Modeling multiphase flows using a novel 3D adaptive remeshing algorithm

Russell Hooper, Vittorio Cristini, Sundeep Shakya[1]
John S. Lowengrub[2], Jeffrey J. Derby and Christopher W. Macosko

Department of Chemical Engineering and Materials Science,
Army HPC Research Center, and Minnesota Supercomputing Institute
University of Minnesota
421 Washington Avenue S.E., Minneapolis MN 55455-0132

[1]*Department of Computer and Information Science*
University of Mississippi
302 University, MS 38677

[2]*School of Mathematics*
University of Minnesota
206 Church Street, Minneapolis MN 55455

Abstract

A novel three-dimensional adaptive meshing algorithm is presented and applied to finite-element simulations of multiphase fluid flows. A three-dimensional domain enclosing another phase is discretized by an unstructured mesh of tetrahedra constructed from a triangulated surface of the phase boundaries. Complete remeshing is performed after each time step. The boundary mesh is reconstructed using an existing algorithm employing element addition/subtraction, edge swapping based on Delaunay triangulation and spring-like dynamical relaxation. The volume mesh is then generated from the boundary using the commercial software Hypermesh. The resulting adaptive discretization maintains resolution of prescribed local length scales. We demonstrate our method with finite-element simulations of deformable drops subjected to simple shear under Stokes flow conditions. Steady drop shapes in agreement with experimental data as well as the evolution of slender fluid filaments characteristic of drop breakup are accurately described.

1 Introduction

Complex three-dimensional geometries are present in a wide variety of physical and biological systems and have applications to many industrial problems. An important example is the motion of multiphase fluids in complex geometries in which the interfaces deform significantly. Finite-element methods (FEM) provide a natural choice to describe evolution of such systems using numerical simulations. They allow the description of such complex physical phenomena as inertial effects, thermal effects,

species transport, surfactant effects and non-Newtonian behavior. Boundary-element methods have also been developed, although limited to a more restricted class of physical systems. Several powerful 3-dimensional mesh generation algorithms have been developed [1, 2, 3, 4] and applied to problems involving mesh distortion [5]. However, little has been done for problems with very large interface deformations where the problem dynamics are dominated by the shape of the interface, e. g. large deformations involving capillarity. The main difficulty in performing such simulations is in accurately resolving very dissimilar length scales within the same domain. Traditional 3-dimensional FEM employing mesh tracking schemes require the finest mesh resolution to be maintained during the entire deformation and therefore are restricted to problems involving modest shape changes [6, 7, 8].

This difficulty has recently been overcome for problems requiring only surface meshes by an adaptive surface remeshing algorithm [9, 10], which has been applied to boundary-integral simulations of Stokes multiphase flow. In this paper, we combine the surface remeshing algorithm with efficient and robust 3-dimensional meshing provided by the commercial package, Hypermesh, to produce an adaptive, 3-dimensional finite element method. The ability of our algorithm to accurately describe significant drop deformations is demonstrated by simulations of deformable drops undergoing simple shearing in Stokes flow [11]. Our goal in performing simulations that can be compared to experiments and other established numerical methods is to establish the credibility of our method and to pave the way for simulating more complex physical systems beyond the capability of other methods.

We summarize the surface and domain remeshing algorithms in §2. Application to the study of subcritical and supercritical drop deformations is contained in §3 followed by conclusions in §4.

2 Algorithm

2.1 Adaptive surface remeshing

The algorithm used for adaptive reconstruction of triangulated surface meshes is described in detail in Cristini *et al.* [9, 10].

Briefly, the resolution of solid angle, i. e. curvature of the drop surface, is kept uniform over the triangulated surface. A prescribed level of accuracy is maintained during a simulation by adaptively remeshing after each time step. This is achieved by introducing a node density function

$$\rho = C_0 <\kappa>^2 , \tag{1}$$

where κ is defined in Cristini *et al.* [10] as a smooth measure of the local curvature of the surface, and C_0 is a constant that determines the resolution. Correspondingly, the instantaneous number of nodes, N_S, that discretize the drop free surface is a function of the drop shape,

$$N_S = C_0 \int_S <\kappa>^2 \, \mathrm{d}S. \tag{2}$$

The constant C_0 in eq. (1) is determined by specifying the number of nodes N_0 used to discretize the initial spherical drop surface. The corresponding resolution of curvature is maintained during a simulation through a sequence of local restructuring operations that are summarized next.

