Available online at www.sciencedirect.com
SCIENCE DIRECT®
Computers and Chemical Engineering 28 (2004) 1873–1879
Computers & Chemical Engineering
www.elsevier.com/locate/compchemeng

Wavelets and fast summations for particle simulations of gravitational flows of miscible drops

Ludwig C. Nitsche [a,*], Gunther Machu [b], Walter Meile [c]

[a] *Department of Chemical Engineering, University of Illinois at Chicago, 810 South Clinton Street, Chicago, IL 60607, USA*
[b] *Hoerbiger Kompressortechnik Services, Braunhubergasse 23, A-1110 Vienna, Austria*
[c] *Institute of Fluid Mechanics and Heat Transfer, Technical University Graz, Inffeldgasse 25F, A-8010 Graz, Austria*

Received 10 June 2003; received in revised form 2 March 2004; accepted 3 March 2004

Available online 26 April 2004

Abstract

For sedimentation of miscible drops through quiescent liquid of the same viscosity, a recent paper [J. Fluid. Mech. 447 (2001) 299] has shown the effectiveness of computer simulations based upon a swarm of point forces in tracking coalescence, mixing and rupture. Robustness of the approach was offset by the slow $O(N_p^2)$ summations needed to calculate the mutual viscous interactions between all N_p particles. Motivated by applications of wavelets to linear operators, this paper develops a conceptually simple scheme for dramatically accelerating the simulations. After lumping the particles together into N local clusters (three-dimensional "pixels"), a Haar discrete wavelet transform (DWT) is used to "compress" the "bitmapped" six-dimensional image of pixel–pixel hydrodynamic interactions. Depending upon the criterion of accuracy, the numerically observed scaling of the operation count seems to be either $O(N)$ or $O[N(\log N)^\alpha]$. The DWT also works without modification for hydrodynamic wall effects, where the kernel is not purely translational and therefore fast convolutions (FFT) cannot be used.

PACS: 47.11.+j; 47.15.Gf; 47.20.Bp; 47.55.Dz; 47.15.Pn

Keywords: Wavelets; Fast summations; Particle methods; Miscible drops; Creeping flow; Buoyancy-driven instability

1. Introduction: a swarm of point forces

This article describes the use of wavelets to substantially accelerate particle simulations of low-Reynolds-number drop flows. As a physical point of reference, within the realm of common experience and having also geophysical analogs in mantle plumes (Manga, 1997; Manga, Stone, & O'Connell, 1993) consider the rising (or sedimentation) of a miscible drop of ink (or suspension of glass beads) through a quiesent bath of solvent (Fig. 1). The resulting nonlinear shape instability has been described and studied extensively (Adachi, Kiriyama, & Koshioka, 1978; Arecchi, Buah-Bassuah, Francini, Pérez-Garcia, & Quercioli, 1989; Joseph & Renardy, 1993; Kojima, Hinch, & Acrivos, 1984; Machu, Meile, Nitsche, & Schaflinger, 2001; Schaflinger & Machu, 1999; Thomson & Newall, 1885) and includes (i) entrainment of solvent and formation of a tail; (ii) intermediate mushroom and ring shapes; and (iii) an eventual cascade of successive breakups. All of these phenomena are observed in the theoretically simplest case of inertialess flow with equal viscosity of the ink and solvent.

The work of Machu et al. (2001) formalized an analogy between a miscible drop of homogeneuos liquid and a swarm of point forces (cf. Adachi et al., 1978; Kojima et al., 1984; Nitsche & Batchelor, 1997), and solved numerically a (nonlinear) dynamical system for the Stokeslet positions:

$$\frac{\mathrm{d}\boldsymbol{r}_i}{\mathrm{d}t} = \sum_j \boldsymbol{G}(\boldsymbol{r}_i - \boldsymbol{r}_j)\,\delta V \tag{1}$$

with $\boldsymbol{G}$ the (dimensionless) disturbance field produced by a unit point force acting in the direction of gravity,

$$\boldsymbol{G}(\boldsymbol{r}) = (||\boldsymbol{r}||^{-1}\boldsymbol{I} + ||\boldsymbol{r}||^{-3}\boldsymbol{r}\boldsymbol{r}) \cdot \boldsymbol{e}_g \tag{2}$$

and δV the equivalent volume of ink associated with one particle. The resultant error in the local liquid velocity scales with the number N_p of particles like $N_p^{-1/2}\log N_p$.

* Corresponding author. Tel.: +1-312-996-3469; fax: +1-312-996-0808.
E-mail addresses: lcn@uic.edu (L.C. Nitsche), ma@hkts.hoerbiger.com (G. Machu), meile@fluidmech.tu-graz.ac.at (W. Meile).

