Available online at www.sciencedirect.com
ScienceDirect

Journal of Computational Physics 225 (2007) 408–426

JOURNAL OF COMPUTATIONAL PHYSICS

www.elsevier.com/locate/jcp

Interfacial dynamics in Stokes flow via a three-dimensional fully-implicit interfacial spectral boundary element algorithm

P. Dimitrakopoulos

Department of Chemical and Biomolecular Engineering, University of Maryland, College Park, MD 20742, USA

Received 10 August 2006; received in revised form 4 December 2006; accepted 8 December 2006
Available online 2 February 2007

Abstract

Since the pioneering work of Youngren and Acrivos [G.K. Youngren, A. Acrivos, On the shape of a gas bubble in a viscous extensional flow, J. Fluid Mech. 76 (1976) 433–442] 30 years ago, interfacial dynamics in Stokes flow has been implemented through explicit time integration of boundary integral schemes which require that the time step is sufficiently small to ensure numerical stability. To avoid this difficulty, we have developed an efficient, fully-implicit time-integration algorithm based on a mathematically rigorous combination of implicit formulas with a Jacobian-free three-dimensional Newton method. The resulting algorithm preserves the stability of the employed implicit formula and thus it has strong stability properties, e.g. it is not affected by the Courant condition or by physical stiffness such as that associated with the critical conditions of interfacial deformation. In our work, the numerical solution of our implicit algorithm is achieved through our spectral boundary element method. Our numerical results for free-suspended droplets are in excellent agreement with experimental findings, analytical predictions and earlier computational results at both subcritical and supercritical conditions, and establish the properties of our fully-implicit spectral boundary element algorithm.

Keywords: Implicit time integration; Spectral elements; Boundary integral equations; Stokes flow; Interfacial dynamics

---

1. Introduction

Interfacial dynamics in Stokes flow via the solution of boundary integral equations has developed considerably since the pioneering work of Youngren and Acrivos [37] 30 years ago. The main benefit of this approach is the reduction of the problem dimensionality by one, e.g. a fully three-dimensional problem can be described and solved by employing only two (curvilinear) coordinates. Extensive research has been done to determine and understand the deformation of droplets and bubbles in external flows, both in infinite media as well as in constrained geometries [26,30,32]. Considerable progress has also been made in the study of membrane-like interfaces such as those in artificial capsules and biological cells [26,27]. During the last few years the interaction of deformable interfaces, e.g. suspensions of droplets, has received a great deal of interest [39].

E-mail address: dimitrak@eng.umd.edu

0021-9991/$ - see front matter © 2006 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2006.12.004

To study these systems, several numerical methodologies have been developed which have one common characteristic, i.e. the fact that they employ explicit time-integration schemes for the determination of the interfacial deformation, e.g. see [9,19,29,38]. While these schemes are straightforward to employ, they require that, to ensure numerical stability, the employed time step must be smaller than any (numerical or physical) time scale appearing in the computational problem. The smallest time scale is commonly of numerical nature, associated with the numerical discretization owing to the well-known Courant condition [9,19,29,38,40]. (The exact form of this condition is given later in Section 3.)

This shortcoming of the explicit interfacial algorithms results in a large number of unnecessary time steps without any gain in accuracy. To avoid this, a typical implicit scheme based on the numerical calculation of the associated Jacobian matrix may allow for large time steps; however, its implementation to realistic problems is prohibited due to the large cost of determining numerically the expensive Jacobian matrix.

To overcome this difficulty, we have developed an efficient, fully implicit, time-integration algorithm for interfacial dynamics in Stokes flow. Our method is based on a mathematically rigorous combination of implicit schemes with our Jacobian-free three-dimensional Newton method [12], and thus it has strong stability properties which permits the utilization of very large time steps.

After the mathematical formulation of multiphase flows in the Stokes regime presented in Section 2, and a short review of the boundary integral solution via explicit time integration in Section 3, in Section 4 we present the mathematical formulation of our implicit interfacial method. Section 5 provides a summary of the spectral boundary discretization employed in our implicit method. In Section 6 we present the properties of the implicit interfacial method including comparisons with previous studies. Section 7 considers the time evolution of free-suspended droplets in subcritical and supercritical planar four-roll mill flows; as shown in this section, our numerical results are in excellent agreement with experimental findings [4,16], and establish the properties of our fully-implicit algorithm.