The surface mesh is modeled as a dynamical system of damped massless springs connecting the nodes: each spring has a tension $l - l_0$, where l is the instantaneous edge length and $l_0 \sim \rho^{-1/2}$ is the equilibrium length. Equilibration velocities of the nodes are determined as the resultant of local spring tensions projected onto the surface. The system of springs has well-defined minimum-energy equilibrium states corresponding to zero equilibration velocities of all nodes. An equilibrium configuration is attained iteratively by evolving the node positions with the equilibration velocities. Between equilibration steps, optimal node connectivity is maintained by local reconnection according to a Delaunay criterion [12]. The desired local refinement (1) is maintained by addition and subtraction of nodes in the regions of the surface were tensions $|l - l_0|$

are large. This allows the global minimum-energy equilibrium state, corresponding to $l - l_0 \approx 0$, to be approached to within a specified tolerance everywhere on the free surface.

2.2 Three-dimensional mesh generation

A 3-dimensional mesh of unstructured tetrahedra is generated from the triangulated surfaces of the walls bounding the matrix fluid and the drop free surface. The latter is adaptively remeshed as described in the preceding section, while the surface mesh for the bounding walls remains the same for the entire transient simulation. Node and connectivity information defining the surface meshes is passed to the commercial meshing package, Hypermesh, via an import translator created from C libraries supplied with the licensed software distribution. The algorithm used by Hypermesh is based on an advancing-front, node placement algorithm [13]. Nodes are inserted into the domain, and new elements are created by subdividing existing elements using the new nodes. Mesh quality is then improved by local reconnection based on combined Delaunay and min-max criteria implemented iteratively. The final 3D mesh is exported using an output template into a form used by the FEM solver.

3 Deformable drop dynamics

In this section we demonstrate the algorithm described in §2 by application to finite-element simulations of deformable drops in Stokes flow.

3.1 Assumptions

We consider non buoyant immiscible fluid drops of volume $\frac{4}{3}\pi a^3$ and viscosity $\hat{\mu}$ in a fluid of viscosity μ. The drop surfaces are surfactant-free, and no temperature gradients are present so that the interfacial tension, σ, is constant. We simulate simple shear flows corresponding to experiments performed by Guido & Simeone [14]. For these, the matrix fluid is bounded by two parallel rigid plates (xz–plane) having

length $2d_x$, width $2d_z$ and separation $2d_y$. The plates translate with respect to each other in the x–direction. The matrix fluid is held between the plates by capillarity. The flow field is then a steady shear flow with magnitude $\dot{\gamma}$ of the velocity gradient $\mathbf{D}$, driven by the motion of the plates impulsively started at time $t = 0$,

$$\mathbf{v}^{\infty} = \mathbf{D} \cdot \mathbf{x} \ . \tag{3}$$

Equation (3) is made dimensionless using the flow time scale $\tau_{\dot{\gamma}} = \dot{\gamma}^{-1}$ and drop radius length scale a. Shear flow is characterized by the only nonzero component $D_{xy} = 1$.

We consider conditions for which the Reynolds number based on the drop size is low, and thus assume that the Stokes equations describe the fluid motion. It follows that the relevant parameters for drop deformation are the capillary number [15]

$$Ca = \tau_{\sigma}/\tau_{\dot{\gamma}}, \tag{4}$$

which is the ratio of capillary relaxation time $\tau_{\sigma} = \mu a/\sigma$ to flow time defined above, the viscosity ratio $\lambda = \hat{\mu}/\mu$, and the ratios d_i/a $(i = x, y, z)$ that account for hydrodynamic interactions with the walls. Stationary deformed drop configurations exist in the flow below a critical $O(1)$ capillary number [11]. Above the critical capillary number, drops continuously elongate until breakup occurs.

3.2 Finite-element method

The finite element method used in this work is derived from Zhou & Derby [16]. A detailed description of their method applied to single-phase particle sintering can be found elsewhere [17]. To summarize, the Galerkin finite element method is used to cast the Stokes and mesh displacement equations in a weak form. Capillarity at the drop free surface is implemented naturally, and second derivatives associated with surface curvature are circumvented by integrating this term by parts [18, 19]. All field variables are discretized over an unstructured mesh of tetrahedra using continuous linear basis functions [20]. Spurious pressure modes are avoided by applying pressure-stabilized Petrov-Galerkin (PSPG) weighting to the continuity residuals [21]. Solving for the discretized velocity, pressure and nodal displacements fully coupled

and implicitly in time leads to a nonlinear system of equations which are solved using Newton-Raphson iterations. The resulting linear systems are solved iteratively using generalized minimum residuals (GMRES) [22] with diagonal preconditioning [23].

