shear flow, rather than subject to gravity; the far-field part (Subsection 3.4), however, presents an additional challenge, since the basic periodic cell changes with time. The physical behavior of drops in the two cases is surprisingly different. For the same volume fraction, the effect of interactions on sedimentation is much stronger than on the rheology in a shear flow. With deformation, sedimenting drops cluster (which would make simulations for $N = O(10)$ meaningless); in contrast, calculations [5] for several drops at small and moderate volume fractions indicate the absence of clustering in a shear flow, making the use of large N in this case less imperative. However, for highly concentrated 3D sheared emulsions, close to liquid foams, large N are needed, at least, to avoid ergodic difficulties, and our techniques can handle such systems.

APPENDIX A

The Construction of Minimal Spherical Shells

We use the simplest stochastic procedure to construct the minimal spherical shell $\mathcal{D}_\gamma$ around $\mathcal{B}_\gamma$ with sufficient accuracy. An initial approximation $(x_\gamma^o)^{(0)}$ to the shell center, x_γ^o, is the midpoint of the block diameter. If $(x_\gamma^o)^{(\nu)}$ is the νth iteration of x_γ^o, and $(d_\gamma^o)^{(\nu)} = \max|x_j - (x_\gamma^o)^{(\nu)}|$ over $x_j \in \mathcal{B}_\gamma$ is the radius of the minimal shell around $\mathcal{B}_\gamma$ with the center $(x_\gamma^o)^{(\nu)}$, then the improvement of $(x_\gamma^o)^{(\nu)}$ is tried as

$$\left(x_\gamma^o\right)^{(\nu+1)} = \left(x_\gamma^o\right)^{(\nu)} + \xi|\mathcal{B}_\gamma|^{-1/2}\left[x_{j*} - \left(x_\gamma^o\right)^{(\nu)}\right], \tag{A.1}$$

where $\xi \in (0, 1)$ is a random number, $|\mathcal{B}_\gamma|$ is the number of nodes in $\mathcal{B}_\gamma$, and $|x_{j*} - (x_\gamma^o)^{(\nu)}| = (d_\gamma^o)^{(\nu)}$, $x_{j*} \in \mathcal{B}_\gamma$; displacement (A.1) is accepted if $(d_\gamma^o)^{(\nu+1)} \leq (d_\gamma^o)^{(\nu)}$, while otherwise a new random number ξ is tried, etc. With $|\mathcal{B}_\gamma|/2$ Monte Carlo steps (accepted or not), the minimal shell construction for all $\mathcal{B}_\gamma$ takes a negligible portion of total expenses.

APPENDIX B

Fast Pointwise Calculation of Lamb's Series

The form (3.19) for Lamb's regular series (3.17) simply follows from the fact that (3.17) minus the first term of (3.19) is a harmonic field (cf. with Subsection 3.4). Explicit expressions for Cartesian components $(H_{n,m})_k$ of $H_{n,m}$ are obtained upon substituting (3.18) into (3.17), and using (3.36), (3.41) with R^n instead of $1/R^{n+1}$, (3.63), and additional recurrent relations

$$(D_1 \pm iD_2)\left[\rho^n Y_{n,m}(\rho)\right] = \mp\left[\frac{(2n+1)(n \mp m - 1)(n \mp m)}{2n-1}\right]^{1/2}\rho^{n-1}Y_{n-1,m\pm1}(\rho). \tag{B.1}$$

The result takes the form

$$\begin{aligned}
(H_{n,m})_1 &= \frac{(n+2)d_\delta^o}{4n(2n+1)}(\alpha_{n-1,-m-1}A_{n-1,m+1} - \alpha_{n-1,m-1}A_{n-1,m-1}) \\
&\quad + \frac{1}{2d_\delta^o}(\alpha_{-n-2,-m-1}B_{n+1,m+1} - \alpha_{-n-2,m-1}B_{n+1,m-1}) \\
&\quad + \frac{i}{2}(\gamma_{n,m-1}C_{n,m-1} + \gamma_{n,-m-1}C_{n,m+1}),
\end{aligned} \tag{B.2a}$$

$$\begin{aligned}
(H_{n,m})_2 &= \frac{i(n+2)d_\delta^o}{4n(2n+1)}(\alpha_{n-1,-m-1}A_{n-1,m+1} + \alpha_{n-1,m-1}A_{n-1,m-1}) \\
&\quad + \frac{i}{2d_\delta^o}(\alpha_{-n-2,-m-1}B_{n+1,m+1} + \alpha_{-n-2,m-1}B_{n+1,m-1}) \\
&\quad + \frac{1}{2}(\gamma_{n,m-1}C_{n,m-1} - \gamma_{n,-m-1}C_{n,m+1}),
\end{aligned} \tag{B.2b}$$