0098-1354/$ – see front matter © 2004 Elsevier Ltd. All rights reserved.
doi:10.1016/j.compchemeng.2004.03.001

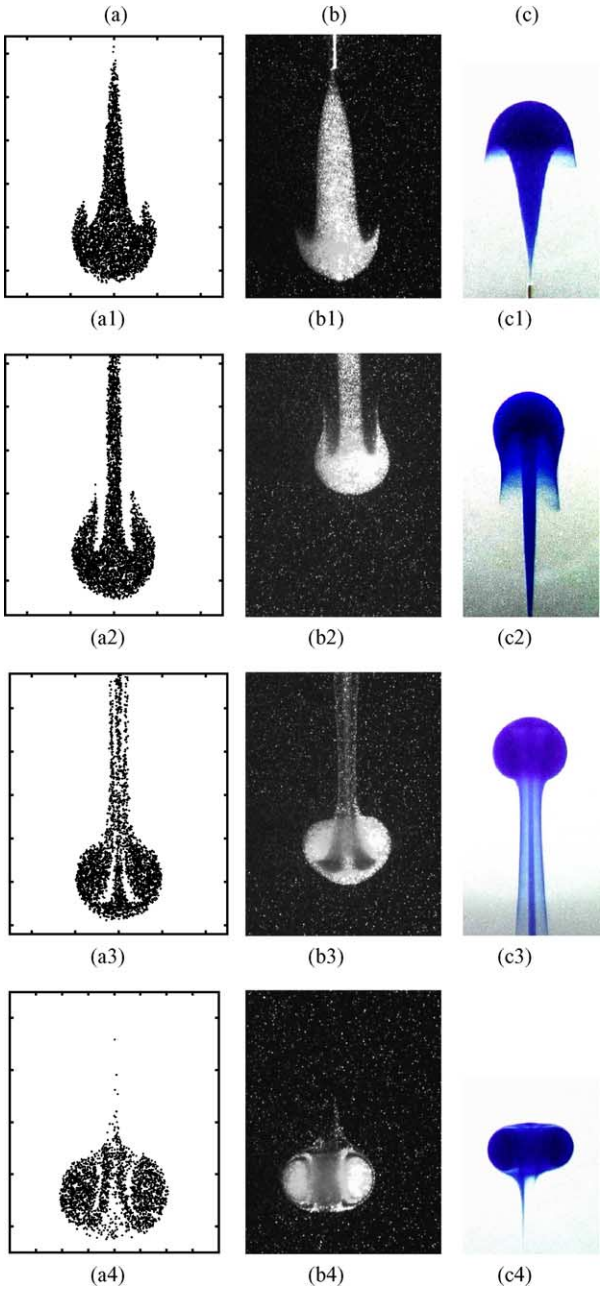


Fig. 1. Four corresponding snapshots of the configurational evolution of a sedimenting or rising miscible drop for three different systems, from the Ph.D. dissertation of Machu (2001). Dimensionless times: $t_1 = 0$, $t_2 = 4$, $t_3 = 13$, $t_4 = 39$. Systems: (a) computer simulation (not accelerated with wavelets) of a swarm of 15,773 sedimenting point forces; (b) sedimenting suspension drop (glass beads of 50 μm diameter, glycerin) illuminated in a sectional plane with a laser light sheet; (c) buoyant miscible drop (commercial blue ink, glycerine, water).

The boundary integral method (Cristini, Blawzdziewicz, & Loewenberg, 1998, 2001; Davis, 1999; Koh & Leal, 1989; Pozrikidis, 1990, 1992; Rallison & Acrivos, 1978; Zinchenko, Rother, & Davis, 1999) is very widely used in low-Reynolds-number drop dynamics because of the efficient premise of two-dimensional discretization. But when interfaces wind around, rupture or coalesce, even sophisticated adaptive surface gridding algorithms (Cristini et al., 1998, 2001) can get bogged down. For simulating the bell-mushroom-ring-breakup sequence described above, a volumetric method such as Machu et al. (2001) seems to be necessary. The purpose of this article is to develop a conceptually and practically simple scheme for speeding up the $O(N_p^2)$ summations of particle interactions. Within the equal-viscosity assumption, particle swarms then emerge as an attractive approach to low-Reynolds-number drop dynamics.

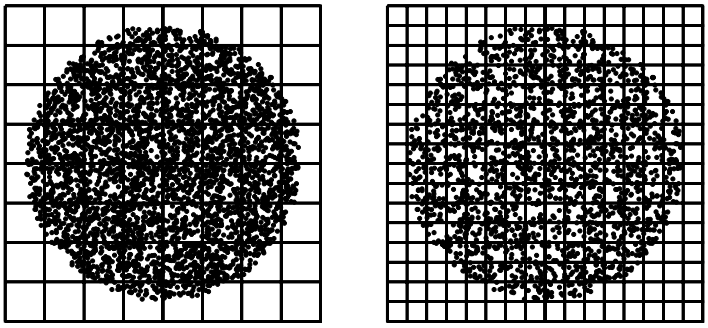
2. Lumping particles together

A common starting point for accelerating particle simulations in a wide variety of physical contexts is to lump nearby particles together within cells or pixels (Hockney & Eastwood, 1988); see Fig. 2. At the crudest, force-monopole level of approximation, one counts the total number ϕ_{lmn} of gravitational point forces in each pixel (l, m, n), and then replaces the superposition of all of their locally distributed hydrodynamic disturbance fields with a single, central point force based upon the summed weight. From a six-dimensional table $\boldsymbol{F}_{ijk,lmn}$ of all pixel–pixel interactions $(i, j, k) \Leftrightarrow (l, m, n)$, the flow velocity $\boldsymbol{v}_{ijk}$ at the center of each pixel (i, j, k) is obtained from the sumation

$$\boldsymbol{v}_{ijk} = \sum_{lmn} \boldsymbol{F}_{ijk,lmn} \phi_{lmn} \, \delta V \tag{3}$$

where the self-interaction $\boldsymbol{F}_{ijk,ijk}$ is taken as the Stokes settling velocity of an equivalent sphere (radius roughly optimized to 0.295 times the pixel dimension in numerical trials), as an approximation of the corresponding weakly singular integral. [See comments on “desingularization” in Phillips (2003, pp. 84–85).] Individual particle velocities in the swarm can then be interpolated between values in the pixel table $\boldsymbol{v}_{ijk}$. Eq. (3) could be renumbered (somewhat inconveniently) to look like a matrix multiplied by a vector.