Our method differs from that of Zhou & Derby [16] by the inclusion of additional fluid phase(s), no assumptions of problem symmetry and, most importantly, the way in which the drop free surface is tracked. Details of the latter are reserved for the next subsection.

In this work, the additional fluid phase is the matrix fluid surrounding the drop and is treated as bounded by a box of dimensions, $2d_x$ x $2d_y$ x $2d_z$. An essential boundary condition is imposed for the velocity at the box walls. For the inlet ($x = -d_x$), outlet ($x = d_x$), top ($y = d_y$) and bottom ($y = -d_y$), fully developed shear flow is imposed,

$$\mathbf{v} = \mathbf{v}^{\infty} \tag{5}$$

with $\mathbf{v}^{\infty}$ defined by eq. (3). At the remaining two boundaries ($z = \pm d_z$), zero tangential stress and zero mass flux are imposed as natural and essential conditions, respectively.

3.3 Mesh evolution

Whereas Zhou & Derby [16] solve a fully implicit time-evolution problem for the velocity, pressure and nodal displacements, we employ a decoupled scheme which facilitates adaptive remeshing. For Stokes flow, this amounts to solving a fixed-boundary problem for velocity and pressure and then updating the nodal positions of the drop free surface via explicit time integration of the velocity,

$$\mathbf{n} \cdot \frac{\mathrm{d}\mathbf{x}_P}{\mathrm{d}t} = \mathbf{n} \cdot \mathbf{v}(\mathbf{x}_P). \tag{6}$$

where $\mathbf{x}_P$ are nodes on the drop free surface, and $\mathbf{n}$ is a unit normal vector pointing away from the drop surface. In our implementation, equation (6) was integrated using a second-order Runge-Kutta scheme with variable time step $\Delta t = \epsilon l_{\min}$ set by a convective-stability constraint

$$\Delta \mathbf{x}_P < l_{\min} \ , \tag{7}$$

where $l_{\min}$ is the shortest node-to-node distance over the mesh, and $\epsilon = 0.5Ca$. Normal vectors $\mathbf{n}$ on the drop free surface are calculated using a local paraboloid fit [24]. After updating $\mathbf{x}_P$, the surface mesh is reconstructed using the algorithm described in §2.1. The new volume mesh is then generated from the boundaries according to §2.2. The next time step is taken by repeating the entire solution process with the new mesh.

3.4 Results of simulations

Finite element simulations were performed on single CPU-nodes of an IBM-SP at the Minnesota Supercomputer Institute. A CPU-node consists of four 222 MHz Power3 processors sharing 16 Gb memory. The level of mesh refinement varied during the course of the simulation, becoming greater during later stages as maximum curvature increased. A typical simulation required on the order of 1000 time steps with the last steps requiring about 30 CPU minutes to complete. Total time required was about 48 CPU hours.

3.4.1 Stationary shapes

Here we illustrate the accuracy of our adaptive finite-element algorithm by simulating a subcritical deformation for which a steady shape is attained. The numerical result is compared to experimental results obtained independently by Guido (private communication).

The evolution of the dimensionless drop length in the flow direction, R_x/a, is shown in figure 1 as a function of dimensionless time, $\gamma = t/t_{\dot{\gamma}}$. The numerical results are seen to agree with the experimental data to within the accuracy of the experiment ($\sim 10\%$). Further agreement is shown in figure 2 where a cross-section of the mesh in the xy-plane through the center of the drop ($z = 0$) is shown at steady state conditions along with the boundary of the drop from the experiment. The higher element density near the drop tips reflects the use of adaptivity to resolve these regions of larger curvature.

