

Cubically Regularized Stokeslets for Fast Particle Simulations of Low-Reynolds-Number Drop Flows

LUDWIG C. NITSCHÉ
AND PRASHANTH PARTHASARATHI

Department of Chemical Engineering, The University of Illinois
at Chicago, Chicago, Illinois, USA

*Swarms of Stokeslets have previously been shown to be effective for simulating three-dimensional, free-surface, buoyancy-driven drop flows at unit viscosity ratio, including interfacial rupture and pass-through phenomena. This article presents an efficient marker-and-cell/fast Fourier transform (MAC-FFT) algorithm that yields $N \log N$ scaling of the operations, thereby enabling accurate simulations involving millions of particles. The formerly separate steps of regularization of the Green's function (commonly used in vortex blob methods for inviscid flow) and discretization are unified by regularizing the Stokeslet over the cubical cells that form the underlying grid—as opposed to the spherical blobs used in the past. A piecewise-constant drop phase function, obtained by simple binning of the particles, thereby yields second-order quadrature of the Green's function to obtain the velocity field. An iterative cascade algorithm (adapted from Daubechies wavelets) allows the Stokes “cubelet” field to be calculated very efficiently. A similar cubelet approach is used for the cohesive forces that mimic interfacial tension. A user-friendly library of Fortran 95 subroutines (**DropLib**, www.droplib.org) has been developed to carry out the simulations and visualize drop shape evolutions in three dimensions.*

Keywords Creeping flow; Discrete convolution; Drop dynamics; Fast convolution; Fast Fourier transform; Green's function; Marker-and-cell; Regularization; Stokeslet

Introduction

This contribution culminates a series of previous articles (Machu et al., 2001; Nitsche and Schafflinger, 2001; Nitsche et al., 2004a,b) that developed the numerical approach of simulating creeping, free-surface flows of liquid drops—thus far at unit viscosity ratio—with swarms of Stokeslets. For extremely large numbers of Stokeslets, we present here a marker-and-cell/fast Fourier transform (MAC-FFT) scheme whereby the particle positions at each time step are mapped onto a regular grid of cubical cells to extract a piecewise constant density profile. By regularizing the Stokeslet over the cubical cell (as opposed to a spherical blob), second-order accuracy is obtained for the grid integration. A cubical distribution is also used for the (non-singular) cohesive kernel that mimics the effect of interfacial tension.

Respectfully dedicated to Professor Howard Brenner on the occasion of his 80th birthday.

Address correspondence to Ludwig C. Nitsche, Department of Chemical Engineering, The University of Illinois at Chicago, 810 S. Clinton St., Chicago, IL 60607, USA. E-mail: lcn@uic.edu

Discrete convolutions for the cell-cell interaction kernels are reduced to $O(N \log N)$ operations using the FFT. Particle velocities are then trilinearly interpolated from the grid to update the positions in a fourth-order Runge-Kutta integration of the dynamical system. Owing to $1/r$ decay of the Stokeslet and fixed radius of cohesion relative to the (small) cell dimension δ , the near-cell hydrodynamic and cohesive interactions are of order δ^2 and δ^3 , respectively. Thus, near-field contributions are sufficiently resolved by the regularized kernels in our pure particle-mesh method, without the need for particle-particle terms from the P³M approach of Hockney and Eastwood (1988).

A main purpose of this article is to present efficient methods for calculating the cubically smeared-out kernels. For the Stokes cubelet, we combine the analytical far-field formula with an iterative recursion scheme that is reminiscent of the cascade algorithm for calculating Daubechies wavelets (Daubechies, 1988; Debnath, 2002). For the cohesive cubelet we obtain an asymptotic formula.

In contrast to the literature on low-Reynolds-number flows, the literature on inviscid flows records a long history of particle-based numerical approaches to the Euler equations. These methods use either singular point vortices (PVM) or regularized vortex blobs (VBM) that get advected by the flow that they themselves generate by superposition (Chorin, 1973; Beale and Majda, 1982; Anderson and Greengard, 1985; Leonard, 1985; Beale, 1986; Hou and Lowengrub, 1993; Hou et al., 1993; Haroldsen and Meiron, 1998; Winckelmans et al., 2005). In this application, the summation of vortices represents a particular discretization for implementing the Biot-Savart law, by which a convolution integral of the Green's function for Poisson's equation against the curl of the vorticity yields the velocity field.

An analogous point dipole method (PDM) was used more recently to simulate two-dimensional non-Newtonian flow in a journal bearing (Phillips, 2003). An integral constitutive equation yielded the (history-dependent) viscoelastic stresses from which the dipole strengths were obtained. These point dipoles were then summed against the relevant propagator (gradient of the Stokeslet) to yield the velocity field by which the dipole positions could be updated.

A multitude of exact and approximate solutions have been obtained for creeping flow in various geometries by distributing the Stokeslet and/or its derivatives over curves and surfaces (Blake, 1971; Dąbroś, 1985; Nitsche and Brenner, 1990; Claugue and Phillips, 1996, 1997). Both slender body theory (Batchelor, 1970; Cox, 1970; Tillett, 1970) and the boundary-integral method (Rallison and Acrivos, 1978; Rallison, 1981; Stone and Leal, 1989; Manga and Stone, 1993; Pozrikidis, 1990, 1992, 2001, 2002; Loewenberg and Hinch, 1996; Cristini et al., 1998, 2001; Davis, 1999; Zinchenko and Davis, 2006; Griggs et al., 2007, 2008) represent versions of the general singularity approach that are particularly amenable to systematic and highly sophisticated discretization schemes. It therefore seems odd that volumetric distributions of freely advected Stokeslets have received much less attention than their counterparts in inviscid flow.

Because the unbounded singular solutions induce unphysically large mutual velocity disturbances at nearby points, they are usually regularized by mathematically smearing out the distribution of force/vorticity/stretching over a small volume—typically in a spherically symmetric fashion using a radial cutoff function. Estimates of quadrature error and considerations of numerical stability speak in favor of choosing the cutoff diameter to exceed the interparticle spacing, so that

these “fuzzy Stokeslets”/“vortex blobs”/“dipole blobs” overlap as they move according to the local velocity field (Hou et al., 1993; Lowengrub et al., 1993; Phillips, 2003; Baker and Beale, 2004; Nitsche et al., 2004a); see also Hockney and Eastwood (1988, p. 270). Our cubically regularized interaction kernels do not advect or overlap: rather, they stack contiguously together to form a fixed grid. Thus, we have unified the formerly separate steps of regularization and discretization.

In vortex-based numerics, grids are used in two different contexts. One (Lagrangian) usage refers simply to the initial placement of the vortex particles in a regular lattice. This lattice enables the spatial quadrature for the Biot-Savart law, but may also be used to calculate the vorticity by a finite-difference approximation (Beale and Majda, 1982; Hou and Lowengrub, 1990).

Two uses of Eulerian grids are standard in inviscid flow numerics for accelerating the particle summations (Winckelmans et al., 2005). The tree codes of fast multipole methods (FMM) use a binary hierarchy of cells, each half as large as those of the preceding level, to organize the particles into nearby versus distant clusters. Local multipole expansions are tuned to the cell sizes and distances of interaction (Chew et al., 2000; Greengard, 1994; Greengard and Rokhlin, 1987; Carrier et al., 1988; Rokhlin, 1985; Sangani and Mo, 1996; Phillips, 2003). For a fixed level of accuracy, the summations can be reduced to $O(N)$ operations. For our purposes fixed error is not entirely realistic as a criterion, because the motivation for adding more particles within the same drop volume is to diminish the discretization error in the convolution integrals. In enforcing the aforementioned $O(N^{-1/2} \log N)$ error scaling, one might expect FMM to come in around $O(N \log N)$ operations.

Eulerian grids are also used in multigrid and other fast Poisson solvers, as an alternative to the Biot-Savart convolution integral of the Green’s function to obtain the velocity field from the grid-mapped vorticity (Hockney and Eastwood, 1988; Liu and Doorly, 2000; Liu, 2001; Winckelmans et al., 2005).

Once the particles are binned into the piecewise-constant density profile, we apply the FFT to carry out the discrete convolution sums for cohesive and hydrodynamic cell-cell interactions in $O(N \log N)$ operations (Press et al., 1992). Spatial periodicity inherent in the Fourier transform does *not* imply periodic boundary conditions in our simulations, because the grid on which we carry out the FFT is doubled in each dimension relative to the original simulation grid. To fill the FFT grid, the particle density and force arrays are padded out with zeros, and the kernel arrays are extended to twice the original range of interaction. Thus, the implied periodic extension of the Stokeslet on the doubled FFT grid is not equivalent to the periodic Stokeslet of Hasimoto (1959): indeed, it preserves the convolution sum for the *free-space* Stokeslet on the original grid.

Particle-mesh Ewald techniques use the FFT to accelerate a *reciprocal-space* convolution sum, which accompanies an additional short-range, physical-space sum of $O(N)$ operations to represent the grid-mapped interaction of particles via a *periodic* kernel of electrostatic (Darden et al., 1993; Essmann et al., 1995) or hydrodynamic (Sierou and Brady, 2001, 2002; Saintillan et al., 2005) origin.

Our MAC-FFT method is appealingly simple yet sufficiently versatile to track exceedingly complicated shape evolutions of drops. Before proceeding with further details, it is appropriate to establish the mathematical formulation.