Refinement of this scheme, by evaluating near particle interactions from neighboring pixels directly (Greengard & Rokhlin, 1987; Hockney & Eastwood, 1988) is not necessary because of the fundamentally global nature of gravitational Stokes flow. Integrated over volume, the $1/r$ decay of the Stokeslet means that local structure of linear dimension ε contributes only at order ε^2 to the local flow field that moves and deforms it: the main contribution is actually from everywhere else in the drop. Thus, local pixel refinement is actually superfluous, and coarse pixels can resolve comparatively fine structures. In particular, accurate resolution of the fuzzy, miscible interface is not necessary. For the drop shown in Fig. 2, the intrinsic quadrature errors for the $8 \times 8 \times 8$ and $16 \times 16 \times 16$ pixel grids are surprisingly low: sampled at the centers of all pixels, rough bounds on the maximum deviation in velocity are 0.59 and 0.24%, respectively, scaled to the maximum velocity. Fig. 3 shows that, until this intrinsic error is reached (horizontal asymptote),



(a) (b)

Fig. 2. Lumping of a spherical swarm of 18,850 particles into pixels, illustrated in a sectional view (1 pixel deep) for two levels of discretization. (a) $8 \times 8 \times 8$ pixel grid; (b) $16 \times 16 \times 16$ pixel grid.

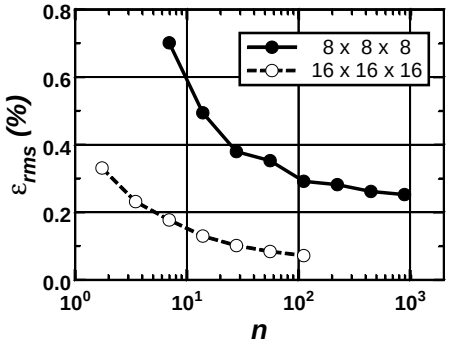


Fig. 3. Difference $\varepsilon_{\mathrm{rms}}$ between the particle sum (1) and the pixel sum (3) as a function of number $n = N_{\mathrm{p}}/N$ of particles per pixel, at two fixed pixel discretizations. The two velocities are computed at the centers of all pixels, and $\varepsilon_{\mathrm{rms}}$ is the root-mean-square discrepancy, normalized as a percentage of the maximum velocity norm.

lumping more particles into each pixel will actually diminish the discrepancy between the particle and pixel sums (1) and (3).

At fixed pixel resolution, the lumped-particle approximation can be improved in a standard way with higher multipoles weighted by the corresponding spatial moments of the local particulate distribution. What distinguishes fast multipole methods (FMM) is their tree-code organization of the pixels into a hierarchy of larger boxes, along with systematic tuning of the associated local multipole expansions based upon box sizes and distances of interaction (Chew, Jin, & Michielssen, 2000; Greengard, 1994; Greengard & Rokhlin, 1987; Phillips, 2003; Rokhlin, 1983; Sangani & Mo, 1996) Adaptive algorithms deal more efficiently with local fine structure (Carrier, Greengard, & Rokhlin, 1988). The resulting truncation of summations reduces the operation count to an $O(N)$ scaling.

3. Wavelets

Leaving aside higher multipoles (which can be incorporated in a straightforward if tedious fashion) the main premise of this article is to exploit a discrete wavelet transform (DWT), which (i) applies, via simple matrix operations, directly to the fine pixel grid; and (ii) automatically compresses the summations without any explicit tree-code data structuring. This approach is suggested generally by applications of wavelets (and multiresolution algorithms) to linear operators (Beylkin, Coifman, & Rokhlin, 1991; Harten & Yad-Shalom, 1994) and more specifically, by the acceleration of matrix multiplications with the DWT (Press, Teukolsky, Vetterling, & Flannery, 1992).

The DWT is essentially a hierarchy of "filtering" operations applied to successively coarser samplings of a pixel image ϕ_{ijk} or $\boldsymbol{F}_{ijk,lmn}$. In each index the filter operates by forming linear combinations of nearby elements in the ordered list given to it, which serve to extract equal amounts of "smooth" versus "detail" information. At each stage the filter is applied only to the smooth half of the output from the previous stage, in what is called a "pyramidal algorithm" (Press et al., 1992). There are two main points. First, all levels in the DWT pyramid can be encapsulated in the action of a single matrix W, which can be stored in packed form to exploit its sparseness (predominance of zero elements); see Fig. 4. Second, the matrix W has the property that most elements of the resultant image in the wavelet basis,

$$\hat{\phi}_{IJK} = \sum_{ijk} W_{Ii} W_{Jj} W_{Kk} \phi_{ijk} \tag{4}$$

are so small that they can be discarded without seriously degrading the image upon reconstituting it with the inverse DWT. Wavelets are particularly good for compressing localized detail, where spectral decompositions become uniformly fuzzy at the same level of truncation (Press et al., 1992). Our illustrative calculations are based on Haar wavelets (Kaiser, 1994; Walker, 1999) which are the simplest.

