Small Oscillations of Viscous Water Droplets: A Numerical Simulation

David Stachnik
Harvey Mudd College
2 May 2008

I Introduction

I.A The Problem

In this project, we determine the time evolution of the surface of a viscous liquid droplet in an inviscid medium. We assume the only forces acting on the droplet are internal. The equilibrium state of a droplet is a sphere since this minimizes the surface tension and we restrict this study to small initial perturbations of spherical symmetry, but allow arbitrary viscosity. Given the initial displacement, we want to know the shape of the droplet at later times.

The approach taken closely follows that given in a 1977 article by Andrea Prosperetti (Ref. [1]). This approach involves numerically finding an inverse Laplace transform. We will explore two methods for this inversion: the fixed Talbot and Gaver-Wynn-rho algorithms.

I.B Motivation for the Problem

The study of droplets has many practical applications. A number of applications are given in *Dynamics of Droplets* by Frohn and Roth [2] and a few are those are described briefly below.

One straightforward application is fire suppression and spray cooling. For fire suppression, the size and spacing of the droplets affects their ability to quench a fire. Similarly, in spray cooling the droplet size affects the rate of cooling. Thus it is important to be able to determine the size of a droplet.

Another application is in medicine where inhalation is a common method of delivering a medicinal agent. Asthma inhalers are a prevalent example. The size and composition of the droplets affects where they are transported and may change by evaporation or condensation within the body. To control the quantity of drug delivered, having a repeatable metering system to deliver specific dosages is necessary.

The development of ink-jet printing has spurred the development of droplet-on-demand generators. Also, droplet formation and bubble growth have been studied experimentally and theoretically in this context.

One more interesting application is droplet manufacturing. Essentially, structures are created by depositing material droplet-by-droplet enabling intricate structures to be fabricated with low tolerances and without the need for finishing or polishing. This process requires highly stable controlled streams of droplets with low speed dispersion. The technology for droplet manufacturing is fairly new and much fundamental work remains to be done.

Other applications mentioned in Ref. [2] include aerosols, acid rain, microencapsulation, cloud physics, thermal spraying (coating with molten droplets), and solder jet technology.

II Theory

The derivation of the equations of motion is given in Ref. [1] and is quite involved; therefore, we only present an outline of some major steps here. Prosperetti describes a more detailed calculation involving the surface between two viscous fluids, and then reduces these to the case of a droplet where the outer fluid has no viscosity or density. In the sketch presented, we assume the droplet case from the outset.

II.A Derivation Sketch of Equation of Motion

To examine the behavior of small oscillations of a free incompressible, viscous droplet we begin with the Navier-Stokes equations, which describe the motion of fluids. In the absence of body forces, the Navier-Stokes equations and the condition of incompressibility are

$$\frac{\partial \mathbf{U}}{\partial t} + (\mathbf{U} \cdot \nabla)\mathbf{U} = \frac{1}{\rho} \operatorname{div} \sigma, \tag{1a}$$

$$\nabla \cdot \mathbf{U} = 0, \tag{1b}$$

where $\mathbf{U}$ is the velocity field, ρ is the density of the liquid, and the stress tensor σ is given in terms of the pressure p and viscosity μ by

$$\sigma_{ij} = -p\delta_{ij} + \mu \left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right)$$

in cartesian coordinates.

We then assume the surface Σ is a slightly perturbed sphere for which the oscillation modes are uncoupled. Thus the surface may be written, in spherical coordinates, as

$$\Sigma(t) : F(r, \theta, \phi, t) \equiv r - R - \epsilon a_n(t) Y_n^m(\theta, \phi) = 0 \tag{2}$$

where $0 < \epsilon \ll 1$ and Y_n^m is a spherical harmonic of degree $n \geq 2$. (The $n = 0$ harmonic is eliminated to conserve mass. For convenience, we work in the center of mass frame, which eliminates the $n = 1$ harmonic.) In all subsequent calculations, only terms of first order in ϵ are retained. (The superscript m is omitted from the amplitudes $a_n(t)$ because, in this linearized approximation, only the n value affects the amplitude.)