Boundary Integral Formulation

For unit viscosity ratio, the inertialess flow field set in motion by gravity and interfacial tension for any drop configuration—volume \mathcal{V} bounded by surface(s) \mathcal{S} —can be represented by a suitable spatial distribution of the (dimensionless) Green's function for Stokes flow, or Stokeslet:

$$\mathbf{G}(\mathbf{r}) = \frac{1}{8\pi\|\mathbf{r}\|} \left[\mathbf{I} + \|\mathbf{r}\|^{-2} \mathbf{r}\mathbf{r} \right] \quad (1)$$

The velocity at each point is given by

$$\mathbf{v}(\mathbf{r}) = \iiint_{\mathcal{V}} \mathbf{e}_g \cdot \mathbf{G}(\mathbf{r} - \mathbf{q}) dV[\mathbf{q}] - 2B^{-1} \iint_{\mathcal{S}} H(\mathbf{q}) \mathbf{n}(\mathbf{q}) \cdot \mathbf{G}(\mathbf{r} - \mathbf{q}) dS[\mathbf{q}] \quad (2)$$

with \mathbf{e}_g the unit vector pointing in the direction of gravity, H the local surface mean curvature, and \mathbf{n} the local, outward pointing unit normal vector on the surface. In terms of the undeformed drop radius a , interfacial tension σ , and density difference $\Delta\rho$, the (dimensionless) Bond number B is given by

$$B = \frac{(\Delta\rho)ga^2}{\sigma} \quad (3)$$

This represents the strength of viscous deforming forces relative to the restoring force of interfacial tension.

The boundary integral formulation applies the divergence theorem to transfer the body force from the volume to the surface:

$$\mathbf{v}(\mathbf{r}) = \iint_{\mathcal{S}} [\mathbf{e}_g \cdot \mathbf{q} - 2B^{-1}H(\mathbf{q})] \mathbf{n}(\mathbf{q}) \cdot \mathbf{G}(\mathbf{r} - \mathbf{q}) dS[\mathbf{q}] \quad (4)$$

The advantage of discretizing in two dimensions rather than three is offset by the difficulty of tracking interfaces when these become substantially distended, roll up, pinch off, or coalesce (Cristini et al., 1998, 2001). Equation (4) will be integrated numerically in order to validate the MAC-FFT velocity field for spheroidal drops.

Volumetric Cohesion as a Model of Interfacial Tension

Volumetric approaches can deal with exceedingly complicated drop shapes; these include front tracking (Unverdi and Tryggvason, 1992; Tryggvason et al., 2001; Olgac et al., 2006) level-set methods (Chang et al., 1996; Osher and Fedkiw, 2002; Bassano, 2004; Tanguy and Berlemont, 2005; Deshpande and Zimmerman, 2006; Xu et al., 2006), the volume-of-fluid method (Gueyffier et al., 1999; Scardovelli and Zaleski, 1999; Li et al., 2000; Renardy and Renardy, 2002), diffuse interface methods (Anderson et al., 1998; Yue et al., 2004), and our Stokeslet method (Machu et al., 2001; Nitsche and Schaflinger, 2001; Nitsche et al., 2004a,b). When viewed in volumetric terms, forces due to interfacial tension are effectively concentrated into a surface delta function; see Equation (2). For numerical purposes, this singular spike must be distributed over a thin but finite zone straddling the interface—at some loss of accuracy. Current volumetric approaches to interfacial tension require spatial derivatives of the normal vector, which itself is calculated from the gradient

of the phase function or level-set function (Brackbill et al., 1992; Lafaurie et al., 1994; Coward et al., 1997; Gueyffier et al., 1999; Scardovelli and Zaleski, 1999; Morris, 2000). Interfacial tension is actually equivalent to a cohesive force acting only between those drop material points that lie at (or within a thin layer covering) the surface; see Nitsche et al. (2004a).

Two previous articles (Nitsche and Schaflinger, 2001; Nitsche et al., 2004a) have developed an alternative volumetric approach that entirely avoids the surface normal vector. A radially symmetric cohesive potential ϕ , acting volumetrically between *all* drop material points within some cutoff radius ϵ (i.e., not just between material points on the surface), was shown to lead to the same velocity field as in Equation (4), within $O(\epsilon^2)$ corrections.

$$\mathbf{v}(\mathbf{r}) = \iiint_{\mathcal{V}} [\mathbf{e}_g + 2B^{-1}\nabla\Phi(\mathbf{q}; \epsilon)] \cdot \mathbf{G}(\mathbf{r} - \mathbf{q}) \, dV[\mathbf{q}] \quad (5)$$

Defining $\mathcal{B}(\mathbf{r}; \epsilon)$ to be a ball of radius ϵ centered at point \mathbf{r} , we have

$$\Phi(\mathbf{r}; \epsilon) = \frac{1}{\pi\epsilon^4 I_3} \iiint_{\mathcal{V} \cap \mathcal{B}(\mathbf{r}; \epsilon)} \phi\left(\frac{\|\mathbf{r} - \mathbf{q}\|}{\epsilon}\right) \, dV[\mathbf{q}] \quad (6)$$

Radial moments of the dimensionless cohesive potential ϕ are defined as follows:

$$I_n = \int_0^1 \rho^n \phi(\rho) \, d\rho \quad (7)$$

There is considerable latitude with regard to the functional form $\phi(\rho)$. It may, but need not, mimic an actual intermolecular potential. Aside from continuous differentiability and compact support ($\phi=0$ for $\rho > 1$), the potential must have a net energetic effect: $I_2 \neq 0$.

The cohesive potential does not actually produce tensile forces at the interface, but it does extract an asymptotic approximation of the local mean curvature when it is evaluated exactly on the interface (Rowlinson and Widom, 1989; Nitsche et al., 2004a).

$$H(\mathbf{r}) = \left[\frac{2I_2}{\epsilon I_3} - \Phi(\mathbf{r}, \epsilon) \right] + A(\mathbf{r})H^3\epsilon^2 + \dots \quad \mathbf{r} \in \mathcal{S} \quad (8)$$

The volumetrically forced velocity field (5) results from neglecting the $\text{ord}(\epsilon^2)$ correction term. Formally then, one would therefore require the radius of cohesion ϵ to be asymptotically small compared to the smallest radius of curvature on the interface (which does *not* mean that the drops being modeled are confined to small deformations). In numerical practice the forgiving nature of the asymptotic expansion (8) allows this formal restriction to be relaxed considerably. For example, Nitsche et al. (2004a) bounded the coefficient A for a prolate spheroid of minor/major axis ratio R and a typical functional form for the dimensionless radial potential ϕ :

$$|A(\mathbf{r})| \leq 0.0746(1 - R^2) \quad (9)$$

The influence of ϵ on a simulated evolution of drop shapes will be seen in Figure 4.

The cohesive pressure field corresponding to the velocity field (5) differs from that due to interfacial tension, but this has no consequence for tracking drop shapes. The actual pressure field could be reconstructed on a post facto basis from its gradient, as given by the Laplacian of the numerically calculated velocity according to the Stokes equations.

We represent each drop with a swarm of particles distributed with statistically uniform number density over its interior. The volume attributed to each particle is denoted by δV . The interface is defined at length scales larger than the mean inter-particle spacing $(\delta V)^{1/3}$, as a jagged zone of transition between the particle-laden and particle-free regions. Summing the cohesive and Stokeslet kernels over the particle positions is equivalent to a discretized, Monte Carlo approximation of the convolution integrals (5) and (6).

$$\frac{d\mathbf{r}_n}{dt} = (\delta V) \sum_{m \neq n} \mathbf{f}_m \cdot \mathbf{G}(\mathbf{r}_n - \mathbf{r}_m) \quad (10)$$

$$\mathbf{f}_n = \mathbf{e}_g + \frac{2\delta V}{B\pi\epsilon^5 I_3} \sum_{m=1}^N \phi' \left(\frac{\|\mathbf{r}_n - \mathbf{r}_m\|}{\epsilon} \right) \frac{\mathbf{r}_n - \mathbf{r}_m}{\|\mathbf{r}_n - \mathbf{r}_m\|} \quad (11)$$

In numerical practice, the Stokeslet kernel $\mathbf{G}(\mathbf{r})$ would be replaced by a regularized version—see Equation (30), below—with a cutoff radius at least as large as $(\delta V)^{1/3}$ in order that neighboring Stokeslet blobs overlap their diffuse clouds of force. Because these regularized kernels do not blow up at $r=0$, the self-interaction term would not be omitted from the summation in Equation (10). As written, the dynamical system (10), (11) would require order N^2 operations to calculate the mutual cohesive and viscous interactions between all particles. Such a direct procedure becomes computationally intractable for tens or hundreds of thousands of particles—hence the need for fast summation techniques.