In some sense, the coarser averages at higher levels of the DWT pyramid are analogs of larger clusters in FMM. In other words, the pyramidal algorithm represents a kind of *automatic, generic tree-code*, independent of the specific operator (physics) involved. Reduction of the whole pyramid to a single matrix operation makes the DWT transparent in its

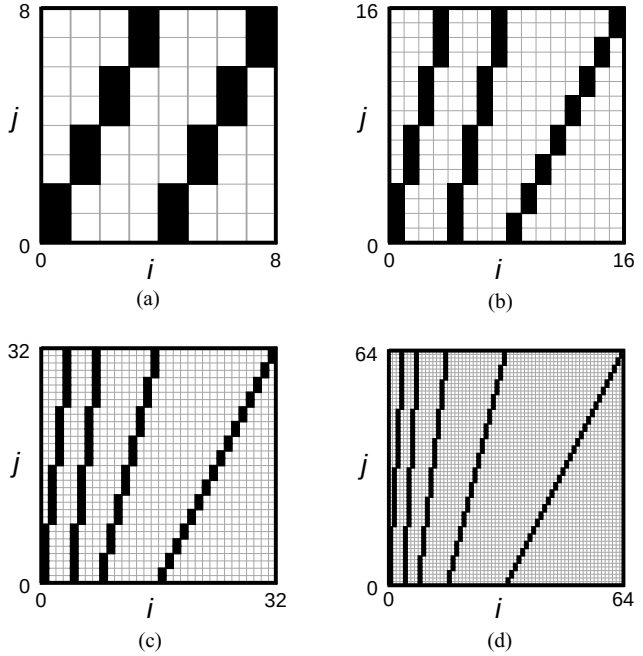


Fig. 4. Structure of the Haar DWT matrix W appearing in Eqs. (4)–(6) and (8), for different linear resolutions. The darkened boxes represent nonzero entries in the array. (a) 8 pixels; (b) 16 pixels; (c) 32 pixels; (d) 64 pixels.

action directly on the fine pixel grid—hence its conceptual appeal.

One can map the six-dimensional array of pixel–pixel interactions $(i, j, k) \Leftrightarrow (l, m, n)$ into wavelet space, and then truncate it into a highly sparse array for faster summations:

$$\hat{\boldsymbol{F}}_{IJK,LMN} = \sum_{ijk,lmn} W_{Ii} W_{Jj} W_{Kk} W_{Ll} W_{Mm} W_{Nn} \boldsymbol{F}_{ijk,lmn} \tag{5}$$

In the absence of wall effects [translational kernel $\boldsymbol{G}(\boldsymbol{r}-\boldsymbol{q})$], the six-dimensional sum can be reduced to triple sum,

$$\hat{\boldsymbol{G}}_{IJK,LMN} = \sum_{ijk,lmn} W_{Ii} W_{Jj} W_{Kk} W_{Ll} W_{Mm} W_{Nn} \boldsymbol{G}_{i-l,j-m,k-n} \tag{6}$$

$$= \sum_{\alpha,\beta,\gamma} A_{IL\alpha} A_{JM\beta} A_{KN\gamma} \boldsymbol{G}_{\alpha\beta\gamma} \tag{7}$$

written in terms of an intermediate three-dimensional array A that is constructed from the wavelet matrix W:

$$A_{IL\alpha} = \sum_{i} W_{Ii} W_{L,i-\alpha} \tag{8}$$

Depending upon both the kernel and the wavelet basis, one may or may not be able to exploit a theoretically justified pattern of truncation in the wavelet-space array $\hat{\boldsymbol{F}}_{IJK,LMN}$ to discard most of its elements a priori, without having to first calculate them and then check their magnitude (Beylkin et al., 1991). This aspect will be investigated in a future paper. The important point is that the truncated table of wavelet coefficients $\tilde{\boldsymbol{F}}_{IJK,LMN}$ needs to be calculated only once for all in advance, and applies equally for *any* instantaneous configuration of any swarm of particles. The slow pixel-by-pixel summation (3) is replaced by the much faster wavelet form

$$\boldsymbol{v}_{ijk} = [\text{inverse DWT}] \left\{ \sum_{LMN} (\tilde{\boldsymbol{F}}_{IJK,LMN})(\hat{\phi}_{LMN}) \right\} \delta V \tag{9}$$

4. Scaling of the operation count

Our drop simulations were limited to the $16 \times 16 \times 16$ pixel array illustrated in Fig. 2b. As will be seen below, this resolution is not fine enough for the asymptotic scaling of the operation count to emerge clearly. Thus, we shall extrapolate the scaling with reference to one- and two-dimensional analogs.

Focusing on the $1/r$ radial decay of the Stokeslet, which is here the determining feature, one recognizes a dimensional progression of analogous kernels, of which the 1D version is of the Calderon–Zygmund type (Beylkin et al., 1991):

$$K^{(1)}(x; \xi) = \frac{1}{|x - \xi|} \tag{10}$$

$$K^{(2)}(x, y; \xi, \eta) = \frac{1}{\sqrt{(x-\xi)^2 + (y-\eta)^2}} \tag{11}$$

$$K^{(3)}(x, y, z; \xi, \eta, \zeta) = \frac{1}{\sqrt{(x-\xi)^2 + (y-\eta)^2 + (z-\zeta)^2}} \tag{12}$$

The two and three-dimensional cases are similar; the major qualitative difference is between one dimension and two, where a singular line ($\xi = x$) gets replaced by a singular point [$(\xi, \eta) = (x, y)$]. However the pixels may be renumbered into linear order, this feature destroys the simple diagonal dominance of the corresponding matrix discretization of $K^{(2)}$, which is now an array of diagonally dominant sub-blocks.