There is a kinematic boundary condition that relates the surface shape to the velocity field given by

$$\frac{\partial F}{\partial t} + (\mathbf{U} \cdot \nabla)F = 0 \tag{3}$$

for the case of a free droplet that we are considering. There are also dynamical boundary conditions on the tangential and normal components of the stress at the surface

$$-\mathbf{n} \times (\sigma \mathbf{n}) = 0, \tag{4a}$$

$$-\mathbf{n} \cdot (\sigma \mathbf{n}) = \zeta \nabla \cdot \mathbf{n}, \tag{4b}$$

where $\mathbf{n}$ is the normal to the surface, which require that the tangential stress is continuous and the discontinuity in the normal stress is equal to the surface tension ζ times the total curvature.

On strategy that has been developed to solve these equations is to divide the velocity and pressure fields into three parts

$$\mathbf{U} = \mathbf{u}_0 + \epsilon \mathbf{u}_p + \epsilon \mathbf{u}_\nu, \qquad p = p_0 + \epsilon p_p + \epsilon p_\nu$$

where the first terms ($\mathbf{u}_0$, p_0) describe entirely radial motion for $\epsilon = 0$, the second terms ($\mathbf{u}_p$, p_p) are the correction to the inviscid, irrotational ($\nabla \times \mathbf{U} = 0$) flow caused by the perturbation of spherical symmetry, and the final terms ($\mathbf{u}_\nu$, p_ν) are the correction due to the viscosity ν.

The solutions to the equations for the first two terms were worked out by Plesset [3]. These solutions must then be used to find equations for ($\mathbf{u}_\nu$, p_ν) which must be solved. Finally, the kinematic and dynamic boundary conditions must be enforced to the total velocity and pressure fields ($\mathbf{U}$, p). The results of this calculation yield an integro-differential equation for $a_n(t)$

$$\ddot{a}_n(t) + 2b_{n0}\dot{a}_n(t) + \omega_{n0}^2 a_n(t) + 2b_{n0}\beta_n \int_0^t Q(t-\tau)\dot{a}(\tau)\, d\tau = 0 \tag{5}$$

where $\dot{a}(t)$ denotes a time derivative, the decay constant $b_{n0} = (n-1)(2n+1)\nu/R^2$, the oscillation frequency $\omega_{n0}^2 = n(n-1)(n+2)\zeta/\rho R^3$, $\beta = (n-1)(n+1)/(2n+1)$, and the function $Q_n(t)$ is defined by its Laplace transform as

$$\tilde{Q}_n(p) = 2I_{n+3/2}(q)[2I_{n+3/2}(q) - qI_{n+1/2}(q)]^{-1} \tag{6}$$

where $\tilde{Q}(p)$ is a quotient of modified Bessel functions of the first kind and $q = R(p/\nu)^{1/2}$. Here we have denote the Laplace transform of a function by a tilde as

$$\tilde{f}(p) = \int_0^\infty f(t)e^{-pt}\, dt \tag{7}$$

where here p is the transformed equivalent to t and has nothing to do with the pressure also denoted by p previously (we are adhering to the notation used by Prosperetti).

We have now arrived at Eq. (5) which we want to solve. This equation reduces to a damped harmonic oscillator for small t. The integral term arises from the vorticity (a measure of local rotation given by the curl of the velocity field, $\nabla \times \mathbf{U}$). Prosperetti notes that the the acceleration at a given time t depends on the velocities in a time interval $\sim R^2/\nu$ prior to t (the time scale for vorticity diffusion) as well as on the current values of position and velocity. Thus the viscous effects lead to the integro-differential nature of Eq. (5). With the assumptions made, the surface Eq. (2) can be expressed

$$r(\theta, t) = R + \epsilon a_n(t) Y_n^m(\theta, \phi) \tag{8}$$

where now $a_n(t)$ is given by Eq. (5).

II.B Laplace Transform Solution to Equation of Motion

We will solve Eq. (5) using a Laplace transform approach. This works out quite nicely since the integral term is a convolution (which becomes a product of transformed functions) and

we can solve for $\tilde{a}(p)$ exactly. However, the inversion of $\tilde{a}_n$ cannot be done analytically (or at least, it has not been done) so we must invert numerically. For simplicity, we drop the subscript n from the following calculations.