Toward more efficient summations, we first define a phase (or color) function

$$c(\mathbf{r}) = \begin{cases} 1, & \mathbf{r} \in \mathcal{V} \\ 0, & \text{otherwise} \end{cases} \quad (12)$$

that allows both Equations (5) and (6) to be written in convolution form:

$$\mathbf{v}(\mathbf{r}) = \iiint_{\mathbb{R}^3} \mathbf{f}(\mathbf{q}) \cdot \mathbf{G}(\mathbf{r} - \mathbf{q}) dV[\mathbf{q}] \quad (13)$$

$$\mathbf{f}(\mathbf{r}) = \mathbf{e}_g c(\mathbf{r}) + \frac{2}{B\pi\epsilon^5 I_3} \iiint_{\mathbb{R}^3} c(\mathbf{q}) \psi(\mathbf{r} - \mathbf{q}; \epsilon) dV[\mathbf{q}] \quad (14)$$

with

$$\psi(\mathbf{r}; \epsilon) \stackrel{\text{def}}{=} \phi' \left(\frac{\|\mathbf{r}\|}{\epsilon} \right) \frac{\mathbf{r}}{\|\mathbf{r}\|}, \quad \phi'(\rho) = \frac{d\phi}{d\rho} \quad (15)$$

Cubically Distributed Kernels and Second-Order Quadrature

For numerical evaluation of the convolution integral (13), we combine the (usually separate) steps of discretization and regularization into a single operation. By averaging the Stokeslet over a cubical cell $\mathcal{C}(\delta)$ of side length δ in the Eulerian grid:

$$\mathcal{C}(\delta) = \{(x, y, z) : |x|, |y|, |z| < \delta/2\} \quad (16)$$

we obtain second-order quadrature errors upon multiplying by the cubewise-constant force density.

$$\begin{aligned} \iiint_{\mathcal{C}(\delta)} \mathbf{f}(\mathbf{q}) \cdot \mathbf{G}(\mathbf{r} - \mathbf{q}) \, dV[\mathbf{q}] &= \delta^3 \overline{\mathbf{f}(\mathbf{q}) \cdot \mathbf{G}(\mathbf{r} - \mathbf{q})} \\ &= \delta^3 \left\{ \overline{\mathbf{f}} \cdot \overline{\mathbf{G}}(\mathbf{r}; \delta) + \overline{[\mathbf{f} - \overline{\mathbf{f}}]} \cdot \overline{[\mathbf{G} - \overline{\mathbf{G}}(\mathbf{r}; \delta)]} \right\} \\ &= \delta^3 \{ \overline{\mathbf{f}} \cdot \overline{\mathbf{G}}(\mathbf{r}; \delta) + O(\delta^2) \} \end{aligned} \quad (17)$$

where the cubical averages are taken over the argument \mathbf{q} . For example,

$$\overline{\mathbf{G}}(\mathbf{r}; \delta) \stackrel{\text{def}}{=} \delta^{-3} \iiint_{\mathcal{C}(\delta)} \mathbf{G}(\mathbf{r} - \mathbf{q}) \, dV[\mathbf{q}] \quad (18)$$

An analogous argument applies to the cohesion integral (14).

$$\iiint_{\mathcal{C}(\delta)} c(\mathbf{q}) \psi(\mathbf{r} - \mathbf{q}; \epsilon) \, dV[\mathbf{q}] = \delta^3 \{ \overline{c} \overline{\psi}(\mathbf{r}; \epsilon, \delta) + O(\delta^2) \} \quad (19)$$

with

$$\overline{\psi}(\mathbf{r}; \delta, \epsilon) \stackrel{\text{def}}{=} \delta^{-3} \iiint_{\mathcal{C}(\delta)} \psi(\mathbf{r} - \mathbf{q}; \epsilon) \, dV[\mathbf{q}] \quad (20)$$

Defining the cubical average of the cohesive potential

$$\overline{\phi}(\mathbf{r}; \epsilon, \delta) \stackrel{\text{def}}{=} \delta^{-3} \iiint_{\mathcal{C}(\delta)} \phi\left(\frac{\|\mathbf{r} - \mathbf{q}\|}{\epsilon}\right) \, dV[\mathbf{q}] \quad (21)$$

we note the relation

$$\overline{\psi}(\mathbf{r}; \epsilon, \delta) = \epsilon \nabla \overline{\phi}(\mathbf{r}; \epsilon, \delta) \quad (22)$$

which will prove useful for evaluating $\overline{\psi}(\mathbf{r}; \epsilon, \delta)$.

In tracking liquid drops with sharp interfaces, one should not expect better than a second-order quadrature error when $c(\mathbf{r})$ and $\mathbf{f}(\mathbf{r})$ have jump discontinuities. Thus, simple binning of the particles for the MAC lumping is sufficient to extract \overline{c} and $\overline{\mathbf{f}}$. Higher order volume interpolation (Hockney and Eastwood, 1988; Liu and Doorly, 2000; Liu, 2001; Walther and Morgenthal, 2002) would be superfluous.

Based upon the binned cell concentrations $\bar{c}_{i,j,k}$, we evaluate the forces at the centers of all of the cells via the discrete convolution sum corresponding to Equation (14).

$$\mathbf{f}_{i,j,k} = \mathbf{e}_g + \frac{2\delta^3}{B\pi\epsilon^5 I_3} \sum_l \sum_m \sum_n \bar{c}_{l,m,n} \bar{\psi}_{i-l,j-m,k-n}(\epsilon, \delta) \quad (23)$$

with

$$\bar{\psi}_{i,j,k}(\epsilon, \delta) \stackrel{\text{def}}{=} \bar{\psi}(i\delta\mathbf{e}_x + j\delta\mathbf{e}_y + k\delta\mathbf{e}_z; \epsilon, \delta) \quad (24)$$

The forces at the individual particles are interpolated trilinearly between the cell-center values $\mathbf{f}_{i,j,k}$ and subsequently binned to obtain the cell-averaged forces $\bar{\mathbf{f}}_{i,j,k}$. These forces in turn figure in the discrete convolution sum for the velocities.

$$\mathbf{v}_{i,j,k} = \delta^3 \sum_l \sum_m \sum_n \bar{\mathbf{f}}_{l,m,n} \bar{\mathbf{G}}_{i-l,j-m,k-n}(\delta) \quad (25)$$

with

$$\bar{\mathbf{G}}_{i,j,k}(\delta) \stackrel{\text{def}}{=} \bar{\mathbf{G}}(i\delta\mathbf{e}_x + j\delta\mathbf{e}_y + k\delta\mathbf{e}_z; \delta) \quad (26)$$

Since the convolution sums over all cells can readily be accelerated with the FFT (Press et al., 1992), what remains to consider is an efficient approach for calculating discrete tabulations of the cubically regularized Stokeslet kernel $\bar{\mathbf{G}}_{i,j,k}(\delta)$ and of the cubically distributed cohesive kernel $\bar{\psi}_{i,j,k}(\epsilon, \delta)$. Note that the kernel tables need be calculated only once and then stored for repeated use at each time step of a simulation.

The Stokes Cubelet

Suitable scaling of the cubically regularized Stokeslet for $\delta=1$ yields $\bar{\mathbf{G}}(\mathbf{r}; \delta)$ for arbitrary side length δ :

$$\bar{\mathbf{G}}(\mathbf{r}; \delta) \stackrel{\text{def}}{=} \delta^{-1} \bar{\mathbf{G}}(\delta^{-1}\mathbf{r}; 1) \quad (27)$$

For large distances $\|\mathbf{r}\|$ one can approximate $\bar{\mathbf{G}}(\mathbf{r}; 1)$ by substituting a Taylor expansion of the Stokeslet (1) into Equation (18) and integrating term-by-term in the dummy argument \mathbf{q} over the unit cube. All terms of odd order vanish immediately by symmetry. Even terms are isotropic, whereby the successive gradients reduce to iterates of the Laplacian: $\nabla^4 \mathbf{G}$ and higher derivatives vanish due to the Stokes equations. Thus we obtain the far-field formula

$$\bar{\mathbf{G}}_{\text{far}}(\mathbf{r}; 1) = \mathbf{G}(\mathbf{r}) + \frac{1}{24} \nabla^2 \mathbf{G}(\mathbf{r}) \quad (28)$$

For the near field one could use a numerical quadrature, but sufficiently fine discretizations for extremely accurate results would be needlessly slow to compute. Instead, we propose an iteration scheme that is reminiscent of the cascade algorithm for calculating Daubechies wavelets (Daubechies, 1988; Debnath, 2002).

We begin by observing that the unit Stokes cubelet field is exactly equivalent to the sum of eight half-size cubelets.

$$\bar{\mathbf{G}}(\mathbf{r}; 1) = \frac{1}{4} \sum_{i=0}^1 \sum_{j=0}^1 \sum_{k=0}^1 \bar{\mathbf{G}} \left[2\mathbf{r} + \left(\frac{1}{2} - i \right) \mathbf{e}_x + \left(\frac{1}{2} - j \right) \mathbf{e}_y + \left(\frac{1}{2} - k \right) \mathbf{e}_z; 1 \right] \quad (29)$$

To launch the iteration scheme one can use any solution of the inhomogeneous Stokes equations that localizes unit total force entirely within the unit cube $\mathcal{C}(1)$; see Equation (16). In order to distribute the force density as uniformly as possible at the outset, we use a spherically regularized Stokeslet

$$\mathcal{G}(\mathbf{r}) = \frac{1}{4\pi a} C_1(\|\mathbf{r}\|/a) \mathbf{I} + C_2(\|\mathbf{r}\|/a) \mathbf{G}(\mathbf{r}) + \frac{a^2}{10} C_3(\|\mathbf{r}\|/a) \nabla^2 \mathbf{G}(\mathbf{r}) \quad (30)$$

$$C_1(\rho) = \begin{cases} 1 - \rho^2 & 0 \leq \rho < 1 \\ 0 & 1 \leq \rho \end{cases} \quad (31)$$

$$C_2(\rho) = \begin{cases} \rho^3 & 0 \leq \rho < 1 \\ 1 & 1 \leq \rho \end{cases} \quad (32)$$

$$C_3(\rho) = \begin{cases} \rho^5 & 0 \leq \rho < 1 \\ 1 & 1 \leq \rho \end{cases} \quad (33)$$

inscribed in the unit cube (i.e., radius $a=1/2$), as illustrated in Figure 1(b). This formula is equivalent to the Hadamard-Rybczynski solution for unit viscosity ratio (Happel and Brenner, 1983; Kim and Karrila, 1991), and would represent a very crude approximation of the Stokes cubelet field $\bar{\mathbf{G}}(\mathbf{r}; 1)$.