$$(H_{n,m})_3 = -\frac{(n+2)d_\delta^o}{2n(2n+1)}\beta_{n-1,m}A_{n-1,m} + \frac{\beta_{-n-2,m}}{d_\delta^o}B_{n+1,m} - imC_{n,m}, \tag{B.2c}$$

where, for brevity,

$$\alpha_{n,m} = \left[\frac{(2n+1)(n+m+1)(n+m+2)}{2n+3}\right]^{1/2},$$

$$\beta_{n,m} = \left[\frac{(2n+1)(n+1-m)(n+1+m)}{2n+3}\right]^{1/2}, \qquad (B.3)$$

$$\gamma_{n,m} = [(n-m)(n+m+1)]^{1/2},$$

and the terms with $A_{n',m'}$ and $C_{n',m'}$ for $n' < 0$ or $|m'| > n$ are omitted.

Similarly, using (3.31), (3.36), and (3.41), Lamb's singular series (3.12) can be written in the form (3.21), with

$$
\begin{aligned}
(F_{n,m})_1 = {} & \frac{(n-1)d_\gamma^o}{4(n+1)(2n+1)}\left[\alpha_{-n-2,m-1}A^{(\gamma)}_{-(n+2),m-1} - \alpha_{-n-2,-m-1}A^{(\gamma)}_{-(n+2),m+1}\right] \\
& + \frac{1}{2d_\gamma^o}\left[\alpha_{n-1,-m-1}B^{(\gamma)}_{-n,m+1} - \alpha_{n-1,m-1}B^{(\gamma)}_{-n,m-1}\right] \\
& + \frac{i}{2}\left[\gamma_{n,m-1}C^{(\gamma)}_{-(n+1),m-1} + \gamma_{n,-m-1}C^{(\gamma)}_{-(n+1),m+1}\right], \qquad (B.4a)
\end{aligned}
$$

$$
\begin{aligned}
(F_{n,m})_2 = {} & -\frac{i(n-1)d_\gamma^o}{4(n+1)(2n+1)}\left[\alpha_{-n-2,m-1}A^{(\gamma)}_{-(n+2),m-1} + \alpha_{-n-2,-m-1}A^{(\gamma)}_{-(n+2),m+1}\right] \\
& + \frac{i}{2d_\gamma^o}\left[\alpha_{n-1,-m-1}B^{(\gamma)}_{-n,m+1} + \alpha_{n-1,m-1}B^{(\gamma)}_{-n,m-1}\right] \\
& + \frac{1}{2}\left[\gamma_{n,m-1}C^{(\gamma)}_{-(n+1),m-1} - \gamma_{n,-m-1}C^{(\gamma)}_{-(n+1),m+1}\right], \qquad (B.4b)
\end{aligned}
$$

$$(F_{n,m})_3 = -\frac{(n-1)d_\gamma^o}{2(n+1)(2n+1)}\beta_{-(n+2),m}A^{(\gamma)}_{-(n+2),m} - \frac{\beta_{n-1,m}}{d_\gamma^o}B_{-n,m} - imC^{(\gamma)}_{-(n+1),m}. \qquad (B.4c)$$

APPENDIX C

Time-Step Strategy

It is difficult to set rational and universal rules for the time step, especially because the present problem does not have a statistically steady-state solution, and so the optimum time step strategy would depend on the time range of interest. We found the common stability criterion (e.g., [30]) $\Delta t < K\Delta x$ (where Δx is the minimum mesh edge) oversimplified, since the optimum factor K depends on shapes and meshes. In the present calculations, the nondimensional time step (based on the length and velocity scales of Section 2) was chosen empirically as

$$\Delta t = c_{\Delta t}(1+\lambda)\mathcal{B}\min(\Delta_1, \Delta_2)/a^3, \qquad (C.1)$$

where $c_{\Delta t} = O(1)$ is a numerical factor,

$$\Delta_1 = \min_i \frac{\Delta x_i}{\max(|k_1(i)|, |k_2(i)|)}, \qquad \Delta_2 = \min_i \frac{2h_i^{1/2}\Delta x_i}{\left[k_1^2(i) + k_2^2(i)\right]^{1/2}}, \qquad (C.2)$$

Δx_i is the minimum of mesh edges with vertex i, h_i is the minimum of distances from node i to nodes on other surfaces, $k_1(i)$ and $k_2(i)$ are the principal curvatures at node i, and the minima in (C.2) are taken over all nodes i. The factor $c_{\Delta t}$ is largely independent of N and triangulations, but is strongly sensitive to the volume fraction.