The present fully-implicit algorithm constitutes a new approach for interfacial dynamics in Stokes flow while it can be employed for a wide range of interfacial problems in porous media, microfluidic devices and physiological systems which are expected to continue having a significant growth in the coming years due to the increased interest in small scales [27,33].

2. Mathematical formulation

We consider a three-dimensional droplet suspended in an infinite fluid; the droplet (fluid 1) has density ρ_1 and viscosity $\lambda\mu$ while the surrounding fluid (fluid 2) has density ρ_2 and viscosity μ. The droplet size is specified by its volume V or equivalently by the radius a of a spherical droplet of volume $4\pi a^3/3 = V$. The gravitational acceleration is g while the surface tension γ is assumed constant. Far from the droplet, the flow approaches the undisturbed flow $\boldsymbol{u}^\infty$, e.g. a planar extensional flow $\boldsymbol{u}^\infty = G(x, -y, 0)$ or a simple shear flow $\boldsymbol{u}^\infty = G(z, 0, 0)$, where G is the shear rate. In this study, the characteristic length a is used as the length scale while the time is scaled with the flow time scale $\tau_\mathrm{f} = G^{-1}$.

The capillary number Ca and Bond number B_d are defined by

$$Ca = \frac{\mu G a}{\gamma}, \quad B_\mathrm{d} = \frac{(\rho_1 - \rho_2) g a^2}{\gamma} \tag{1}$$

These dimensionless parameters represent the ratio of viscous flow forces and gravitational forces to interfacial forces, respectively.

The governing equations in fluid 2 are the Stokes equations together with continuity

$$\nabla \cdot \boldsymbol{\sigma} = -\nabla p + \mu \nabla^2 \boldsymbol{u} = 0 \tag{2}$$

$$\nabla \cdot \boldsymbol{u} = 0 \tag{3}$$

while in the droplet, the same equations apply with the viscosity replaced by $\lambda\mu$.

At the interface, the boundary conditions on the velocity $\boldsymbol{u}$ and surface stress $\boldsymbol{f}$ are

$$\boldsymbol{u}_1 = \boldsymbol{u}_2 \tag{4}$$

$$\Delta \boldsymbol{f} = \boldsymbol{f}_2 - \boldsymbol{f}_1 = \gamma(\nabla \cdot \boldsymbol{n})\boldsymbol{n} + (\rho_2 - \rho_1)(\boldsymbol{g} \cdot \boldsymbol{x})\boldsymbol{n} \tag{5}$$

Here the subscripts designate quantities evaluated in fluids 1 and 2, respectively. The surface stress is defined as $\boldsymbol{f} = \boldsymbol{\sigma} \cdot \boldsymbol{n}$ where $\boldsymbol{n}$ is the unit normal which we choose to point into fluid 2. The pressure as defined in $\boldsymbol{\sigma}$ is the dynamic pressure; hence the gravity force is absent from Eq. (2) and appears in the interfacial stress boundary condition, Eq. (5).

The time evolution of the interfacial shape may be determined via the kinematic condition at the interface

$$\frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}t} = (\boldsymbol{u} \cdot \boldsymbol{n})\boldsymbol{n} \tag{6}$$

if the interfacial shape is known at some time instance.

Although the governing equations and boundary conditions are linear in $\boldsymbol{u}$ and $\boldsymbol{f}$, the problem of determining the dynamic droplet shape constitutes a non-linear problem, i.e. the velocity $\boldsymbol{u}$, stress $\boldsymbol{f}$ and curvature $\nabla \cdot \boldsymbol{n}$ are non-linear functions of the geometrical variables describing the interface shape.

3. Explicit time integration

Since the pioneering work of Acrivos and coworkers [28,37], a common way to solve the interfacial problem presented in Section 2, is by transforming the partial differential equations, Eqs. (2) and (3), into boundary integral equations, and utilizing an explicit time-integration scheme for the determination of the interfacial evolution. The main benefit of this transformation is the great reduction in computational time since a fully three-dimensional problem can be described and solved using only two (curvilinear) coordinates.