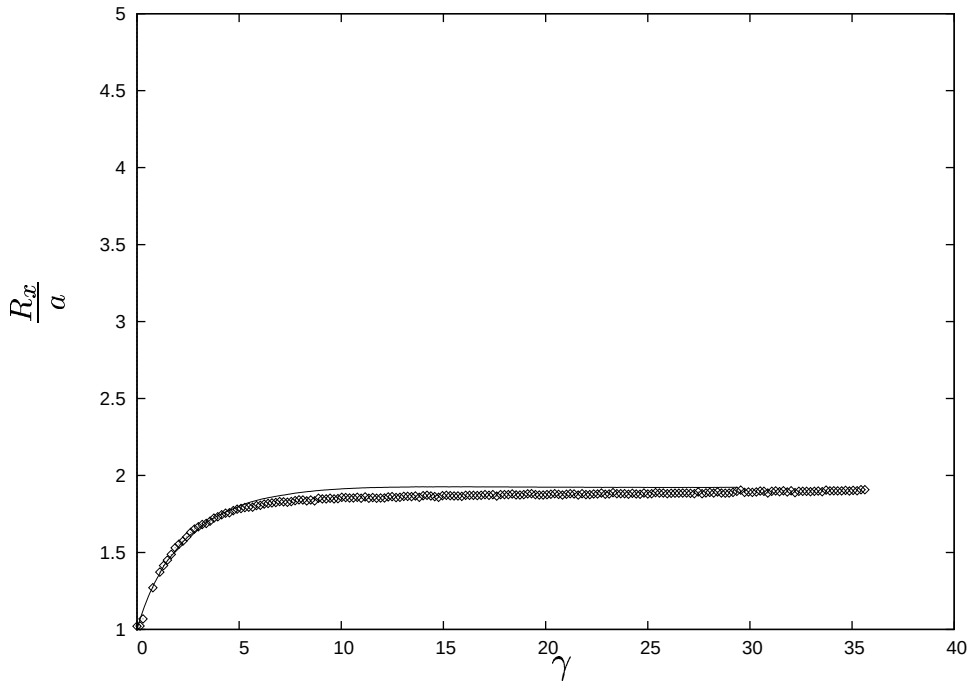

Figure 1: Dimensionless drop length in the flow direction, R_x/a, as a function of flow time, γ, for simple shear with $Ca = 0.38$, $\lambda = 1.0$ and $d_x = d_y = d_z = 10a$. Numerical simulation (solid curve) corresponds to $N_0 = 110$ and is compared to experimental data (open symbols) with the same parameter values obtained by S. Guido (personal communication).