We evaluate the initial field (30) at the discrete half-cell positions

$$\mathbf{r}_{I,J,K} = (I/2)\mathbf{e}_x + (J/2)\mathbf{e}_y + (K/2)\mathbf{e}_z \quad (34)$$

and store the resulting tensorial values of \mathcal{G} for $|I|, |J|, |K| \leq 11$ in a five-dimensional array ($3 \times 3 \times 23 \times 23 \times 23$). A cubelet subroutine returns the (current) stored values of $\bar{\mathbf{G}}$ when $|I|, |J|, |K| \leq 11$, and applies the far-field formula (28) otherwise; see

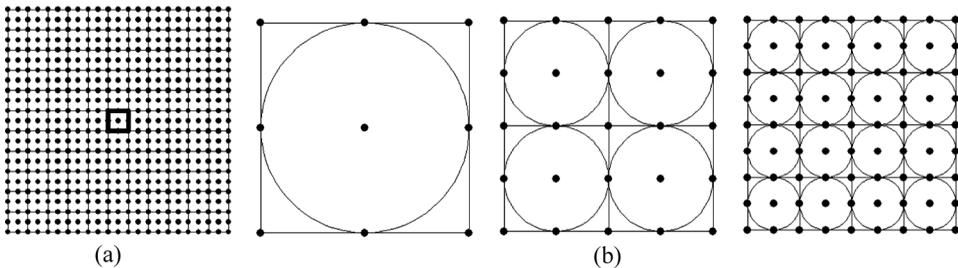


Figure 1. Efficient evaluation of the Stokes cubelet $\bar{\mathbf{G}}_{i,j,k}$. (a) The applicable subroutine retrieves stored values for the unit cubelet (outlined in bold) at the discrete points shown, and applies the far-field formula (28) at more distant points. (b) First three stages of the iteration scheme.

Figure 1(a). This subroutine is now applied to the iteration Equation (29) to update the array for all $|I|, |J|, |K| \leq 11$. Two observations are in order. First, the half-cell array (34)—as opposed to just the cell-center array in Equation (26)—is necessary in order that the half-cell values of the half-size cubelets on the right-hand side of Equation (29) cover the discrete array positions needed for the full-size cubelet on the left-hand side; see Figure 1(b). Second, the points for $7 \leq |I|, |J|, |K| \leq 11$ lie beyond the array for all eight *half-size* cubelets on the right-hand side of Equation (29), which is precisely why the far-field formula is necessary. Figure 1(b) shows how successive updates of the array values are equivalent to increasingly uniform distributions of force density over the unit cube. In other words, n iterations are equivalent to a spatial resolution of 2^{-n} . We obtained 11-digit accuracy for the cube center with 18 iterations, which required about 3 s of computation time on a Dell Latitude D830 notebook PC running the **gfortran** compiler (<http://gcc.gnu.org/fortran/>).

The Cohesive Cubelet

A judicious choice for the cohesive potential (Figure 2)

$$\phi(\rho) = \begin{cases} (1 - \rho^2)^2, & 0 \leq \rho \leq 1 \\ 0, & 1 < \rho \end{cases} \quad (35)$$

allows the cubical average (21) to be integrated easily in closed form for positions \mathbf{r} such that $\|\mathbf{r}\| \leq \epsilon - \delta\sqrt{3}/2$, because then all points \mathbf{q} in the cube $\mathcal{C}(\delta)$ lie safely within the cutoff radius:

$$\bar{\phi}(x, y, z; \epsilon, \delta) = \int_{-\delta/2}^{\delta/2} \int_{-\delta/2}^{\delta/2} \int_{-\delta/2}^{\delta/2} \left[1 - \frac{(x - \xi)^2 + (y - \eta)^2 + (z - \zeta)^2}{\epsilon^2} \right]^2 d\xi d\eta d\zeta \quad (36)$$

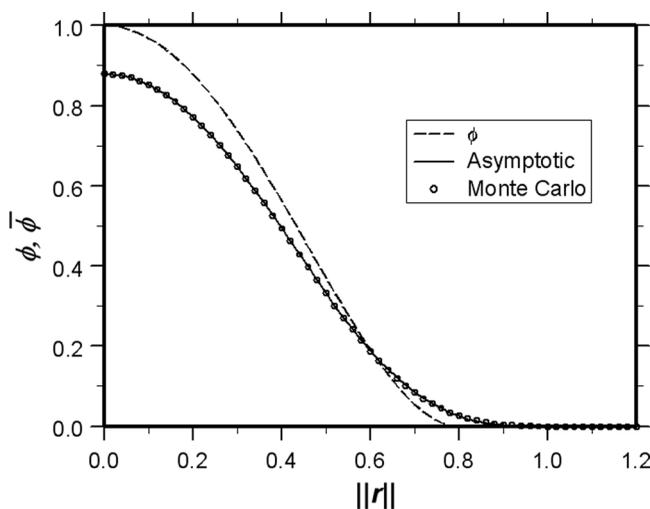


Figure 2. Radial dependence of the cohesive cubelet field $\bar{\phi}(\mathbf{r}; \epsilon, \delta)$ for $\epsilon = 0.8$ and cube side $\delta = 0.4$, evaluated by two alternative methods: (i) asymptotic formulas (39) and (40), and (ii) Monte Carlo evaluation of Equation (21). Also shown is the original potential ϕ , Equation (35), before taking the cubical average.

Similarly, $\bar{\phi}$ must vanish for $\|\mathbf{r}\| \geq \epsilon + \delta\sqrt{3}/2$, because then all points in the cube $\mathcal{C}(\delta)$ lie beyond the cutoff radius. To facilitate integration for the intervening regime, we consider points \mathbf{r} lying along the x axis. For the asymptotic limit of a cube that is small compared to the radius of cohesion ($\delta \ll \epsilon$) we can replace the spherical cutoff surface with a plane

$$\{(\xi, \eta, \zeta) : (x - \xi)^2 + \eta^2 + \zeta^2 = \epsilon^2\} \approx \{(\xi, \eta, \zeta) : \xi = x - \epsilon\} \quad (37)$$

Accordingly the transitional values of x become $\epsilon \pm \delta/2$ instead of $\epsilon \pm \delta\sqrt{3}/2$. The resulting integral

$$\bar{\phi}(x, y, z; \epsilon, \delta) = \int_{x-\epsilon}^{\delta/2} \int_{-\delta/2}^{\delta/2} \int_{-\delta/2}^{\delta/2} \left[1 - \frac{(x - \xi)^2 + \eta^2 + \zeta^2}{\epsilon^2} \right]^2 d\xi d\eta d\zeta \quad (38)$$

is again easily evaluated. Thus we obtain the asymptotic formula

$$\bar{\phi}(\mathbf{r}; \epsilon, \delta) \sim \alpha\left(\frac{\|\mathbf{r}\|}{\epsilon}, \frac{\delta}{\epsilon}\right) \quad (39)$$

with

$$\alpha(\rho, \eta) = \begin{cases} (\rho^2 - 1)^2 + \eta^2 \left(-\frac{1}{2} + \frac{5}{6}\rho^2 + \frac{19}{240}\eta^2 \right) & 0 \leq \rho < 1 - \frac{\eta}{2} \\ \frac{1}{2}(\rho^2 - 1)^2 + \left(\frac{8}{15} - \rho + \frac{2}{3}\rho^3 - \frac{1}{5}\rho^5 \right) \eta^{-1} \\ \quad + \left(-\frac{2}{9} + \frac{5}{6}\rho - \frac{11}{18}\rho^3 \right) \eta & \\ \quad + \left(-\frac{1}{4} + \frac{5}{12}\rho^2 \right) \eta^2 & \\ \quad + \left(\frac{7}{180} - \frac{133}{720}\rho \right) \eta^3 + \frac{57}{1440}\eta^4, & 1 - \frac{\eta}{2} \leq \rho < 1 + \frac{\eta}{2} \\ 0, & 1 + \frac{\eta}{2} \leq \rho \end{cases} \quad (40)$$

For $\epsilon = 0.8$ and $\delta = 0.4$, Figure 2 compares this asymptotic curve with a Monte Carlo integration of Equation (21), with distance $\|\mathbf{r}\|$ being taken along the ray $\theta = \pi/6$, $\phi = \pi/3$ in polar coordinates—i.e., specifically *not* along the x axis, as was used to derive Equation (40). This shows that radial symmetry in Equation (39) is actually a very good approximation.