For a given droplet surface S_{B}, the velocity at a point $\boldsymbol{x}_0$ on the interface may be determined via the boundary integral equation [25]

$$(1+\lambda)\boldsymbol{u}(\boldsymbol{x}_0) - 2\boldsymbol{u}^{\infty}(\boldsymbol{x}_0) = -\frac{1}{4\pi\mu}\int_{S_{\mathrm{B}}} [\boldsymbol{S} \cdot \Delta\boldsymbol{f} - (1-\lambda)\mu\boldsymbol{T} \cdot \boldsymbol{u} \cdot \boldsymbol{n}](\boldsymbol{x})\,\mathrm{d}S \tag{7}$$

where S_{ij} is the fundamental solution for the three-dimensional Stokes equations and T_{ijk} the associated stress defined by

$$S_{ij} = \frac{\delta_{ij}}{r} + \frac{\hat{x}_i\hat{x}_j}{r^3}, \quad T_{ijk} = -6\frac{\hat{x}_i\hat{x}_j\hat{x}_k}{r^5} \tag{8}$$

where $\hat{\boldsymbol{x}} = \boldsymbol{x} - \boldsymbol{x}_0$ and $r = |\hat{\boldsymbol{x}}|$ [25]. Similar equations hold in the presence of solid boundaries and for drop/cell suspensions [12,39].

In order to determine the droplet shape as a function of time, an explicit time-integration scheme may be employed to solve the kinematic condition at the interface, Eq. (6), e.g. [3,19,20,38]. However, the employed time step Δt should be sufficiently small to ensure numerical stability; in particular Δt must be smaller than any (numerical or physical) time scale appearing in the computational problem. The strictest requirement is commonly of numerical nature (i.e. associated with the numerical grid), the well-known Courant condition which in dimensionless form may be written as

$$\Delta \boldsymbol{t} < \mathrm{O}(Ca\Delta x_{\min}) \tag{9}$$

where $\Delta x_{\min}$ is the minimum length scale appearing in the computational problem, e.g. the minimum grid spacing or the distance between interfaces in close contact [9,19,29,38,40]. (For gravity-only induced deformation, the time step should be reduced as the Bond number decreases [11].) This shortcoming of the explicit interfacial algorithms results in a large number of unnecessary time steps without any gain in accuracy.

4. Implicit interfacial method

To overcome the limitations imposed by the Courant condition, we have developed an efficient, fully implicit, time-integration algorithm for interfacial dynamics in Stokes flow. Our method is based on a mathematically rigorous combination of implicit schemes with our Jacobian-free three-dimensional Newton method [12], and thus it has strong stability properties which permits the utilization of very large time steps.

The interfacial evolution is determined by the time integration of the kinematic condition at the interface $\boldsymbol{x}$ used in the general form

$$\frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}t} = \boldsymbol{a} \equiv (\boldsymbol{u} \cdot \boldsymbol{n})\boldsymbol{n} + U_{\mathrm{t}}\boldsymbol{t} \tag{10}$$

where the first term of the surface derivative $\boldsymbol{a}$ denotes the contribution of the normal interfacial velocity $(\boldsymbol{u} \cdot \boldsymbol{n})$ while the second term denotes the contribution of some velocity U_{t} tangential to the interface.

The basic integration scheme is the A-stable, first-order, implicit Euler scheme

$$\boldsymbol{x}(t + \Delta t) = \boldsymbol{x}(t) + \Delta t\boldsymbol{a}(t + \Delta t) \tag{11}$$

which relates the unknown interfacial shape at time $(t + \Delta t)$ with the known shape at time t and the surface derivative $\boldsymbol{a}(t + \Delta t)$ of the unknown interface. The backward differentiation formulas shown in Table 1 utilize the known interfacial shapes from previous times to achieve higher-order accuracy. These multi-step schemes may be written in the form

$$\boldsymbol{x}(t + \Delta t) = \boldsymbol{x}(t) + \kappa\Delta t\boldsymbol{\Pi} + \kappa\Delta t\boldsymbol{a}(t + \Delta t) \tag{12}$$

where κ is a known scalar coefficient for each method while the vector $\boldsymbol{\Pi}$ describes a known quantity from previous time steps. We note that, these schemes have very good stability properties, even though they are not A-stable [6].