We Laplace transform both sides of Eq. (5) using Eq. (7) and solve for the transformed version $\tilde{a}(p)$ of the amplitudes $a(t)$ to find

$$\tilde{a}(p) = \frac{1}{p}\left[a_0 + \frac{u_0 p - \omega_0^2 a_0}{p^2 + 2b_0 p + \omega_0^2 + 2\beta b_0 p\tilde{Q}(p)}\right] \tag{9a}$$

where we have used the initial conditions $a(0) = a_0$ and $\dot{a}(0) = u_0$ and the convolution theorem for Laplace transforms $\mathcal{L}$:

$$\mathcal{L}\left[\int_0^t f(t-\tau)g(\tau)\,d\tau\right] = \tilde{f}(p)\cdot\tilde{g}(p).$$

To simplify the numerical inversion, we make the following change to dimensionless variables:

$$\tau = (\zeta/\rho R^3)^{1/2}t, \qquad s = (\rho R^3/\zeta)^{1/2}p,$$
$$a^*(\tau) = a(t)/a_0, \qquad u_0^* = (\rho R^3/\zeta)^{1/2}u_0/a_0, \qquad \epsilon = \nu(\rho/R\zeta)^{1/2}.$$

(Note: Again adhering to Prosperetti's notation, here ϵ is a dimensionless viscosity and is not related to the parameter controlling the smallness of the perturbation from sphericity in Eq. (8).) In terms of these dimensionless variables Eq. (9a) becomes

$$\tilde{a}^*(s) = \frac{1}{s}\left[1 + \frac{su_0^* - (n-1)n(n+2)}{s^2 + 2(2n+1)(n-1)\epsilon s + +n(n+2)(n-1) + 2(n-1)^2(n+1)\epsilon s\tilde{Q}^*(s)}\right] \tag{9b}$$

where we have remembered an additional time factor arising from the dt in the transform integral and

$$\tilde{Q}^*(s) = 2I_{n+3/2}(q^*)[2I_{n+3/2}(q^*) - qI_{n+1/2}(q^*)]^{-1}$$

where $q^* = (s/\epsilon)^{1/2}$. Eq. (9b)

III Numerical Inversion of Laplace Transform

The inverse Laplace transform is given by

$$f(p) = \frac{1}{2\pi i}\int_{\gamma-i\infty}^{\gamma+i\infty}\tilde{f}(p)e^{pt}\,dp \tag{10}$$

where γ is a positive real number that lies to the right of all singularities of $\tilde{f}(p)$. Unfortunately, for Eq. (9b), the Bromwich integral Eq. (10) does not have an analytical solution so we must use numerical techniques (actually, this is the reason why it is a scientific computing project).

We will investigate two numerical methods for performing the inversion which both rely on multiple-precision computing to avoid numerical round-off errors, which arise in traditional sixteen-digit precision computing. An article by Abate and Valkó [4] details both methods and provides Mathematica implementations of them. In the following sections, we summarize the findings of Abata and Valkó.

III.A Fixed Talbot Algorithm

The fixed Talbot (FT) method involves deforming the Bromwich integral Eq. (7) so that the real part of p can take on negative values, making the integral converge more quickly. We can deform the contour (typically a vertical line at $s = \gamma$) into another open path that includes large, negative real values. By Cauchy's integral theorem ($\oint_B f(z)\,dz = 0$, provided no singularities of $f(z)$ are enclosed by the path B), this deformation is valid as long as the deformed path does not enclose any singularities. Talbot chose the fixed path

$$p(\theta) = \gamma\theta(\cot\theta + i), \qquad -\pi < \theta < +\pi \tag{11}$$

where γ a parameter (see Fig. 1).

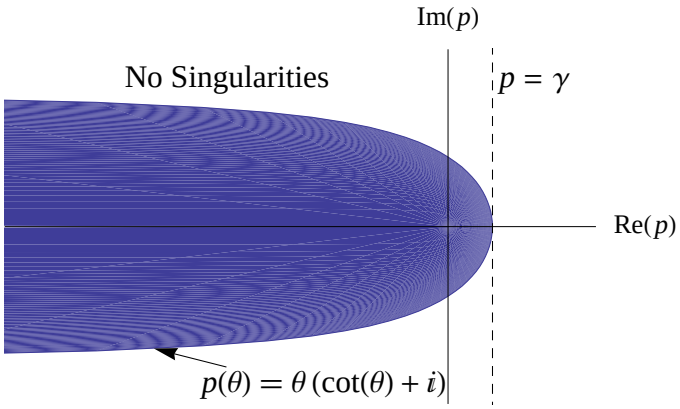

Figure 1: Plot of the standard path $p = \gamma$ for Eq. (10) and the Talbot path $p(\theta)$ given in Eq. (11). For the deformed path $p(\theta)$ to be valid, all the singularities of $\tilde{f}(p)$ must lie in the shaded region.