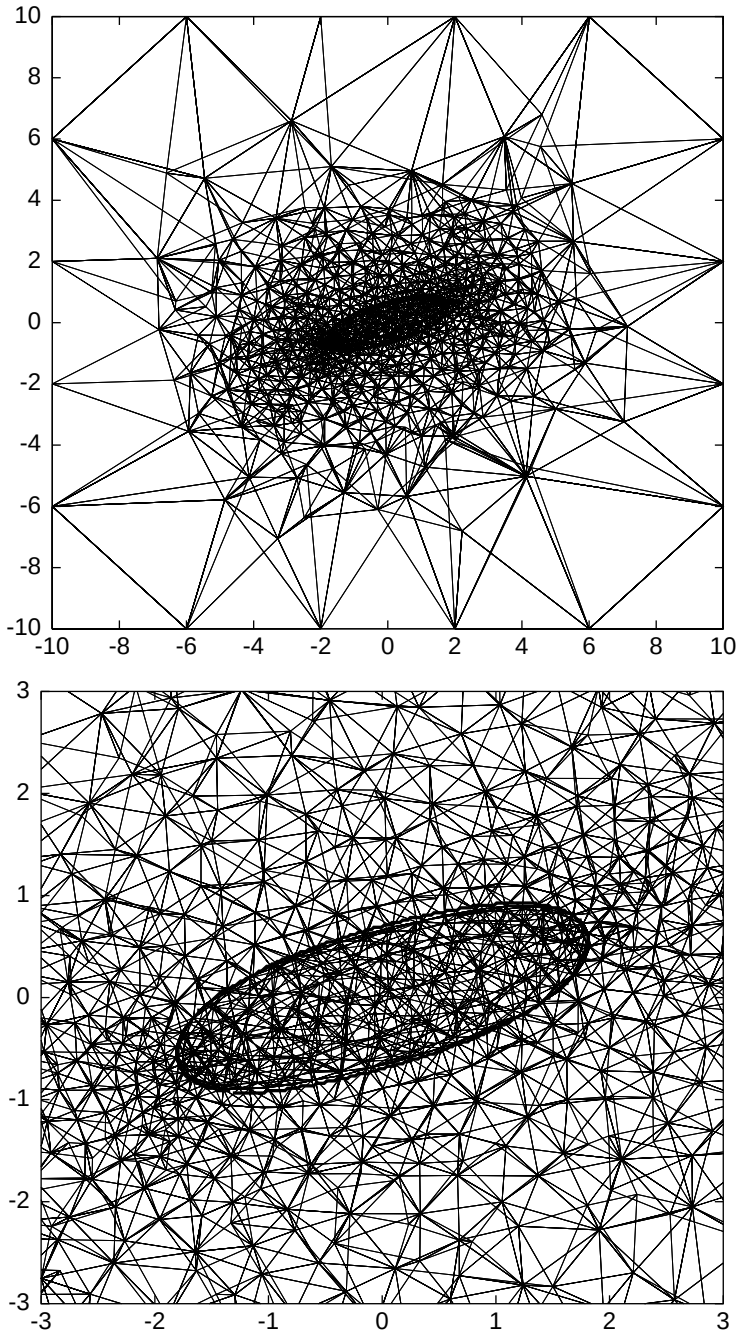

Figure 2: Mesh cross-section in xy-plane at $z = 0$ for stationary drop configuration from numerical simulation corresponding to $N_0 = 110$ with $Ca = 0.38$ and $\lambda = 1.0$. Heavy line is drop shape determined experimentally by S. Guido (personal communication). Complete domain (top) and zoom of drop region (bottom) are shown.

3.4.2 Drop breakup

We now test the ability of our adaptive FEM to handle a more severe simulation of supercritical deformation approaching drop breakup in simple shear flow. In figure 3 the drop surface mesh is shown at various times for the parameters reported. Fewer elements are needed early in the simulation allowing for efficient computations. The evolution of dimensionless drop length is compared to converged boundary integral results in figure 4. Good agreement is maintained to a drop strain, R_x/a, of about 2.5. Faster drop deformation is predicted by the FEM beyond this point likely as a result of the presence of the walls 10 radii away. Use of the FEM necessitates a finite outer fluid domain, whereas the boundary element method treats the outer fluid as truly infinite in extent.

In spite of the discrepancy at larger drop deformation, the qualitative features of the approach to breakup are accurately described. The drop elongates and rotates becoming dumbbell in shape and characterized by two symmetric bulbous ends separated by a neck. The bulbs assume a roughly stable size and shape determined by the value of Ca and translate approximately parallel to the flow. Correspondingly the neck shrinks, and two bridges of fluid eventually form that connect the slender neck to the bulbs. The bridges are unstable and would ultimately pinch off in finite time [9]. The approach to pinch-off is evident in figure 5 which shows the mesh cross-section as before for the final time step simulated. The blow-up in the neck region depicts the high concentration of elements required to maintain a specified resolution of curvature. The variation in discretization illustrates how dissimilar length scales are well-resolved by adaptivity and that computational efficiency is realized by adding elements only as needed.

4 Conclusions

We have described and demonstrated a novel 3-dimensional adaptive finite element algorithm capable of accurately simulating deformable drop dynamics involving significant deformations. Both subcritical and supercritical behavior in Newtonian sys-