To derive an efficient, fully-implicit, time-integration algorithm we combine the implicit formula with our Newton method for interfacial dynamics in Stokes flow [12]. We emphasize that this way the resulting algorithm *preserves the stability properties* of the corresponding implicit formula, e.g. Euler or backward differentiation method [6,13,17]. Our implicit time-integration algorithm is based on a Jacobian-free integration of the kinematic condition at the interface, Eq. (10), by employing boundary perturbations involving the unknown shape $\boldsymbol{x}(t + \Delta t)$. The analysis presented below is similar to that employed for the determination of equilibrium interfaces in Stokes flow via our Newton method; the interested reader is referred to our earlier publication for more details on the Newton method [12].

During the k iteration of the Newton method, let $d^k(\xi, \eta)$ be the point-by-point displacement, along a prescribed direction $\boldsymbol{p}(\xi, \eta)$, between the unknown interface $\boldsymbol{x}^k(t + \Delta t)$ and the (now known) interface $\boldsymbol{x}^{k-1}(t + \Delta t)$ of the previous Newton iteration, i.e.

$$\boldsymbol{x}^k(t + \Delta t) = \boldsymbol{x}^{k-1}(t + \Delta t) + d^k\boldsymbol{p} = \boldsymbol{x}(t) + \left(\sum_{j=1}^{k-1} d^j\right)\boldsymbol{p} + d^k\boldsymbol{p} \tag{13}$$

where we use the known interface $\boldsymbol{x}(t)$ at time t as the initial estimate of the Newton iteration for the time $(t + \Delta t)$.

All geometric and physical variables on $\boldsymbol{x}^k(t + \Delta t)$ may be expressed as functions of the unknown displacement $d^k(\xi, \eta)$ via boundary perturbations involving the two interfacial shapes above. For example, the normal vector and curvature may be written as

$$\boldsymbol{n}^k = \boldsymbol{n}^{k-1} + \mathscr{N}(d) + \mathrm{O}(d^2) \tag{14}$$

$$(\boldsymbol{\nabla} \cdot \boldsymbol{n})^k = (\boldsymbol{\nabla} \cdot \boldsymbol{n})^{k-1} + \mathscr{K}(d) + \mathrm{O}(d^2) \tag{15}$$

while the interfacial velocities and forces of the fluid inside the interface $(i = 1)$ or the surrounding fluid $(i = 2)$ may be written as

$$\boldsymbol{u}_i^k = \boldsymbol{u}_i + \mathscr{U}_i(d) + \mathrm{O}(d^2) \tag{16}$$

$$\boldsymbol{f}_i^k = \boldsymbol{f}_i + \mathscr{F}_i(d) + \mathrm{O}(d^2) \tag{17}$$

Table 1
Backward differentiation (BDF) methods of second to fourth order

$\boldsymbol{x}_{n+1} = \frac{1}{3}(-\boldsymbol{x}_{n-1} + 4\boldsymbol{x}_n + 2\Delta t\boldsymbol{a}_{n+1})$
$\boldsymbol{x}_{n+1} = \frac{1}{11}(2\boldsymbol{x}_{n-2} - 9\boldsymbol{x}_{n-1} + 18\boldsymbol{x}_n + 6\Delta t\boldsymbol{a}_{n+1})$
$\boldsymbol{x}_{n+1} = \frac{1}{25}(-3\boldsymbol{x}_{n-3} + 16\boldsymbol{x}_{n-2} - 36\boldsymbol{x}_{n-1} + 48\boldsymbol{x}_n + 12\Delta t\boldsymbol{a}_{n+1})$

The subscripts $(n + l)$, where $l = 1, 0, -1, \ldots$, denote quantities evaluated at times $(t + l\Delta t)$ [6].

The associated linear functions of displacement, i.e. $\mathscr{N}$, $\mathscr{K}$, $\mathscr{U}_i$ and $\mathscr{F}_i$, are given in Ref. [12]. We emphasize that in Eqs. (16) and (17), $\boldsymbol{u}_{\mathrm{i}}$ and $\boldsymbol{f}_{\mathrm{i}}$ are the velocities and forces on the known surface $\boldsymbol{x}^{k-1}(t+\Delta t)$ when the interface is $\boldsymbol{x}^k(t+\Delta t)$. The flow over the interface $\boldsymbol{x}^k(t+\Delta t)$ may be described as an integral over the known surface $\boldsymbol{x}^{k-1}(t+\Delta t)$,

$$(\boldsymbol{u}_2+\lambda\boldsymbol{u}_1-2\boldsymbol{u}^\infty)(\boldsymbol{x}_0) = -\frac{1}{4\pi\mu}\int_{\boldsymbol{x}^{k-1}}[\boldsymbol{S}\cdot(\boldsymbol{f}_2-\boldsymbol{f}_1)-\mu\boldsymbol{T}\cdot(\boldsymbol{u}_2-\lambda\boldsymbol{u}_1)\cdot\boldsymbol{n}](\boldsymbol{x})\,\mathrm{d}S \qquad (18)$$

Similar generalized boundary integral equations hold in the presence of solid boundaries and for drop/cell suspensions in all possible types of domains, i.e. infinite, semi-infinite or closed systems [12].