As an example of the scaling of the operation count for wavelet compression applied to the kernels $K^{(1)}(x; \xi)$ and $K^{(2)}(x, y; \xi, \eta)$, respectively, analogous mound-shaped signal functions in one and two dimensions,

$$\phi^{(1)}(\xi) = \begin{cases} 1 - 3\xi^2 + 2|\xi|^3, & |\xi| \le 1 \\ 0, & |\xi| > 1 \end{cases},$$

$$\phi^{(2)}(\xi, \eta) = \phi^{(1)}\left(\sqrt{\xi^2 + \eta^2}\right) \tag{13}$$

are "run through" these operators to produce the corresponding output signals $\psi^{(1)}(x)$ and $\psi^{(2)}(x, y)$,

$$\psi^{(1)}(x) = \int_{-1}^{1} K^{(1)}(x, \xi) \phi^{(1)}(\xi)\, \mathrm{d}\xi \tag{14}$$

$$\psi^{(2)}(x, y) = \int_{-1}^{1} \int_{-1}^{1} K^{(2)}(x, y; \xi, \eta) \phi^{(2)}(\xi, \eta)\, \mathrm{d}\xi\, \mathrm{d}\eta \tag{15}$$

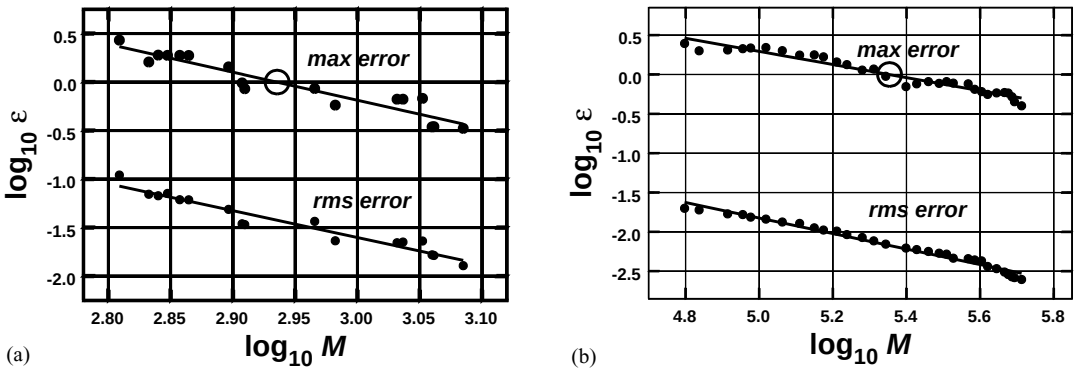


Fig. 5. Error ε in the output signals $\psi_i^{(1)}$ and $\psi_{ij}^{(2)}$ as a function of the number M of retained wavelet coefficients in the truncated kernel arrays $\tilde{K}_{il}^{(1)}$ and $\tilde{K}_{ij,lm}^{(2)}$ at fixed, 16 pixel resolution. (a) 1D; (b) 2D.

Residing conceptually between $K^{(1)}$ and $K^{(2)}$ is a bilinear functional:

$$\iint \mathcal{K}(x;\xi,\eta)\, f(\xi) g(\eta)\, \mathrm{d}\xi\, \mathrm{d}\eta,$$

$$|\mathcal{K}(x;\xi,\eta)| \le \frac{1}{(x-\xi)^2 + (x-\eta)^2} \tag{16}$$

For $K^{(1)}$ and $\mathcal{K}$ theorems on $O(N)$ summations are available (Beylkin et al., 1991; Harten & Yad-Shalom, 1994).

For a given discretization of the ξ or (ξ, η) domain (represented by the total number N of pixels, irrespective of dimension), various truncation levels were applied to the DWT of the corresponding kernel array $K_{i,l}^{(1)}$ or $K_{ij,lm}^{(2)}$, and the maximum error $\varepsilon_{\max}$ of the discrete output $\psi_i^{(1)}$ or $\psi_{ij}^{(2)}$ was recorded as a function of the total number M of retained wavelet coefficients, which directly reflects the operation count in the summations. Plotted logarithmically, these $\varepsilon_{\max}$ versus M data were smoothed by least-squares linear regression (Fig. 5), and the linear fit was then used to interpolate the number of wavelet coefficients needed to achieve a given accuracy. Results for various resolutions in both one and two dimensions are combined in Fig. 6, which shows the scaled operation count M/N or $M/[N(\log N)^\alpha]$ as a function of the number N of pixels. Holding the error fixed, the scaling in both one and two dimensions seems to be $O(N)$ [Fig. 6a]; it is certainly bounded by $N(\log N)^{1/2}$. Fixed error is not entirely realistic as a criterion, because the whole purpose of higher resolution is to diminish the discretization error in the integrals (14) and (15). A safe overestimate comes from taking $\varepsilon_{\max} \propto N^{-1/2}$. This assumption (i) is more stringent than the $O(N_\mathrm{p}^{-1/2} \log N_\mathrm{p})$ error scaling of the particulate discretization (1) of a liquid drop; and (ii) assumes that N is increased proportionally to N_p, which is unnecessary according to Fig. 3. From Fig. 6b we see that $M = O[N(\log N)^\alpha]$. The same asymptotic scalings would be expected to apply in 3D simulations as well.