Differentiating Equation (40) with respect to ρ yields $\bar{\psi}(\mathbf{r}; \epsilon, \delta)$, according to Equation (22).

$$\bar{\psi}(\mathbf{r}; \epsilon, \delta) = \beta\left(\frac{\|\mathbf{r}\|}{\epsilon}, \frac{\delta}{\epsilon}\right) \frac{\mathbf{r}}{\|\mathbf{r}\|} \quad (41)$$

with

$$\beta(\rho, \eta) = \begin{cases} 4\rho(\rho^2 - 1) + \frac{5}{3}\rho\eta^2, & 0 \leq \rho < 1 - \frac{\eta}{2} \\ 2\rho(\rho^2 - 1) - (\rho^2 - 1)^2\eta^{-1} \\ \quad + \frac{1}{6}(5 - 11\rho^2)\eta + \frac{5}{6}\rho\eta^2 - \frac{133}{720}\eta^3, & 1 - \frac{\eta}{2} \leq \rho < 1 + \frac{\eta}{2} \\ 0, & 1 + \frac{\eta}{2} \leq \rho \end{cases} \quad (42)$$

Implementation and Validation of the Numerical Scheme

As is described in the Appendix, a library of Fortran 95 routines (**DropLib**, www.droplib.org) was written to implement the numerical scheme outlined above. For all calculations we used the cell size $\delta=0.08$ with a nominal number density of 64 particles per cell. Thus each drop (of unit undeformed radius) was represented by a statistically uniform swarm of about 523,600 randomly placed particles. The drop configurations were rendered as three-dimensional images by the following procedure. At each time step the particles associated with each drop (regarded as a separate phase) were binned to yield a number density attributed to the center of each cell. In order to damp out local statistical fluctuations, the discrete array of densities was first smoothed by two successive passes of weighted averaging with the 26 immediately neighboring cells. For each drop the phase function $c(\mathbf{r})$ analogous to Equation (12) was obtained by interpolating trilinearly between the corresponding smoothed density array. The interface was then rendered as the $c=0.5$ level surface (including shadows and reflections) with a ray-tracing module (www.quickie3d.org) called by the particle simulation code. All computations were carried out on a Dell Latitude D830 notebook PC running the **gfortran** compiler (<http://gcc.gnu.org/fortran/>). For the two-drop system with cohesion (Figure 5), calculation of all particle velocities from their positions required about 70 s.

For a sufficiently small radius of cohesion ϵ , the cohesive potential $\Phi(\mathbf{r};\epsilon)$ defined in Equation (6) extracts the local surface mean curvature $H(\mathbf{r})$ when the point \mathbf{r} lies exactly on the interface; see Equation (8). In exchange for an asymptotically small error, we have extended throughout three-dimensional space what was originally defined as purely a surface quantity. In order to test the accuracy of Equation (8) in numerical practice, consider a spheroidal drop (semi-axes a and b) and the associated deformation D :

$$\frac{x^2 + y^2}{b^2} + \frac{z^2}{a^2} = 1, \quad D = \frac{a - b}{a + b} \quad (43)$$

Note that D is positive/negative for prolate/oblate spheroids. In order to match the volume of the unit sphere with $D=0.3$, we take $a=1.5109$ and $b=0.8136$. Standard formulas from differential geometry (Gray, 1998) applied to the surface parametrization

$$x = b \sin \theta \cos \phi, \quad y = b \sin \theta \sin \phi, \quad z = a \cos \theta \quad (44)$$

yield the surface mean curvature:

$$H(\theta) = \frac{1}{2} \left[\frac{ab}{[N(\theta)]^{3/2}} + \frac{a/b}{[N(\theta)]^{1/2}} \right], \quad N(\theta) = a^2 \sin^2 \theta + b^2 \cos^2 \theta \quad (45)$$

For various radii of cohesion ϵ , Figure 3(a) employs this exact formula to test the volumetric-cohesive model (6), (8), and (35), as implemented with our MAC-FFT scheme. The progression $\epsilon=0.5, 0.63, 0.8, 1.0$ is chosen to yield roughly a twofold increase in volume (proportional to the number of particles involved in the local cohesive summation) between successive values of ϵ . As ϵ is decreased, the high curvatures at the poles ($\theta=0, \pi$) are resolved more accurately. The smallest sum

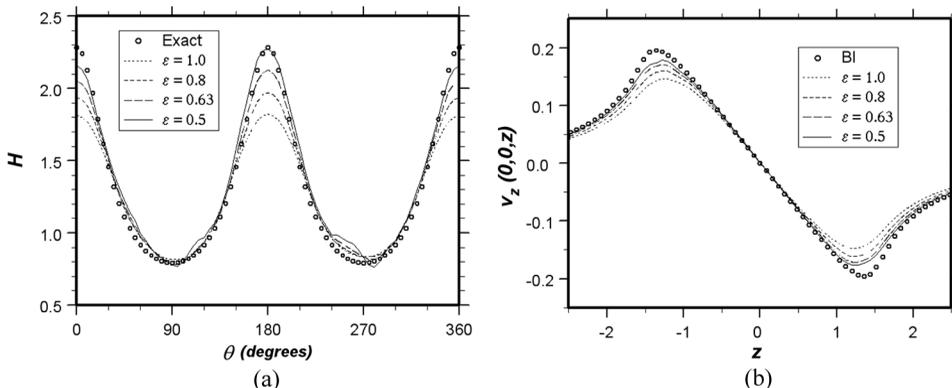


Figure 3. Validation of the MAC-FFT scheme using four different radii of cohesion ($\epsilon = 0.5, 0.63, 0.8, 1.0$) for a prolate, spheroidal drop with deformation $D = 0.3$. (a) Mean curvature as a function of polar position. Equation (8) is compared with the exact result (45). (b) Velocity field. Traversing along the axis of revolution, the numerically calculated axial velocity $v_z(0, 0, z)$ is compared with the boundary-integral formula (46).

($\epsilon = 0.5$) is seen to be the most susceptible to local statistical fluctuations; this effect would be mitigated by refining the particulate and grid discretizations.

The interfacial-tension contribution to the boundary-integral velocity field (4) can be written out explicitly as a double integral over the surface variables θ and ϕ :

$$\begin{aligned} \mathbf{v}(\mathbf{r}) &= \iint_S 2H(\mathbf{q})\mathbf{n}(\mathbf{q}) \cdot \mathbf{G}(\mathbf{r} - \mathbf{q})dS[\mathbf{q}] \\ &= \int_0^\pi \int_0^{2\pi} \frac{b^2 + N(\theta)}{[N(\theta)]^{3/2}} \begin{pmatrix} a \sin \theta \cos \phi \\ a \sin \theta \sin \phi \\ b \cos \theta \end{pmatrix} \bullet \mathbf{G} \begin{pmatrix} x - b \sin \theta \cos \phi \\ y - b \sin \theta \sin \phi \\ z - a \cos \theta \end{pmatrix} a \sin \theta d\phi d\theta \end{aligned} \quad (46)$$

with \mathbf{G} being evaluated according to Equation (1). The integral (46) is then approximated numerically via Simpson's rule, using a step size of $\pi/100$ in both θ and ϕ . For a traverse along the z axis, Figure 3(b) compares the theoretical versus numerical velocity components $v_z(0, 0, z)$ for several values of the cohesive radius ϵ . Again, accuracy increases with decreasing ϵ .

The MAC-FFT simulation of the axisymmetric shape evolution of two trailing drops was tested against a boundary-integral code (generously provided by Professor Robert H. Davis, University of Colorado at Boulder), as illustrated in Figure 4. The overall shape evolution is well captured even when the radius of cohesion is as large as the undeformed drop radius ($\epsilon = 1$). Higher-curvature details are more accurately obtained with smaller values of ϵ . High mean curvature (and correspondingly large tensile forces) form at the lower tip of the trailing drop to balance strong local viscous stretching forces. This fine structure is not well resolved with our radii of cohesion, $\epsilon \geq 0.5$. Thus, the resultant cohesive force is insufficient to hold together the particles at the tip, and they are left behind in an accumulating tail. Particles that trail too far behind do not get counted in the drop-following grid and therefore do not contribute to the velocity field. The cumulative loss of particles does not

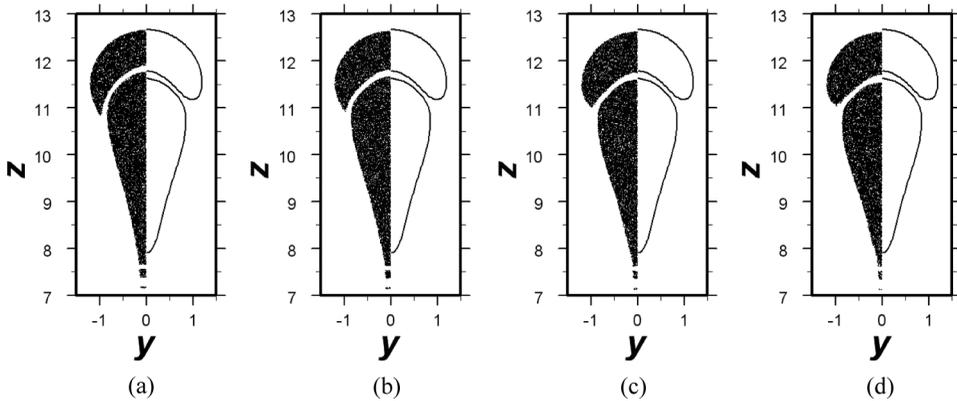


Figure 4. Validation of the volumetric MAC-FFT simulations (sectional view of points on the left) against the boundary-integral method (interfacial outlines on the right; see Davis, 1999) for two buoyant drops at Bond number $B=5$ and unit viscosity ratio. The initially spherical drops are both of unit radius with a gap of 0.1, and the configurations depicted correspond to $t=25$. The overall shape evolution is well captured even when the radius of cohesion is as large as the undeformed drop radius ($\epsilon=1$). Higher-curvature details are more accurately obtained with smaller values of ϵ . The trail of particles is due to (i) extreme curvature at the lower tip of the trailing drop, which is poorly resolved by the cohesive potential, and (ii) neglect of particles that pass outside the drop-following grid. (a) $\epsilon=1.0$, (b) $\epsilon=0.8$, (c) $\epsilon=0.63$, (d) $\epsilon=0.5$.

significantly degrade accuracy of the overall simulation, and could be diminished by reducing the radius of cohesion (which would require a finer mesh to resolve accurately). A direct comparison of our CPU times with the boundary-integral code is not meaningful because the latter (which finished in seconds) is confined to the axisymmetric case and requires the interfacial curves to remain unbroken.