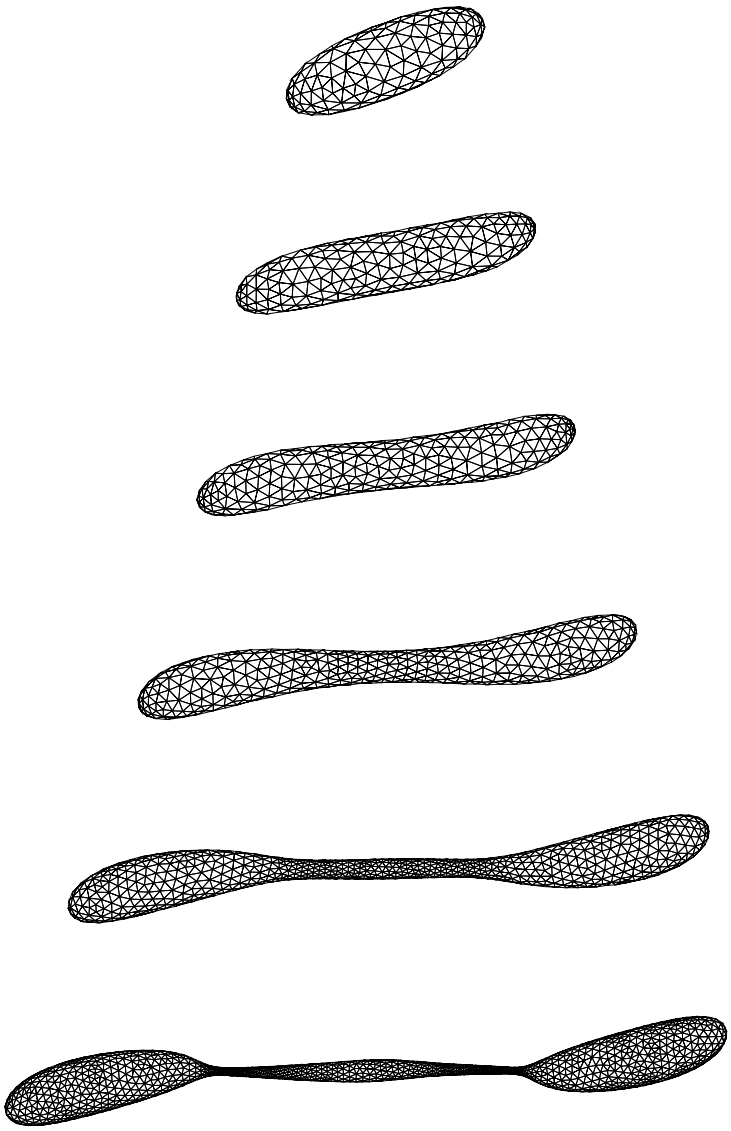

Figure 3: Drop evolution approaching breakup in shear flow for $Ca = 0.44$, $\lambda = 1$, and $d_x = d_y = d_z = 10a$. Mesh resolution corresponds to $N_0 = 110$. Dimensionless times from top to bottom are 3.6, 14.7, 25.1, 35.4, 43.8 and 48.1.

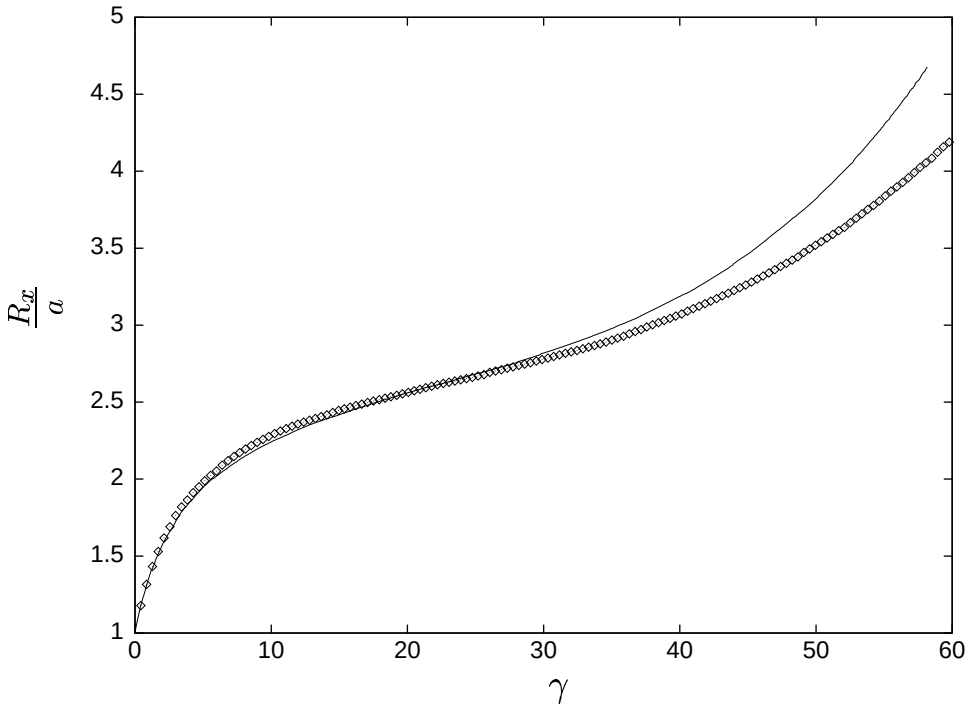

Figure 4: Dimensionless drop length in the flow direction, R_x/a, as a function of flow time, γ, for simple shear with $Ca = 0.44$, $\lambda = 1.0$ and $d_x = d_y = d_z = 10a$. Finite element simulation with $N_0 = 110$ (solid curve) is compared to boundary element simulation (open symbols) using the method of Cristini et al. [10]

tems without inertia are well-described as judged by comparison to experiments and converged boundary integral simulations, respectively. While such simulations are much more efficiently performed using other methods such as the boundary integral method, our numerical framework allows straightforward incorporation of additional physical phenomena that is awkward or impossible to implement with other methods. We are currently expanding our algorithm to include viscoelastic fluids via differential constitutive equations of the Jeffreys and Maxwell types. We are also able to explore wall effects, generalized Newtonian fluid behavior and multiple particle systems with our algorithm in its present form. Results for these and progress in implementing viscoelasticity are forthcoming.