ACKNOWLEDGMENTS

REFERENCES

1. C. Pozrikidis, *Boundary Integral and Singularity Methods for Linearized Viscous Flow* (Cambridge Univ. Press, Cambridge, MA, 1992).

2. A. Z. Zinchenko, M. A. Rother, and R. H. Davis, A novel boundary-integral algorithm for viscous interaction of deformable drops, *Phys. Fluids* **9**, 1493 (1997).

3. A. Z. Zinchenko, M. A. Rother, and R. H. Davis, Cusping, capture and breakup of interacting drops by a curvatureless boundary-integral algorithm, *J. Fluid Mech.* **391**, 249 (1999).

4. M. Manga, Interactions between mantle diapirs, *Geophys. Res. Lett.* **24**, 1871 (1997).

5. M. Loewenberg and E. J. Hinch, Numerical simulation of a concentrated emulsion in shear flow, *J. Fluid Mech.* **321**, 395 (1996).

6. X. Li, R. Charles, and C. Pozrikidis, Simple shear flow of suspensions of liquid drops, *J. Fluid Mech.* **320**, 395 (1996).

7. R. Charles and C. Pozrikidis, Significance of the dispersed-phase viscosity on the simple shear flow of suspensions of two-dimensional liquid drops, *J. Fluid Mech.* **365**, 205 (1998).

8. A. J. C. Ladd, Sedimentation of homogeneous suspensions of non-Brownian spheres, *Phys. Fluids* **9**, 491 (1997).

9. A. J. C. Ladd, Numerical simulations of particulate suspensions via a discretized Boltzmann equation. Part I. Theoretical foundation, *J. Fluid Mech.* **271**, 285 (1994).

10. A. J. C. Ladd, Numerical simulations of particulate suspensions via a discretized Boltzmann equation. Part II. Numerical results, *J. Fluid Mech.* **271**, 311 (1994).

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

12. L. Greengard and V. Rokhlin, Rapid evaluation of potential fields in three dimensions, in *Vortex Methods*, Lecture Notes in Mathematics, edited by C. Anderson and C. Greengard (Springer-Verlag, New York/Berlin, 1988), Vol. 1360, p. 121.

13. A. S. Sangani and G. Mo, Inclusion of lubrication forces in dynamic simulations, *Phys. Fluids* **6**, 1653 (1994).

14. A. Z. Zinchenko, Efficient numerical solution of the electrostatic problem for a randomly periodic fibrous composite dielectric, *Sov. Phys. Tech. Phys.* **34**, 15 (1989).

15. A. Z. Zinchenko, Effective dielectric constant of concentrated fiber composites studied by direct numerical simulation, *Sov. Phys. Tech. Phys.* **35**, 1236 (1990).

16. A. Z. Zinchenko, An efficient algorithm for calculating multiparticle thermal interaction in a concentrated dispersion of spheres, *J. Comput. Phys.* **111**, 120 (1994).

17. A. Z. Zinchenko, Effective conductivity of loaded granular materials by numerical simulation, *Philos. Trans. R. Soc. London A* **356**, 2953 (1998).

18. L. C. Biedenharn and J. D. Louck, *Angular Momentum in Quantum Physics: Theory and Application* (Addison–Wesley, Reading, MA, 1981).

19. R. W. Hockney and J. W. Eastwood, *Computer Simulation Using Particles* (McGraw–Hill, New York, 1981).

20. E. L. Pollock and J. Glosli, Comments on P³M, FMM, and the Ewald method for large periodic Coulombic systems, *Comput. Phys. Comm.* **95**, 93 (1996).

21. H. M. P. Couchman, Mesh-refined P³M: A fast adaptive N-body algorithm, *Astrophys. J.* **368**, L23 (1991).

22. H. M. P. Couchman, P. A. Thomas, and F. R. Pearce, HYDRA: An adaptive-mesh implementation of P³M-SPH, *Astrophys. J.* **452**, 797 (1995).