With regard to the cohesive forces, distinct drops are regarded as belonging to different species even though they have the same viscosity. Particles representing a given drop cohere to each other, but *not* to particles in any other drop. This provision is necessary to suppress unphysical coalescence between nearby drops, which would otherwise occur in the simulations.

Distribution of Particles

Our random initial distribution of markers was obtained by choosing a box large enough to contain all drops, then populating the box with a suitable number of Cartesian coordinates from a uniformly distributed random number generator, and finally retaining only those points lying inside the drops. For an *instantaneous* drop configuration thus randomly discretized, the resulting Monte Carlo integration (Press et al., 1992, §7.6) of the Green's function according to Equation (13) yields a better scaling of the quadrature error ($N^{-1/2} \log N$; see Machu et al., 2001) than would a periodic lattice of markers ($N^{-1/3}$) as is typically used in vortex methods (Beale, 1986; Goodman et al., 1990; Cottet et al., 1991; Hou and Lowengrub, 1990). Quadrature errors in the velocity field are addressed by Figure 3(b); see also Adachi et al. (1978) and Ekiel-Jezewska et al. (2006).

Particle redistribution schemes or adaptive creation of new particles is sometimes necessary in order to prevent anisotropic clustering or local depletion of

particles in the course of a simulation, which can degrade accuracy; see, e.g., Liu and Doorly (2000) and Winkelmann et al. (2005) in connection with inviscid flow. A dipole regeneration scheme was also used by Phillips (2003) for non-Newtonian flows. For Stokeslet-based simulations of liquid drops without interfacial tension, a random initial particle distribution *without* redistribution is sufficient for tracking the bell-mushroom-ring transition and the subsequent cascade of bifurcations (Machu et al., 2001). Cohesive forces can engender some spurious clustering effects for coarser swarms (Nitsche and Schaflinger, 2001; Nitsche et al., 2004a), but these have been largely obviated by our MAC-FFT approach—because the enhanced computational efficiency allows much finer spatial discretization and smaller time steps. For the durations of the cohesive simulations presented below (Figures 4 and 5), nonuniformity of the particle distribution was not a significant issue.

We regularize/distribute the interaction kernels over the same fixed, cubical grid cells used to represent the phase function—as opposed to using particle-attached cutoff functions (Chorin, 1973; Anderson and Greengard, 1985; Hou et al., 1993; Phillips, 2003). Because our particles act *only* as markers for the local presence of the drop phase, and because each cell contains a large number of particles, we do not anticipate any significant dynamical effects associated with local statistical fluctuations in the particle number density.

Even in sedimentation of suspension drops (where the particles and their statistical distribution have more tangible physical significance), substructural effects beyond the Stokes hydrodynamics become asymptotically negligible as the particulate discretization is refined (Nitsche and Batchelor, 1997; Schaflinger and Machu, 1999; Machu et al., 2001; Ekiel-Jezewska et al., 2006). For example, in tracking the statistics of a spherical swarm of 4978 sedimenting Stokeslets through a distance of 150 drop radii, Machu et al. (2001) found that the thin trail of exiting particles did not cause a significant statistical redistribution of particles in the drop.

Results and Conclusions

The buoyancy-driven motions and shape evolutions for immiscible ($B=5$) versus miscible ($B \rightarrow \infty$) trailing drops are shown in Figures 5 versus 6, respectively. A cut-away view reveals internal structures, and the reflection from the back wall of the

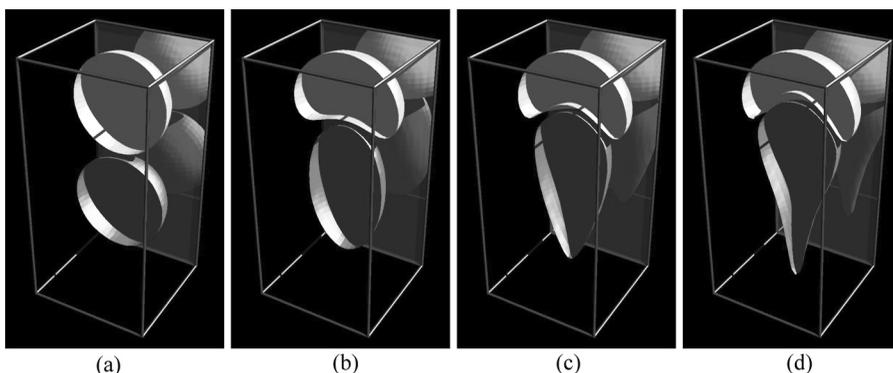


Figure 5. Shape evolution of two buoyant drops for $B=5$. Both initially spherical drops are of unit radius, and the radius of cohesion is $\varepsilon=0.63$. (a) $t=0$, (b) $t=10$, (c) $t=20$, (d) $t=30$.

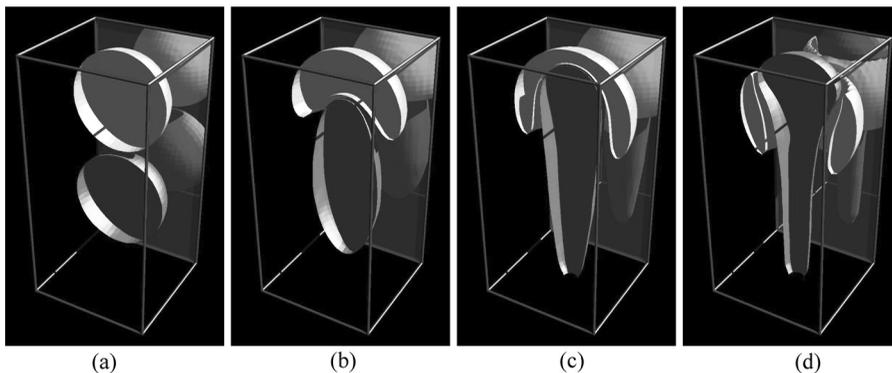


Figure 6. Shape evolution of two buoyant drops for $B \rightarrow \infty$. Both initially spherical drops are of unit radius. (a) $t = 0$, (b) $t = 10$, (c) $t = 20$, (d) $t = 30$.

simulation box shows the external structure. Penetration of the leading drop by the trailing drop has occurred by dimensionless time $t = 30$ in the miscible case; see Machu et al. (2001). Interfacial tension (represented here by cohesion) is seen to reduce the deformation and prevent penetration (Davis, 1999). Finally, a non-axisymmetric shape evolution involving three drops appears in Figure 7.

In our volumetric MAC method, an “interface” is represented by—and visually interpreted as—the abrupt zone of transition between the (ideally, uniform) particulate number density within the drop(s) and zero number density outside. Although the interfacial location and shape are defined only within a distance on the order of the mean interparticle spacing, the “fuzzy” interface itself exists as a meaningful entity independent of the individual identities of the particles that collectively represent the macroscopic shape of a drop at any instant. Because the interfacial zone is automatically renewed with particles from the interior as the free-surface flow proceeds, rupture, penetration, or evolving topology of this thin zone poses no special difficulty for the MAC approach, in contrast to methods that track the interface with markers or other elements of surface discretization (Rallison and Acrivos, 1978; Rallison, 1981; Stone and Leal, 1989; Manga and Stone, 1993; Pozrikidis, 1990, 1992, 2001, 2002; Loewenberg and Hinch, 1996; Cristini et al., 1998, 2001; Davis, 1999; Zinchenko and Davis, 2006; Griggs et al., 2007, 2008). For example, we can track the pass-through mechanism described by Rother et al. (2001); see also Machu et al. (2001).

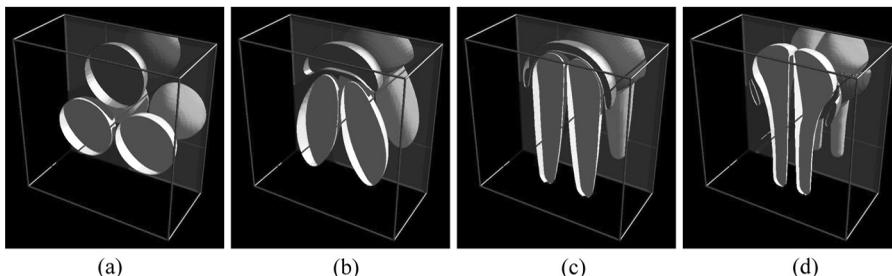


Figure 7. Shape evolution of three buoyant drops for $B \rightarrow \infty$. All three initially spherical drops are of unit radius. (a) $t = 0$, (b) $t = 10$, (c) $t = 20$, (d) $t = 30$.