By combining the boundary conditions on the interface $\boldsymbol{x}^k(t+\Delta t)$, i.e. equal velocities $\boldsymbol{u}_1^k=\boldsymbol{u}_2^k$ and known force jump $(\boldsymbol{f}_2^k-\boldsymbol{f}_1^k)$, with the perturbation equations (16) and (17) above, the generalized boundary integral equation, Eq. (18), may be solved for the unknown displacement $d^k(\xi,\eta)$ if a condition for the normal velocity of the unknown interface $\boldsymbol{x}^k(t+\Delta t)$ is given. For equilibrium interfaces this condition is the requirement of zero normal interfacial velocity, $(\boldsymbol{u}\cdot\boldsymbol{n})^k(t+\Delta t)=0$ [12].

For transient dynamics, the condition for the normal velocity of the unknown interface $\boldsymbol{x}^k(t+\Delta t)$ is derived by employing the implicit time-integration scheme, i.e. by combining Eqs. (13) and (12) we obtain

$$(\boldsymbol{u}\cdot\boldsymbol{n})^k = \left[\sum_{j=1}^{k-1} d^j + d^k\right]\frac{\boldsymbol{p}\cdot\boldsymbol{n}^k}{\kappa\Delta t} - \boldsymbol{\Pi}\cdot\boldsymbol{n}^k \qquad (19)$$

Observe that the kinematic condition above is independent of the tangential velocity U_{t} appearing in the surface derivative $\boldsymbol{a}$ (see Eq. (10) above). However, U_{t} is utilized in the determination of the interfacial shape, since during the Newton iteration the points $\boldsymbol{x}(\xi,\eta)$ are advanced along the prescribed direction $\boldsymbol{p}(\xi,\eta)$ which in general is different from the normal direction of the anticipated shape $\boldsymbol{x}(t+\Delta t)$ (see also Eq. (13)).

Having developed an efficient fully-implicit dynamics algorithm involving multi-step (one-stage) implicit formulas (e.g. Euler and backward differentiation schemes), it is straightforward to extend it to sequential multi-stage implicit formulas such as the diagonally implicit Runge–Kutta schemes (DIRK) shown in Table 2. We emphasize that each stage of these schemes is independent of the following ones, and thus they are evaluated sequentially (as it happens with the corresponding explicit Runge–Kutta schemes). These high-order implicit schemes have excellent stability properties being at least A-stable; in particular, the second- and third-order DIRK schemes are L-stable while the fourth-order DIRK is A-stable [2,6].

Our implicit algorithm requires 2–3 Newton iterations for each time step to achieve suitable convergence; each Newton iteration requires a maximum of three boundary integral solutions of systems with the same size as the standard explicit algorithms (see p. 197 in Ref. [12]). Therefore, one implicit time step is less than 10 times more expensive than the corresponding explicit time step. This difference is easily covered by the ability to use much larger time steps, even of O(1), as we discuss in Section 6.

Table 2
Diagonally implicit Runge–Kutta (DIRK) methods of second to fourth order

Second-order diagonally implicit Runge–Kutta (DIRK2)

$\boldsymbol{x}_{s1}=\boldsymbol{x}_n+\Delta t a_0\boldsymbol{a}(\boldsymbol{x}_{s1})$

$\boldsymbol{x}_{n+1}=\boldsymbol{x}_n+\Delta t[(1-a_0)\boldsymbol{a}(\boldsymbol{x}_{s1})+a_0\boldsymbol{a}(\boldsymbol{x}_{n+1})]$

Third-order diagonally implicit Runge–Kutta (DIRK3)

$\boldsymbol{x}_{s1}=\boldsymbol{x}_n+\Delta t b_0\boldsymbol{a}(\boldsymbol{x}_{s1})$