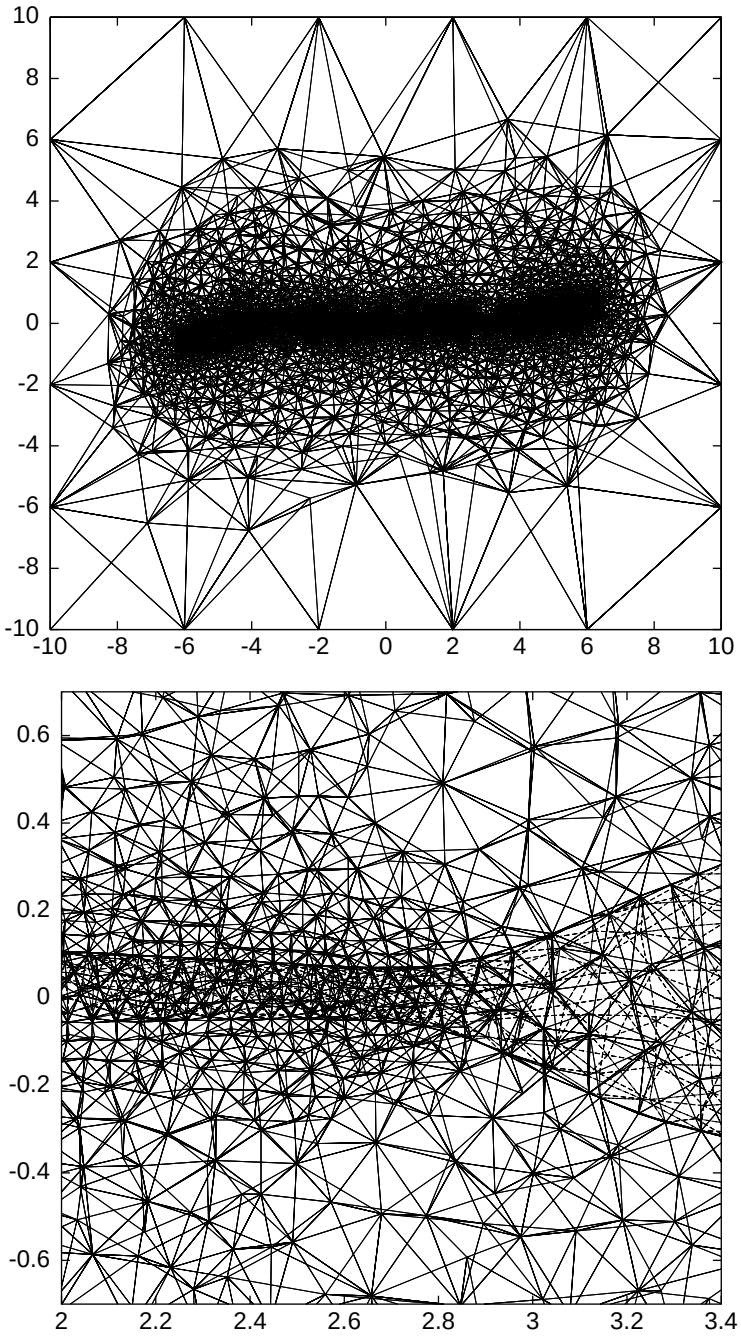

Figure 5: Mesh cross-section in xy-plane at $z = 0$ for the last time step of figure 4. Complete domain (top) and zoom of the pinching liquid bridge (bottom) are shown.

Acknowledgments

This work was supported in part by the Army High Performance Computing Research Center under the auspices of the Department of the Army, Army Research Laboratory cooperative agreement DAAH04-95-2-0003/contract DAAH04-95-C-0008, the content of which does not necessarily reflect the position or policy of the government, and no official endorsement should be inferred.

References

[1] P. L. George. *Automatic mesh generation, application to finite-element methods.* Wiley, Chichester, 1991.

[2] T. J. Baker. Developments and trends in three-dimensional mesh generation. *Appl. Numer. Math.*, 5:275–304, 1989.