Then substituting the contour Eq. (11) into the inversion integral Eq. (10), we find

$$f(t) = \frac{\gamma}{\pi}\int_0^\pi \mathrm{Re}\left[e^{p(\theta)t}\tilde{f}(p(\theta))(1 + i\kappa(\theta)\right]\,d\theta \tag{12}$$

where

$$\kappa(\theta) = \theta + (\theta\cot\theta - 1)\cot\theta.$$

We discretize Eq. (12) using the trapezoidal rule with step size π/M and $\theta_k = k\pi/M$ to get

$$f(t, M) = \frac{\gamma}{M}\left\{\frac{1}{2}\tilde{f}\gamma\exp(\gamma t) + \sum_{k=1}^{M-1}\mathrm{Re}\left[\exp(tp(\theta_k))\tilde{f}(p(\theta_k))(1 + i\kappa(\theta_k))\right]\right\}. \tag{13}$$

Numerical experiments by Abate and Valkó fix the parameter γ to the value

$$\gamma = \frac{2M}{5t} \tag{14}$$

where M is the number of terms to sum and is the only free parameter. For this algorithm, the precision requirement is M digits of precision to control round-off error when computing Eq. (13). For a large class of transforms, the relative error estimate is

$$\left| \frac{f(t) - f(t, M)}{f(t)} \right| \approx 10^{-0.6M} \tag{15}$$

which gives about $0.6M$ significant digits in the approximant $f(t, M)$. The Mathematica implementation of the fixed Talbot algorithm is given in Appendix A.

III.B Gaver-Wynn-Rho Algorithm

The Gaver-Wynn-Rho (GWR) approach uses the Post-Widder formula which gives an analytic solution to the inverse problem in terms of high-order derivatives of $\tilde{f}(p)$

$$\psi_k(t) = \frac{(-1)^k}{k!} \left(\frac{k}{t}\right)^{k+1} \tilde{f}^{(k)} \left(\frac{k}{t}\right) \tag{16}$$

then as $\psi_k(t) \to f(t)$ as $k \to \infty$. Gaver presents a discrete analog of Eq. (16)

$$f_k(t) = \frac{(-1)^k \alpha k}{t} \left(\begin{array}{c} 2k \\ k \end{array} \right) \Delta^k \tilde{f}(k\alpha/t) \tag{17}$$

where $\alpha = \log(2)$ and Δ is the difference operator, $\Delta \tilde{f}(nx) = \tilde{f}((n+1)x) - \tilde{f}(nx)$.

Unfortunately, the Gaver functionals Eq. (17) only slowly converge to $f(t)$, in particular $|f(t) - f_k(t)| \sim 1/k$ as $k \to \infty$. Thus a convergence acceleration algorithm is required for the sequence to achieve a good approximation. In a study of algorithms, Abate and Valkó found that the best method is the Wynn rho algorithm given recursively as

$$\begin{aligned} &\rho_{-1}^{(n)} = 0, \qquad \rho_0^{(n)} = f_n(t), \qquad n \geq 0 \\ &\rho_k^{(n)} = \rho_{k-2}^{(n+1)} + \frac{k}{\rho_{k-1}^{(n+1)} - \rho_{k-1}^{(n)}}, \qquad k \geq 1. \end{aligned} \tag{18}$$

Then the approximant to $f(t)$ is

$$f(t, M) = \rho_M^{(0)} \tag{19}$$

where M is an even integer that governs the accuracy of the solution. The parameter M is also the number of Gaver functionals which must be computed from Eq. (17). One notable concern is round-off errors which might invalidate the results. To eliminate this problem, we can increase the computational precision used. It is found that the required precision is $2.1M$ and the relative error estimate is, for a wide range of transforms,

$$\left| \frac{f(t) - f(t, M)}{f(t)} \right| \approx 10^{-0.8M} \tag{20}$$

so the number of significant digits in $f(t, M)$ is about equal to M. The Mathematica implementation is given in Appendix B.