5. Simulations of drop dynamics

Since the 16 pixel resolution that was computationally tractable for 3D problems lies well below the asymptotic regime of the operation count (Fig. 6), we shall here be content with the reasonable dimensional extrapolation of the previous section, and simply exhibit typical acceleration factors with reference to two test problems. Shown in Fig. 7 are results of computer simulations whereby the configurational evolution of two buoyant, miscible drops is modeled with swarms of particles. The smaller, leading drop flattens out

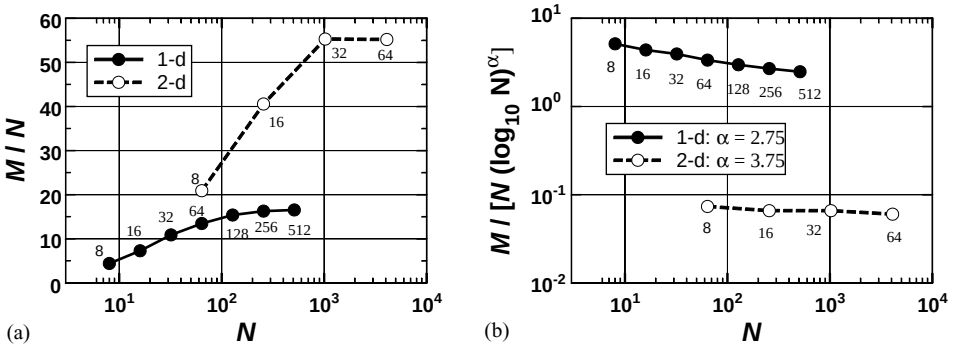


Fig. 6. Scaling of the operation count M with the number of pixels N for wavelet compression of the 1D and 2D kernels defined by Eqs. (10) and (11). Numbers by the points indicate the linear pixel resolution. (a) Fixed accuracy $\varepsilon_{\max} = 1\%$; (b) increasing accuracy $\varepsilon_{\max} \propto N^{-1/2}$.

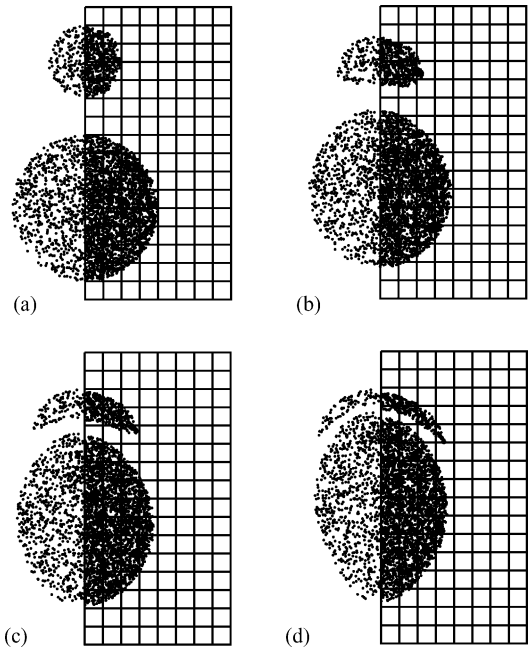


Fig. 7. Point-force simulations of two buoyant, miscible drops rising through quiescent liquid. Results from the boundary-integral method are available (Manga & Stone, 1993) for comparison, although axisymmetry of this particular test problem was not exploited here. Lengths are made dimensionless with respect to the radius of the larger drop in the starting configuration (a). We use a second-order Runge–Kutta scheme to integrate the ODE system (1) numerically; the interval between successive images amounts to 1000 time steps. To reveal internal structures, only particles within a slice of width 0.4 straddling the meridian plane are shown. The left (sparser) half of each swarm corresponds to the direct particle summation method (1) for 5000 particles. The right (denser) half illustrates the wavelet method applied to a swarm of 18,850 particles, lumped according to a $16 \times 16 \times 16$ pixel grid. The original list of about 50 million (3×16^6) wavelet coefficients $\hat{\boldsymbol{F}}_{IJK,LMN}$ was truncated to only 3.22%. (a) Initial configuration, for which the radius of the larger drop is unity and both the radius of the smaller drop and the gap between the drops is 0.5. (b) $t = 5$; (c) $t = 10$; (d) $t = 15$.

into a thin lens and coats the larger trailing drop, which becomes elongated (Manga & Stone, 1993). The wavelet-based calculations for the dots on the right-hand side of each graph were based upon a swarm of 18,850 particles lumped within a $16 \times 16 \times 16$ pixel grid, and used 3.22% of the wavelet coefficients $\hat{\boldsymbol{F}}_{IJK,LMN}$. Three thousand time steps took less than 36 min to compute on an IBM P690 Unix Server, and these included nonessential diagnostic computations. The dots on the left represent a direct implementation of the summation (1) for 5000 particles, which required more than 28 h. Extrapolated to 18,850 particles, the computation time would be more than 16 days. The accelerated simulations would thus be 676 times faster, of which a speed-up factor of 31.1 is attributable to the truncated wavelet basis, and the remaining factor of 21.8 comes from lumping the particles together into pixels. (Generating the wavelet coefficients themselves required less than 3 min, and these could be used for any simulation.) Note particularly that the thin lens shape and narrow gap between the two drops are accurately resolved by