23. N. Y. Gnedin and E. Bertschinger, Building a cosmological hydrodynamic code: Consistency condition, moving mesh gravity, and SLH-P³M, *Astrophys. J.* **470**, 115 (1996).

24. W. D. Elliot and J. A. Board, Jr., Fast Fourier transform accelerated fast multipole algorithm, *SIAM J. Sci. Comput.* **17**, 398 (1996).

25. C. A. White and M. Head-Gordon, Rotating around the quartic angular momentum barrier in fast multipole method calculations, *J. Chem. Phys.* **105**, 5061 (1996).

26. L. Greengard and V. Rokhlin, A new version of the fast multipole method for the Laplace equation in three dimensions, *Acta Numer.* 229 (1997).

27. H. Hasimoto, On the periodic fundamental solutions of the Stokes equations and their application to viscous flow past a cubic array of spheres, *J. Fluid Mech.* **5**, 317 (1959).

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

29. C. Pozrikidis, Computation of periodic Green's functions of Stokes flow, *J. Eng. Math.* **30**, 79 (1996).

30. J. M. Rallison, A numerical study of the deformation and burst of a viscous drop in general shear flows, *J. Fluid Mech.* **109**, 465 (1981).

31. M. Loewenberg and E. J. Hinch, Collision of two deformable drops in shear flow, *J. Fluid Mech.* **338**, 299 (1997).

32. J. Happel and H. Brenner, *Low Reynolds Number Hydrodynamics* (Nijhoff, Dordrecht, 1973).

33. A. F. Nikiforov and V. B. Uvarov, *The Foundations of Special Function Theory* (Nauka, Moscow, 1974). [In Russian]

34. G. A. Korn and T. M. Korn, *Mathematical Handbook for Scientists and Engineers* (McGraw–Hill, New York, 1968).

35. G. Mo and A. S. Sangani, A method for computing Stokes flow interactions among spherical objects, *Phys. Fluids* **6**, 1637 (1994).

36. A. Z. Zinchenko, Calculation of the effectiveness of gravitational coagulation of drops with allowance for internal circulation, *Prikl. Mat. Mech.* **46**, 58 (1982).

37. A. Z. Zinchenko, Hydrodynamical interaction of two identical liquid spheres in a linear flow field, *Prikl. Mat. Mech.* **47**, 37 (1983).

38. D. R. McKenzie, R. C. McPhedran, and G. H. Derrick, The conductivity of lattices of spheres. II. The body-centred and face-centred cubic lattices, *Proc. R. Soc. London A* **362**, 211 (1978).

39. A. S. Sangani, D. Z. Zhang, and A. Prosperetti, The added mass, Basset, and viscous drag coefficients in nondilute bubbly liquids undergoing small-amplitude oscillatory motion, *Phys. Fluids A* **3**, 2955 (1991).

40. S. O. Unverdi and G. Tryggvason, A front-tracking method for viscous, incompressible, multi-fluid flows, *J. Comput. Phys.* **100**, 25 (1992).

41. V. Cristini, J. Blawzdziewicz, and M. Loewenberg, Drop breakup in three-dimensional viscous flows, *Phys. Fluids* **10**, 1781 (1998).

42. S. Kwak and C. Pozrikidis, Adaptive triangulation of evolving, closed, or open surfaces by the advancing-front method, *J. Comput. Phys.* **145**, 61 (1998).

43. M. Manga and H. A. Stone, Buoyancy-driven interactions between two deformable viscous drops, *J. Fluid Mech.* **256**, 647 (1993).

44. P. K. MacKeown, *Stochastic Simulation in Physics* (Springer-Verlag, New York, 1997).

45. A. Z. Zinchenko, Effect of hydrodynamical interactions between the particles on the rheological properties of dilute emulsions, *Prikl. Mat. Mech.* **48**, 198 (1984).

46. R. H. Davis, Sedimentation of axisymmetrical particles in shear flows, *Phys. Fluids A* **3**, 2051 (1991).

47. D. L. Koch and E. S. G. Shaqfeh, The instability of dispersion of sedimenting spheroids, *J. Fluid Mech.* **209**, 521 (1989).

48. S. Kim and S. Karilla, *Microhydrodynamics: Principles and Selected Applications* (Butterworth-Heinemann, Boston, 1991).

49. M. R. Kennedy, C. Pozrikidis, and R. Skalak, Motion and deformation of liquid drops, and the rheology of dilute emulsions in simple shear flow, *Comput. & Fluids* **23**, 251 (1994).