IV Results

We are now ready to invert the amplitudes Eq. (9*b*) to determine the behavior of the surface. Either of the inversion methods works and yields consistent results; however, for a desired accuracy the GWR method is slightly faster so it was used to prepare the plots shown. *Note*: that although some parameters are expressed below in decimal form, the algorithms require multi-precision numbers so the decimals were converted to fractions before being input to the algorithms.

We first examined the effect of viscosity on the lowest-degree amplitude $a_2(\tau)$ with no initial velocity. As shown in Fig. 2, increasing the viscosity ϵ causes the amplitude to decay more quickly.

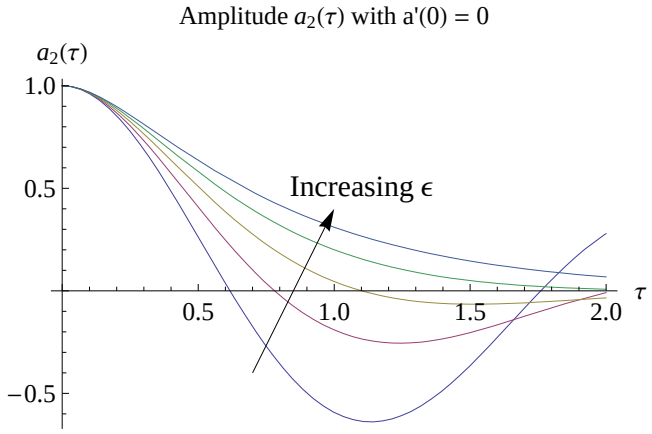

Figure 2: Plot of the amplitude $a_2(\tau)$ for values of the dimensionless viscosity ϵ from 0.1 to 0.9 in 0.2 increments with zero initial velocity $\dot{a}_2(0) = 0$.

We then investigated the effect of initial velocity on the lowest-degree amplitude $a_2(\tau)$ with a fixed viscosity $\epsilon = 0.3$. Figure 3 shows that increasing the initial velocity increases the maximum amplitude and thus increases the decay time.

Finally, we looked at different degree amplitudes $a_n(\tau)$ with fixed viscosity $\epsilon = 0.3$ and fixed initial velocity $u_0^* = 0.5$. We see in Fig. 4 that increasing the degree causes the amplitude to decay more rapidly.

V Conclusion

We have determined the shape of a free droplet for small oscillations that satisfies Eq. (5) using two numerical Laplace transform inversion schemes. Both methods yield consistent and reasonable results. One outstanding mystery is the comparison with Prosperetti's 1977 and 1980 articles [1] [5]. The plots for the amplitudes are qualitatively similar but the scale and effect of viscosity and initial velocity are somewhat different. The plots in Prosperetti's article show larger effects of ϵ and u_0^* than the ones presented here.

Further work might include investigating the other special case of bubbles or the more general case of two viscous fluids. Prosperetti's papers cover both cases. Another direction would be to explore other methods for the numerical inversion of the Laplace transform such as Fourier series or Laguerre polynomial decomposition.

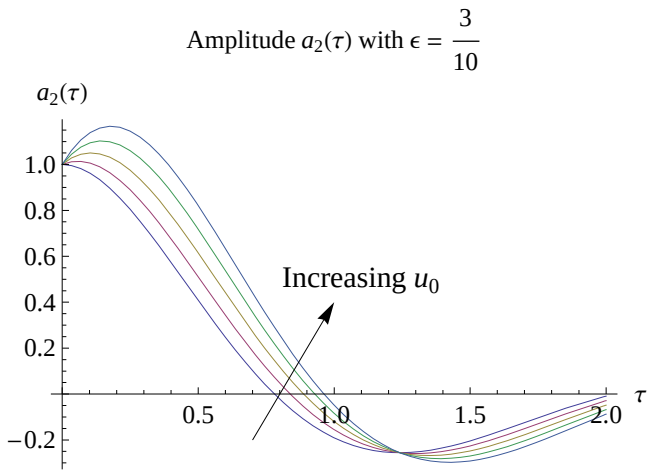

Figure 3: Plot of the amplitude $a_2(\tau)$ for values of the dimensionless initial velocity u_0^* from 0 to 2 in 0.5 increments.