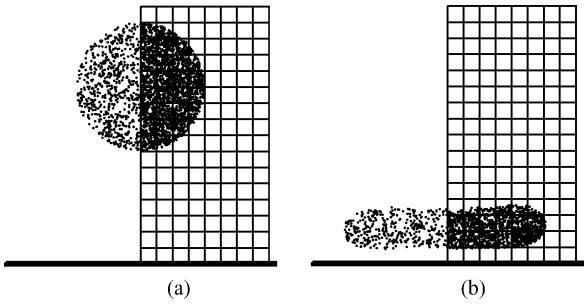


Fig. 8. Point-force simulations of a miscible drop sedimenting toward a horizontal, no-slip surface. Same description as for Fig. 7, except for the wall-effect Stokeslet (Blake, 1971; Dąbroś, 1985) for the particle interactions. Left (sparser) side: direct summation (1) for 4189 particles. Right (denser) side: wavelet method for 16,755 particles. The original list of about 50 million wavelet coefficients was trunctated to 3.94%. (a) Initial configuration, for which the drop-wall gap is 1.75 drop radii; (b) $t = 30$ (6000 time steps).

pixels which are of similar dimensions, owing to the global nature of the hydrodynamic interactions; cf. Eqs. (1) and (2).

Figure 8 shows the flattening and distention into an oblate shape of a miscible drop sedimentating toward a horizontal wall with no-slip conditions. The wavelet method (right side) required less than 85 min for 6000 time steps with 16,755 particles. (The one-time generation of the wavelet coefficients required less than 70 min.) The direct summation (left side) required more than 69 h for 4189 particles; see Machu (2001) for the same type of simulation with different parameter values. Extrapolated to 16,755 particles the computation time would be 46 days. Wavelet truncation and particle lumping are responsible for speed-up factors of 25.3 and 31.1, respectively.

6. Concluding remarks

Widely used in physics (Carrier et al., 1988; Greengard, 1994; Greengard & Rokhlin, 1987; Hockney & Eastwood, 1988; Rokhlin, 1983) the simplicity and robustness of particle methods makes them useful for simulating low-Reynolds-number flows of drops at unit viscosity ratio that involve complex interfacial evolutions such as coalescence and rupture (Machu et al., 2001; Nitsche & Schaflinger, 2001). For the case of infinite Bond number, this article has illustrated how wavelets remove the bottleneck in computational speed, with only inconsequential losses of accuracy.

The pyramical structure of the DWT, which is vaguely analogous to the tree-coding of fast multipole methods, is encapsulated in one simple matrix multiplication, and may be useful for other particle interaction laws as well: molecular dynamics (Greenspan, 1990; Moseler & Landman, 2000; Murad & Law, 1999), dissipative particle dynamics (Clark, Lal, Ruddock, & Warren, 2000; Hoogerbrugge & Koelman, 1992; Jones, Lal, Ruddock, & Spenley, 1999), and smoothed particle hydrodynamics (Monaghan, 1992; Nitsche & Zhang, 2002).

Acknowledgements

L.C.N. would like to thank Dr. Teng Yong Ng for his generous hospitality and access to the excellent facilities and enlightening atmosphere at the Institute of High Performance Computing (IHPC), Singapore, where part of this work was completed. Grateful thanks are also due to Professor Nhan Phan-Thien, National University of Singapore, for kindly helping to facilitate that 5-week research visit. This paper is dedicated to the memory of Professor Uwe Schaflinger (Institute of Fluid Dynamics and Heat Transfer, Technical University Graz), with whom we began investigating particles and drops.

References

Adachi, K., Kiriyama, S., & Koshioka, N. (1978). The behaviour of a swarm of particles moving in a viscous fluid. *Chemical Engineering Science, 33*, 115–121.

Arecchi, F. T., Buah-Bassuah, P. K., Francini, F., Pérez-Garcia, C., & Quercioli, F. (1989). An experimental investigation of the break-up of a liquid-drop falling in a miscible fluid. *Europhysics Letters, 9*, 333–338.

Beylkin, G., Coifman, R., & Rokhlin, V. (1991). Fast wavelet transforms and numerical algorithms I. *Communications on Pure and Applied Mathmetics, XLIV*, 141–183.

Blake, J. R. (1971). A note on the image system for a stokeslet in a no-slip boundary. *Proceedings of the Cambridge Philosophical Society, 70*, 303–310.

Carrier, J., Greengard, L., & Rokhlin, V. (1988). A fast adaptive multipole algorithm for particle simulations. *SIAM Journal on Scientific and Statistical Computing, 9*, 669–686.

Chew, W. C., Jin, J.-M., & Michielssen, E. (Eds.), (2000). *Fast and efficient algorithms in computational electromagnetics*. Norwood, MA: Artech House.

Clark, A. T., Lal, M., Ruddock, J. N., & Warren, P. B. (2000). Mesoscopic simulation of drops in gravitational and shear fields. *Langmuir, 16*, 6342–6350.

Cristini, V., Blawzdziewicz, J., & Loewenberg, M. (1998). Drop breakup in three-dimensional viscous flows. *Physics of Fluids, 10*, 1781–1783.

Cristini, V., Blawzdziewicz, J., & Loewenberg, M. (2001). An adaptive mesh algorithm for evolving surfaces: Simulations of drop breakup and coalescence. *Journal of Computational Physics, 168*, 445–463.

Dąbroś, T. (1985). A singularity method for calculating hydrodynamic forces and particle velocities in low-Reynolds-number flows. *Journal of Fluid Mechanics, 156*, 1–21.