[3] M. S. Shephard and M. K. Georges. Automatic three-dimensional mesh generation by the finite octree technique. *Int. J. Numer. Methods Eng.*, 32:709–749, 1991.

[4] P. L. George, F. Hecht, and E. Saltel. Automatic mesh generator with specified boundary. *Comput. Methods Appl. Mech. Eng.*, 92:269–288, 1991.

[5] A. A. Johnson and T. E. Tezduyar. Parallel computation of incompressible flows with complex geometries. *Intern. J. Numer. Meths. Fluids*, 24:1321–1340, 1997.

[6] P. A. Sackinger, P. R. Schunk, and R. R. Rao. A newton-raphson pseudo-solid domain mapping technique for free and moving boundary problems: a finite element implementation. *J. Comput. Phys.*, 125:83–103, 1996.

[7] R. A. Cairncross, P. R. Schunk, T. A. Baer, R. R. Rao, and P. A. Sackinger. A finite element method for free surface flows of incompressible fluids in three dimensions. part i. boundary fitted mesh motion. *Int. J. Num. Meth. Fluids*, 33:375–403, 2000.

[8] T. A. Baer, R. A. Cairncross, P. R. Schunk, R. R. Rao, and P. A. Sackinger. A finite element method for free surface flows of incompressible fluids in three dimensions. part ii. dynamic wetting lines. *Int. J. Num. Meth. Fluids*, 33:405, 2000.

[9] V. Cristini, J. Bławzdziewicz, and M. Loewenberg. Drop breakup in three-dimensional viscous flows. *Phys. Fluids*, 10:1781–1783, 1998.

[10] V. Cristini, J. Bławzdziewicz, and M. Loewenberg. An adaptive mesh algorithm for evolving surfaces: simulations of drop breakup and coalescence. in review, 2000.

[11] Howard A. Stone. Dynamics of drop deformation and breakup in viscous fluids. *Annu. Rev. Fluid Mech.*, 26:65–102, 1994.

[12] D. J. Mavriplis. Unstructured grid techniques. *Ann. Rev. Fluid Mech.*, 29:473–514, 1997.

[13] D. L. Marcum and N. P. Weatherill. Unstructured grid generation using iterative point insertion and local reconnection. *AIAA Journal*, 33:1619–1625, 1995.

[14] S. Guido and M. Simeone. Binary collisions of drops in simple shear flow by computer assisted video optical microscopy. *J. Fluid Mech.*, 357:1–20, 1998.

[15] S. Kim and S. J. Karrila. *Microhydrodynamics: principles and selected applications*. Butterworth-Heinemann, London, 1991.

[16] H. Zhou and J. J. Derby. A parallel, finite element method for three-dimensional, moving-boundary flows driven by capillarity and its application to viscous sintering. in review, 2000.

[17] Hua Zhou. *Three-dimensional finite-element modeling of viscous sintering*. PhD thesis, University of Minnesota, Minneapolis, 1998.

[18] Kenneth J. Ruschak. A method for incorporating free boundaries with surface tension in finite element fluid-flow simulators. *Intl. J. for Num. Meth. in Engr.*, 15:639–648, 1980.

[19] L.-W. Ho and A. T. Patera. Variational formulation of three-dimensional free-surface flows: natural imposition of surface tension to boundary conditions. *Intern. J. Numer. Meths. Fluids*, 13:691–698, 1991.

[20] T. J. R. Hughes. *The finite-element method: linear static and dynamic finite element analysis*. Prentice-Hall, Inc., Englewood Cliffs, NJ, 1987.

[21] T. J. R. Hughes, L. P. Franca, and M. Balestra. A new finite element formulation for computational fluid dynamics: V. circumventing the babuska-brezzi condition: A stable petrov-galerkin formulation of stokes problem accomodating equal-order interpolations. *Comput. Methods Appl. Mech. Engrg.*, 59:85–99, 1986.

[22] Y. Saad and M. Schultz. Gmres: A generalized minimum residual algorithm for solving nonsymmetric linear systems. *SIAM*, 7:856–869, 1986.

[23] A. Yeckel and J. J. Derby. Parallel computation of incompressible flows in materials processing: Numerical experiments in diagonal preconditioning. *Parallel Computing*, 23:1379–1400, 1997.

[24] A. Z. Zinchenko, M. A. Rother, , and R. H. Davis. A novel boundary-integral algorithm for viscous interaction of deformable drops. *Phys. Fluids A*, 9:1493–1511, 1997.