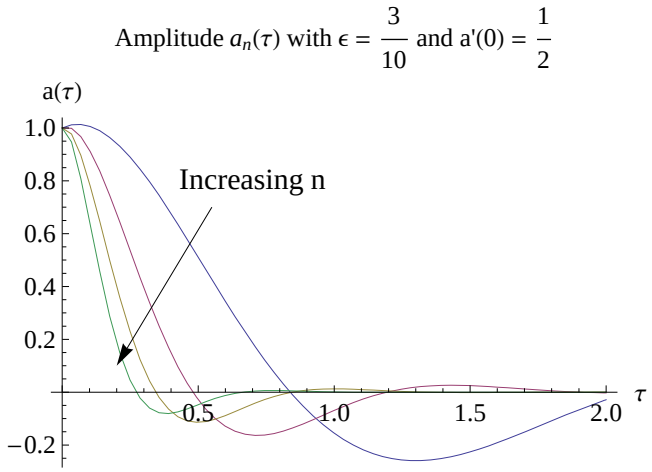

Figure 4: Plot of the amplitude $a_n(\tau)$ for values of n from 2 to 5 with $\epsilon = 0.3$ and $u_0^* = 0.5$. We see that the higher order amplitudes decay more quickly than the lower order ones.

VI Acknowledgements

I would like to thank Professor Yong for his tremendous help in finding and evaluating resources for this project.

Appendices

A FT algorithm code

```
FT[F_, t_, M_] :=
 Module[{np, r, S, theta, sigma}, np = Max[M, $MachinePrecision];
  r = SetPrecision[2 M/(5 t), np];
  S = r theta (Cot[theta] + I);
  sigma = theta + (theta Cot[theta] - 1) Cot[theta];
  (r/M) Plus @@
    Append[Table[
      Re[Exp[t S] (1 + I sigma) F[S]], {theta, Pi/M, (M - 1) Pi/M,
       Pi/M}], (1/2) Exp[r t] F[r]]]
```

B GWR algorithm code

```
GWR[F_, t_, M_: 32, precin_: 0] := Module[
      {M1, G0, Gm, Gp, best, expr, \[Tau] = Log[2]/t, Fi, broken,
   prec},
      If[precin <= 0, prec = 21 M/10, prec = precin];
      If[prec <= $MachinePrecision, prec = $MachinePrecision];
      broken = False;
      If[
   Precision[\[Tau]] < prec, \[Tau] = SetPrecision[\[Tau], prec]];
      Do[Fi[i] = N[F[i \[Tau]], prec], {i, 1, 2 M}];
      M1 = M;
      Do[
         G0[n - 1] = \[Tau] (2 n)!/(n! (n - 1)!) Sum[
                  Binomial[n, i] (-1)^i  Fi[n + i], {i, 0, n}];

   If[Not[NumberQ[G0[n - 1]]], M1 = n - 1; G0[n - 1] =.; Break[]];
         , {n, 1, M}];
      Do[Gm[n] = 0, {n, 0, M1}];
      best = G0[M1 - 1];

      Do[
         Do[
            expr = G0[n + 1] - G0[n];

    If[Or[Not[NumberQ[expr]], expr == 0], broken = True; Break[]];
            expr = Gm[n + 1] + (k + 1)/expr;
            Gp[n] = expr;
            If[OddQ[k],
               If[n == M1 - 2 - k, best = expr]
               ];
            , {n, M1 - 2 - k, 0, -1}];
```

```
    If[broken, Break[]];
    Do[Gm[n] = G0[n]; G0[n] = Gp[n], {n, 0, M1 - k}];
    , {k, 0, M1 - 2}];
best
]
```

References

[1] A. Prosperetti, *Viscous Effects on Perturbed Spherical Flows*, Quart. Appl. Math. **35**, 339-352, (1977).

[2] A. Frohn and N. Roth, *Dynamics of Droplets* (Springer, 2000), pp 245-260.

[3] M. S. Plesset, *On the stability of fluid flows with spherical symmetry*, J. Appl. Phys. **25**, 96-98 (1954).

[4] J. Abate and P. P. Valkó, *Multi-precision Laplace transform inversion*, Int. J. Numer. Meth. Engng **60**, 979-993 (2004).

[5] A. Prosperetti, *Free oscillations of drops and bubbles: the initial-value problem*, J. Fluid Mech. **100**, 333-347 (1980).