Davis, R. H. (1999). Buoyancy-driven viscous interaction of a rising drop with a smaller trailing drop. *Physics of Fluids, 11*, 1016–1028.

Greengard, L. (1994). Fast algorithms for classical physics. *Science, 265*, 909–914.

Greengard, L., & Rokhlin, V. (1987). A fast algorithm for particle simulations. *Journal of Computational Physics, 73*, 325–348.

Greenspan, D. (1990). Quasi-molecular simulation of large liquid drops. *International Journal for Numerical Methods in Fluids, 10*, 247–257.

Harten, A., & Yad-Shalom, I. (1994). Fast multiresolution algorithms for matrix–vector multiplication. *SIAM Journal for Numerical Analysis, 31*, 1191–1218.

Hockney, R.W., & Eastwood, J.W. (1988). *Computer simulation using particles*. Philadelphia: IOP Publishing.

Hoogerbrugge, P. J., & Koelman, J. M. V. A. (1992). Simulating microscopic hydrodynamic phenomena with dissipative particle dynamics. *Europhysics Letters, 19*, 155–160.

Jones, J. L., Lal, M., Ruddock, J. N., & Spenley, N. A. (1999). Dynamics of a drop at a liquid/solid interface in simple shear fields: A mesoscopic simulation study. *Faraday Discussions, 112*, 129–142.

Joseph, D. D., & Renardy, Y. Y. (1993). *Fundamentals of two-fluid dynamics*. New York: Springer.

Kaiser, G. (1994). *A friendly guide to wavelets* (section 7.2). Boston: Birkhaeuser.

Koh, C. J., & Leal, L. G. (1989). The stability of drop shapes for translation at zero Reynolds-number through quiescent fluid. *Physics of Fluids A, 1*, 1309–1313.

Kojima, M., Hinch, E. J., & Acrivos, A. (1984). The formation and expansion of a toroidal drop moving in a viscous fluid. *Physics of Fluids, 27*, 19–32.

Machu, G. (2001). Interfacial phenomena in suspensions. Ph.D. thesis, Institute of Fluid Dynamics and Heat Transfer, Technical University Graz, Austria.

Machu, G., Meile, W., Nitsche, L. C., & Schaflinger, U. (2001). Coalescence, torus formation and break up of sedimenting drops: experiments and computer simulations. *Journal of Fluid Mechanics, 447*, 299–336.

Manga, M. (1997). Interactions between mantle diapirs. *Geophysical Research Letters, 24*, 1871–1874.

Manga, M., & Stone, H. A. (1993). Buoyancy-driven interactions between two deformable viscous drops. *Journal of Fluid Mechanics, 256*, 647–683.

Manga, M., Stone, H. A., & O'Connell, R. J. (1993). The interaction of plume heads with compositional discontinuities in the earth's mantle. *Journal of Geophysics Research-Solid Earth, 98*, 19979–19990.

Monaghan, J. J. (1992). Smoothed particle hydrodynamics. *Annals of Astronomy and Astrophysics, 30*, 543–574.

Moseler, M., & Landman, U. (2000). Formation, stability, and breakup of nanojets. *Science, 289*, 1165–1169.

Murad, S., & Law, C. K. (1999). Molecular simulation of droplet collision in the presence of ambient gas. *Molecular Physics, 96*, 81–85.

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

Nitsche, L. C., & Schaflinger, U. (2001). A swarm of Stokeslets with interfacial tension. *Physics of Fluids, 13*, 1549–1553.

Nitsche, L. C., & Zhang, W. (2002). Atomistic SPH and a link between diffusion and interfacial tension. *AIChE Journal, 48*, 201–211.

Phillips, R. J. (2003). A singularity method for calculating time-dependent viscoelastic flows with integral constitutive equations. *Journal of Fluid Mechanics, 481*, 77–104.

Pozrikidis, C. (1990). The instability of a moving viscous drop. *Journal of Fluid Mechanics, 210*, 1–21.

Pozrikidis, C. (1992). *Boundary integral and singularity methods for linearized viscous flow*. New York: Cambridge University Press.

Press, W. H., Teukolsky, S. A., Vetterling, W. T., & Flannery, B. P. (1992). Numerical recipes in Fortran 77. *The art of scientific computing* (2nd ed., section 13.10). New York: Cambridge University Press.

Rallison, J. M., & Acrivos, A. (1978). A numerical study of the deformation and burst of a viscous drop in an extensional flow. *Journal of Fluid Mechanics, 89*, 191–200.

Rokhlin, V. (1983). Rapid solution of integral equations of classical potential theory. *Journal of Computational Physics, 60*, 187–207.

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

Schaflinger, U., & Machu, G. (1999). Interfacial phenomena in suspensions. *Chemical Engineering & Technology, 22*, 617–619.

Thomson, J. J., & Newall, H. F. (1885). On the formation of vortex rings by drops falling into liquids, and some allied phenomena. *Proceedings of the Royal Society of London, 39*, 417–435.

Walker, J. S. (1999). *A primer on wavelets and their scientific applications* (Chapter 1). New York: Chapman & Hall/CRC Press.

Zinchenko, A. Z., Rother, M. A., & Davis, R. H. (1999). Cusping, capture, and breakup of interacting drops by a curvatureless boundary-integral algorithm. *Journal of Fluid Mechanics, 391*, 249–292.