$\boldsymbol{x}_{s2}=\boldsymbol{x}_n+\Delta t[0.5(1-b_0)\boldsymbol{a}(\boldsymbol{x}_{s1})+b_0\boldsymbol{a}(\boldsymbol{x}_{s2})]$

$\boldsymbol{x}_{n+1}=\boldsymbol{x}_n+\Delta t[b_1\boldsymbol{a}(\boldsymbol{x}_{s1})+b_2\boldsymbol{a}(\boldsymbol{x}_{s2})+b_0\boldsymbol{a}(\boldsymbol{x}_{n+1})]$

Fourth-order diagonally implicit Runge–Kutta (DIRK4)

$\boldsymbol{x}_{s1}=\boldsymbol{x}_n+\Delta t 0.5(1+c_0)\boldsymbol{a}(\boldsymbol{x}_{s1})$

$\boldsymbol{x}_{s2}=\boldsymbol{x}_n+\Delta t[-0.5c_0\boldsymbol{a}(\boldsymbol{x}_{s1})+0.5(1+c_0)\boldsymbol{a}(\boldsymbol{x}_{s2})]$

$\boldsymbol{x}_{s3}=\boldsymbol{x}_n+\Delta t[(1+c_0)\boldsymbol{a}(\boldsymbol{x}_{s1})-(1+2c_0)\boldsymbol{a}(\boldsymbol{x}_{s2})+0.5(1+c_0)\boldsymbol{a}(\boldsymbol{x}_{s3})]$

$\boldsymbol{x}_{n+1}=\boldsymbol{x}_n+\Delta t/(6c_0^2)[\boldsymbol{a}(\boldsymbol{x}_{s1})+(6c_0^2-2)\boldsymbol{a}(\boldsymbol{x}_{s2})+\boldsymbol{a}(\boldsymbol{x}_{s3})]$

Note that $\boldsymbol{x}_n\equiv\boldsymbol{x}(t)$, $\boldsymbol{x}_{n+1}\equiv\boldsymbol{x}(t+\Delta t)$ while $\boldsymbol{x}_{s1}$, $\boldsymbol{x}_{s2}$ and $\boldsymbol{x}_{s3}$ denote intermediate stages. For DIRK2, $a_0=1-\sqrt{2}/2$. For DIRK3, $b_0=0.43586652150846$ is the root of the equation $6b_0^3-18b_0^2+9b_0-1=0$ in $[1/6,1/2]$ while $b_1=(-6b_0^2+16b_0-1)/4$ and $b_2=(6b_0^2-20b_0+5)/4$. For DIRK4, $c_0=2\cos(\pi/18)/\sqrt{3}$. [2,6].

5. Spectral boundary discretization

The fully-implicit time integration for interfacial dynamics in Stokes flow described in Section 4, may be combined with any existing boundary integral algorithm. In our work, the numerical solution of our implicit algorithm is achieved through an extension of the spectral boundary element method described in [12,23]. The initial interface is divided into a moderate number $N_{\rm E}$ of elements as shown in Fig. 1. The geometric variables on each element are discretized using Lagrangian interpolation in terms of parametric variables ξ, η on the square interval $[-1,1]^2$, e.g.

$$\boldsymbol{x}(\xi,\eta)=\sum_{i=1}^{N_{\rm B}}\sum_{j=1}^{N_{\rm B}}\boldsymbol{x}(\xi_i,\eta_j)h_i(\xi)h_j(\eta) \tag{20}$$

where h_i is the $(N_{\rm B}-1)$-order Lagrangian interpolant polynomial. The physical variables $\boldsymbol{u}$ and $\boldsymbol{f}$ are represented similarly. The base points (ξ_i,η_j) for the interpolation are chosen as the zeros of $N_{\rm B}$-order orthogonal polynomials; this yields the spectral convergence associated with the orthogonal polynomial expansion.

The boundary integral equations, e.g. Eqs. (7) and (18), admit two different types of points. The collocation points $\boldsymbol{x}_0$ of the left-hand side where the equation is required to hold and the basis points $\boldsymbol{x}$ of the right-hand side where the physical variables $\boldsymbol{u}$ and $\boldsymbol{f}$ are defined. The spectral element method as implemented here employs collocation points of Gauss quadrature, i.e. in the interior of the element. As a result the boundary integral equation holds even for singular elements where the normal vector is not uniquely defined.