The MAC-FFT method presented here affords computational efficiency for extremely large numbers of particles. Future work is contemplated in three main directions. First, perturbation and iteration techniques hold promise for generalizing the Stokeslet method to unequal viscosities of the drop phase and surrounding liquid. Second, short-range interspecies potentials will be investigated as a means of modeling coalescence with the retarding effect of film drainage; see Chi and Leal (1989), Yiantsios and Davis (1991), Manga and Stone (1993), and Rother et al. (1997). Finally, the rheology of emulsions (Loewenberg and Hinch, 1996; Zinchenko and Davis, 2002) could be modeled by superposing an ambient shear flow field inside the unit cell of a spatially periodic array of drops and using the particle mesh Ewald analog (Sierou and Brady, 2001, 2002; Saintillan et al., 2005) of our FFT approach to accelerate convolution sums of Hasimoto's (1959) periodic Stokeslet. The effective viscosity of the emulsion could then be extracted from the imposed shear rate and either the resultant viscous dissipation or the total forces on opposing sides of the unit cell.

Acknowledgments

The authors are grateful for the boundary-integral code that was generously provided by Professor Robert H. Davis, University of Colorado, Boulder. This enabled the comparison shown in Figure 4.

References

- Adachi, K., Kiriya, S., and Yoshioka, N. (1978). The behavior of a swarm of particles moving in a viscous fluid, *Chem. Eng. Sci.*, **33**, 115–121.
- Anderson, C. and Greengard, C. (1985). On vortex methods, *SIAM J. Numer. Anal.*, **22**, 413–440.
- Anderson, D. M., McFadden, G. B., and Wheeler, A. A. (1998). Diffuse-interface methods in fluid mechanics, *Annu. Rev. Fluid Mech.*, **30**, 139–165.
- Baker, G. R. and Beale, J. T. (2004). Vortex blob methods applied to interfacial motion, *J. Comput. Phys.*, **196**, 233–258.
- Bassano, E. (2004). Level-set based numerical simulation of a migrating and dissolving liquid drop in a cylindrical cavity, *Int. J. Numer. Methods Fluids*, **44**, 409–429.
- Batchelor, G. K. (1970). Slender-body theory for particles of arbitrary cross-section in Stokes flow, *J. Fluid Mech.*, **44**, 419.
- Beale, J. T. (1986). A convergent 3-d vortex method with grid-free stretching, *Math. Comput.*, **46**, 401–424.
- Beale, J. T. and Majda, A. (1982). Vortex methods. I: Convergence in three dimensions, *Math. Comput.*, **39**, 1–27.
- Blake, J. R. (1971). A note on the image system for a stokeslet in a no-slip boundary, *Math. Proc. Camb. Phil. Soc.*, **70**, 303–310.
- Brackbill, J. U., Kothe, D. B., and Zemach, C. (1992). A continuum method for modeling surface tension, *J. Comput. Phys.*, **100**, 335–354.
- Carrier, J., Greengard, L., and Rokhlin, V. (1988). A fast adaptive multipole algorithm for particle simulations, *SIAM J. Sci. Stat. Comput.*, **9**, 669–686.
- Chang, Y. C., Hou, T. Y., Merriman, B., and Osher, S. (1996). A level set formulation of eulerian interface capturing methods for incompressible fluid flows, *J. Comput. Phys.*, **124**(12), 449–464.
- Chew, W. C., Jin, J.-M., Michielssen, E., and Song, J. eds. (2000). *Fast and Efficient Algorithms in Computational Electromagnetics*, Artech House, Norwood, Mass.

- Chi, B. K. and Leal, L. G. (1989). A theoretical study of the motion of a viscous drop toward a fluid interface at low Reynolds number, *J. Fluid Mech.*, **201**, 123–146.
- Chorin, A. J. (1973). Numerical study of slightly viscous flow, *J. Fluid Mech.*, **57**, 785–796.
- Clague, D. S. and Phillips, R. J. (1996). Hindered diffusion of spherical macromolecules through dilute fibrous media, *Phys. Fluids*, **8**, 1720–1731.
- Clague, D. S. and Phillips, R. J. (1997). A numerical calculation of the hydraulic permeability of three-dimensional disordered fibrous media, *Phys. Fluids*, **9**, 1562–1572.
- Coward, A. V., Renardy, Y. Y., Renardy, M., and Richards, J. R. (1997). Temporal evolution of periodic disturbances in two-layer Couette flow, *J. Comput. Phys.*, **132**, 346–361.
- Cottet, G. H., Goodman, J., and Hou, T. Y. (1991). Convergence of the grid-free point vortex method for the three-dimensional Euler equations, *SIAM J. Numer. Anal.*, **28**, 291–307.
- Cox, R. G. (1970). The motion of long slender bodies in a viscous fluid. Part 1. General theory, *J. Fluid Mech.*, **44**, 791–810.
- Cristini, V., Blawdziewicz, J., and Loewenberg, M. (1998). Drop breakup in three-dimensional viscous flows, *Phys. Fluids*, **10**, 1781–1783.
- Cristini, V., Blawdziewicz, J., and Loewenberg, M. (1998). Near-contact motion of surfactant-covered spherical drops, *J. Fluid Mech.*, **366**, 259–287.
- Dąbros, T. (1985). A singularity method for calculating hydrodynamic forces and particle velocities in low-Reynolds-number flows, *J. Fluid Mech.*, **156**, 1–21.
- Darden, T., York, D., and Pedersen, L. (1993). Particle mesh Ewald: An $N \cdot \log(N)$ method for Ewald sums in large systems, *J. Chem. Phys.*, **98**, 10089–10092.
- Daubechies, I. (1988). Orthonormal bases of compactly supported wavelets, *Commun. Pure Appl. Math.*, **41**, 909–996.
- Davis, R. H. (1999). Buoyancy-driven viscous interaction of a rising drop with a smaller trailing drop, *Phys. Fluids*, **11**, 1016–1028.
- Debnath, L. (2002). *Wavelet Transforms and Their Applications*, Birkhäuser, Boston.
- Deshpande, K. B. and Zimmerman, W. B. (2006). Simulation of interfacial mass transfer by droplet dynamics using the level set method, *Chem. Eng. Sci.*, **61**, 6486–6498.
- Ekiel-Jezewska, M. L., Metzger, B., and Guazzelli, É. (2006). Spherical cloud of point particles falling in a viscous fluid, *Phys. Fluids*, **18**, 038104.
- Essmann, U., Berkowitz, M. L., Perera, L., Darden, T., Lee, H., and Pedersen, L. G. (1995). A smooth particle mesh Ewald method, *J. Chem. Phys.*, **103**, 8577–8593.
- Goodman, J., Hou, T. Y., and Lowengrub, J. (1990). Convergence of the point vortex method for the 2-d Euler equations, *Commun. Pure Appl. Math.*, **43**, 415–430.
- Gray, A. (1998). *Modern Differential Geometry of Curves and Surfaces with Mathematica*, 2nd ed., CRC Press, New York.
- Greengard, L. (1994). Fast algorithms for classical physics, *Science*, **265**, 909–914.
- Greengard, L. and Rokhlin, V. (1987). A fast algorithm for particle simulations, *J. Comput. Phys.*, **73**, 325–348.
- Griggs, A. J., Zinchenko, A. Z., and Davis, R. H. (2007). Low-Reynolds-number motion of a deformable drop between two parallel plane walls, *Int. J. Multiph. Flow*, **33**, 182–206.
- Griggs, A. J., Zinchenko, A. Z., and Davis, R. H. (2008). Gravity-driven motion of a deformable drop or bubble near an inclined plane at low Reynolds number, *Int. J. Multiph. Flow*, **34**, 408–418.
- Gueyffier, D., Li, J., Nadim, A., Scardovelli, R., and Zaleski, S. (1999). Volume-of-fluid interface tracking with smoothed surface stress methods for three-dimensional flows, *J. Comput. Phys.*, **152**, 423–456.
- Happel, J., and Brenner, H. (1983). *Low Reynolds Number Hydrodynamics: With Special Applications to Particulate Media*, Kluwer.
- Haroldsen, D. J. and Meiron, D. I. (1998). Numerical calculation of three-dimensional interfacial potential flows using the point vortex method, *SIAM J. Sci. Comput.*, **20**, 648–683.

- Hasimoto, H. (1959). 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.
- Hockney, R. W. and Eastwood, J. W. (1988). *Computer Simulation Using Particles*, IOP Publishing, Philadelphia.
- Hou, T. Y. and Lowengrub, J. (1993). Convergence of the point vortex method for the 3-d euler equations, *Commun. Pure Appl. Math.*, **43**, 965–981.
- Hou, T. Y., Lowengrub, J., and Shelley, M. J. (1993). The convergence of an exact desingularization for vortex methods, *SIAM J. Sci. Comput.*, **14**, 1–18.
- Kim, S., and Karrila, S. J. (1991). *Microhydrodynamics: Principles and Selected Applications*, Butterworth-Heinemann, Boston.
- Lafaurie, B., Nardone, C., Scardovelli, R., Zaleski, S., and Zanetti, G. (1994). Modelling merging and fragmentation in multiphase flows with SURFER, *J. Comput. Phys.*, **113**, 134–147.
- Leonard, A. (1985). Computing three-dimensional incompressible flows with vortex elements, *Ann. Rev. Fluid Mech.*, **17**, 523–559.
- Li, J., Renardy, Y. Y., and Renardy, M. (2000). Numerical simulation of breakup of a viscous drop in simple shear flow through a volume-of-fluid method, *Phys. Fluids*, **12**, 269–282.
- Liu, C. H. (2001). A three-dimensional vortex particle-in-cell method for vortex motions in the vicinity of a wall, *Int. J. Numer. Methods Fluids*, **37**, 501–523.
- Liu, C. H. and Doorly, D. J. (2000). Vortex particle in cell method for three dimensional viscous unbounded flow computations, *Int. J. Numer. Methods Fluids*, **32**, 29–50.
- Loewenberg, M. and Hinch, E. J. (1996). Numerical simulation of a concentrated emulsion in shear flow, *J. Fluid Mech.*, **321**, 395–419.
- Lowengrub, J. S., Shelley, M. J., and Merriman, B. (1993). High-order and efficient methods for the vorticity formulation of the Euler equations, *SIAM J. Sci. Comput.*, **14**, 1107–1142.
- Machu, G., Meile, W., Nitsche, L. C., and Schafflinger, U. (2001). Coalescence, torus formation and break-up of sedimenting drops: experiments and computer simulations, *J. Fluid Mech.*, **447**, 299–336.
- Manga, M. and Stone, H. A. (1993). Buoyancy-driven interactions between two deformable viscous drops, *J. Fluid Mech.*, **256**, 647–683.
- Morris, J. P. (2000). Simulating surface tension with smoothed particle hydrodynamics, *Int. J. Numer. Methods Fluids*, **33**, 333–353.
- Nitsche, J. M., and Batchelor, G. K. (1997). Break-up of a falling drop containing dispersed particles, *J. Fluid Mech.*, **340**, 161–175.
- Nitsche, L. C. and Brenner, H. (1990). Hydrodynamics of particulate motion in sinusoidal pores via a singularity method, *AIChE J.*, **36**, 1403–1419.
- Nitsche, L. C., Nguyen, A., and Evans, G. (2004a). Globally cohesive drops without interfacial tension, *Chem. Phys. Lett.*, **397**, 417–421.
- Nitsche, L. C., Machu, G., and Meile, W. (2004b). Wavelets and fast summations for particle simulations of gravitational flows of miscible drops, *Computers Chem. Eng.*, **28**, 1873–1879.
- Nitsche, L. C. and Schafflinger, U. (2001). A swarm of stokeslets with interfacial tension, *Phys. Fluids*, **13**, 1549–1553.
- Olgac, U., Kayaalp, A. D., and Muradoglu, M. (2006). Buoyancy-driven motion and breakup of viscous drops in constricted capillaries, *Int. J. Multiph. Flow*, **32**, 1055–1071.
- Osher, S. J. and Fedkiw, R. P. (2002). *Level Set Methods and Dynamic Implicit Surfaces*, Springer, New York
- Phillips, R. J. (2003). A singularity method for calculating time-dependent viscoelastic flows with integral constitutive equations, *J. Fluid Mech.*, **481**, 77–104.
- Pozrikidis, C. (1990). The instability of a moving viscous drop, *J. Fluid Mech.*, **210**, 1–21.
- Pozrikidis, C. (1992). *Boundary Integral and Singularity Methods for Linearized Viscous Flow*, Cambridge University Press, New York.
- Pozrikidis, C. (2001). Interfacial dynamics for stokes flow, *J. Comput. Phys.*, **169**, 250–301.

- Pozrikidis, C. (2002). *A Practical Guide to Boundary-Element Methods with the Software Library BEMLIB*, Chapman & Hall/CRC Press, Boca Raton, Fla.
- Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P. (1992). *Numerical Recipes in FORTRAN 77. The Art of Scientific Computing*, 2nd ed. Cambridge University Press, New York.
- Rallison, J. M. (1981). A numerical study of the deformation and burst of a viscous drop in general shear flow, *J. Fluid Mech.*, **109**, 465–482.
- Rallison, J. M. and Acrivos, A. (1978). A numerical study of the deformation and burst of a viscous drop in an extensional flow, *J. Fluid Mech.*, **89**, 191–200.
- Renardy, Y. and Renardy, M. (2002). PROST: A parabolic reconstruction of surface tension for the volume-of-fluid method, *J. Comput. Phys.*, **183**, 400–421.
- Rokhlin, V. (1983). Rapid solution of integral equations of classical potential theory, *J. Comput. Phys.*, **60**, 187–207.
- Rother, M. A., Zinchenko, A. Z., and Davis, R. H. (1997). Buoyancy-driven coalescence of slightly deformable drops, *J. Fluid Mech.*, **346**, 117–148.
- Rother, M. A., Kushner IV, J. and Davis, R. H. (2001). Buoyancy-driven interactions of viscous drops with deforming interfaces, *J. Fluid Mech.*, **446**, 253–269.
- Rowlinson, J. S., and Widom, B. (1989). *Molecular Theory of Capillarity*, Oxford University Press, Oxford.
- Saintillan, D., Darve, E., and Shaqfeh, E. S. G. (2005). A smooth particle-mesh Ewald algorithm for Stokes suspension simulations: The sedimentation of fibers, *Phys. Fluids*, **17**, 033301.
- Sangani, A. S. and Mo, G. B. (1996). An $o(N)$ algorithm for Stokes and Laplace interactions of particles, *Phys. Fluids*, **8**, 1990–2010.
- Scardovelli, R. and Zaleski, S. (1999). Direct numerical simulation of free-surface and interfacial flow, *Ann. Rev. Fluid Mech.*, **31**, 567–603.
- Schaflinger, U. and Machu, G. (1999). Interfacial phenomena in suspensions, *Chem. Eng. Technol.*, **22**, 617–619.
- Sierou, A. and Brady, J. F. (2001). Accelerated stokesian dynamics simulations, *J. Fluid Mech.*, **448**, 115–146.
- Sierou, A. and Brady, J. F. (2002). Rheology and microstructure in concentrated noncolloidal suspensions, *J. Rheol.*, **46**, 1031–1056.
- Stone, H. A. and Leal, L. G. (1989). Relaxation and breakup of an initially extended drop in an otherwise quiescent fluid, *J. Fluid Mech.*, **198**, 399–427.
- Tanguy, S. and Berlemont, A. (2005). Application of a level set method for simulation of droplet collisions, *Int. J. Multiphase Flow*, **31**, 1015–1035.
- Tillett, J. P. K. (1970). Stokes flow past slender axisymmetric bodies, *J. Fluid Mech.*, **44**, 401.
- Tryggvason, G., Bunner, B., Esmaeeli, A., Juric, D., Al-Rawahi, N., Tauber, W., Han, J., Nas, S., and Jan, Y. J. (2001). A front-tracking method for the computations of multiphase flow, *J. Comput. Phys.*, **169**, 708–759.
- Unverdi, S. O. and Tryggvason, G. (1992). A front-tracking method for viscous, incompressible, multi-fluid flows, *J. Comput. Phys.*, **100**, 25–37.
- Walther, J. H. and Morgenthal, C. (2002). An immersed interface method for the vortex-cell algorithm, *J. Turbulence*, **3**, 039.
- Winckelmans, G., Cocle, R., Dufresne, L., and Capart, R. (2005). Vortex methods and their application to trailing wake vortex simulations, *C. R. Phys.*, **6**, 467–486.
- Xu, J., Li, Z., Lowengrub, J., and Zhao, H. (2006). A level-set method for interfacial flows with surfactant, *J. Comput. Phys.*, **212**, 590–616.
- Yiantsios, S. G. and Davis, R. H. (1991). Close approach and deformation of two viscous drops due to gravity and van der Waals forces, *J. Colloid Interface Sci.*, **144**, 412–433.
- Yue, P., Feng, J. J., Liu, C., and Shen, J. (2004). A diffuse-interface method for simulating two-phase flows of complex fluids, *J. Fluid Mech.*, **515**, 293–317.

- Zinchenko, A. Z. and Davis, R. H. (2002). Shear flow of highly concentrated emulsions of deformable drops by numerical simulations, *J. Fluid Mech.*, **455**, 21–61.
- Zinchenko, A. Z. and Davis, R. H. (2006). A boundary-integral study of a drop squeezing through interparticle constrictions, *J. Fluid Mech.*, **564**, 227–266.

Appendix: DropLib Simulation Package

A user-friendly Fortran 95 subroutine library called **DropLib** was developed to carry out the simulations that are presented in this article. For several Fortran compilers (Silverfrost FTN95, Portland Group PGI Fortran, gfortran) the website www.droplib.org provides precompiled link libraries that may be freely downloaded for use with various computer platforms: Windows/Vista (.dll files), Mac OS X (.dylib files), and Linux (.so files). Linked to the **DropLib** package, the following short program automatically generates as bitmap images (.bmp files) the three-dimensional renderings appearing in Figure 5, above.

```

program drop_run
integer :: i
double precision :: radius1, deform1, buoyancy1, cohesion1
double precision :: radius2, deform2, buoyancy2, cohesion2
double precision, dimension(3) :: center1, center2
call unlock ('2008:af#8-g4fd-&khj:f4s2-k2hj-t4k2:k*(f-hjk&-
j2#3')
buoyancy1=1.0d0 ; cohesion1=0.2d0 ; deform1=0.0d0 ;
radius1=1.0d0
buoyancy2=1.0d0 ; cohesion2=0.2d0 ; deform2=0.0d0 ;
radius2=1.0d0
center1=(/ 0.0d0, 0.0d0, 0.0d0 /)
center2=(/ 0.0d0, 0.0d0, 2.1d0 /)
call drop3d_param (0.63d0,0.08d0,64,200,0.05d0,'top')
call drop3d_spher (3,deform1,radius1,center1,buoyancy1,
cohesion1,'red')
call drop3d_spher (3,deform2,radius2,center2,buoyancy2,
cohesion2,'green')
call drop3d_start
call drop3d_image (600)
do i=1, 3
call drop3d_tstep (2.5d0,10.0d0)
call drop3d_image (600)
end do
end program drop_run

```