In addition, we use basis points of Gauss–Lobatto quadrature [7]. Owing to this choice, the position is always continuous on the interface. (Note that our implicit algorithm determines the shape $\boldsymbol{x}(t+\Delta t)$ at the basis points.) Additional constraints are required at the ends of the elements to enforce continuity of the normal vector and curvature on the interface. To achieve this, we require continuity of the term $(\boldsymbol{\nabla}\cdot\boldsymbol{n})\boldsymbol{n}$ at the edges of the spectral elements; these non-linear constraints are embodied into the interfacial system via linearization of the associated terms (see Eqs. (14) and (15) above). As the Newton iteration converges, the interfacial shape satisfies the geometrical conditions to any desired numerical accuracy, e.g. the machine precision.

The Gauss-type points may be derived from the Jacobi polynomials $P_N^{\alpha,\beta}$ where the parametric constants α and β are greater than -1 [1,7]. Different values of α and β yield diverse orthogonal polynomials, e.g. $\alpha=\beta=0$ yields the Legendre polynomials while $\alpha=\beta=-1/2$ produces the Chebyshev polynomials. Similar to the interior Gauss points, Gauss–Lobatto points (which include the end points ± 1) can also be derived from the Jacobi polynomials; the distribution of the interior points is again controlled by the associated parameters α and β. Note that equal values of the two parameters, α and β, result in points symmetric around zero in the interval $[-1,1]$. As the value of $\alpha=\beta$ approaches -1, the roots are closer to the ends of the interval; larger values produce roots closer to the center of the interval $[-1,1]$.

The discretized expressions for the geometry and the physical variables are substituted into the boundary integral equations yielding a linear system of algebraic equations $\boldsymbol{u}=\boldsymbol{A}\boldsymbol{f}+\boldsymbol{B}\boldsymbol{u}$. The system matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ are defined as integrals of the kernels $\boldsymbol{S}$ and $\boldsymbol{T}$ and the basis functions over the set of the surface elements. The

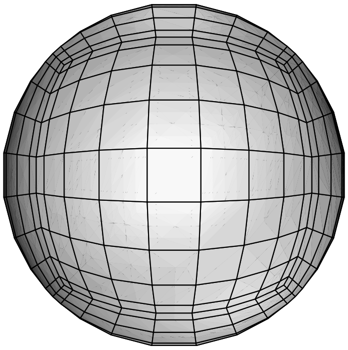

Fig. 1. Spectral boundary element discretization of a spherical droplet into $N_{\rm E}=6$ elements. The figure illustrates Gauss–Lobatto Legendre distribution of nodal lines with $N_{\rm B}=10$ spectral points in each direction.

numerical integration is performed by Gauss–Legendre quadrature with the aid of variable transformations. Owing to the singularity in the kernels, special care must be taken to ensure the accurate numerical evaluation of these integrations as described in [23]. Due to their small size for the problems studied in this paper, the discretized systems of equations are solved via direct solvers from the LAPACK software library.

Our numerical code has the ability to exploit the symmetry planes of a specific interfacial problem. Exploiting m symmetry levels reduces the memory requirements by a factor of 4^m, the computational time for determining the system matrices by a factor of 2^m, and the solution time via direct system solvers by factor of 8^m. In addition, our numerical code has been parallelized on shared-memory multiprocessor computers (such as the SGI Origin 2000 and IBM pSeries 690) by employing OpenMP directives for the calculation of the system matrix, and highly optimized parallel routines from the LAPACK system library for the solution of the dense system matrix. Multiprocessor runs exploit the parallel nature of calculating the system matrices by the boundary integral equations, e.g. Eqs. (7) and (18), resulting in an overall very good parallel efficiency as we discuss in Ref. [35].

6. Properties of implicit spectral interfacial method

To verify the accuracy and the (expected strong) stability properties of our implicit algorithm, we have conducted extensive tests for droplets and bubbles suspended freely in linear flows. To determine the droplet deformation with time we monitor the droplet's longest and shortest semi-axes, L and S, respectively, as well as Taylor's deformation parameter

$$D = \frac{L-S}{L+S} \tag{21}$$

These semi-axes are determined as the maximum and minimum distance from the droplet's centroid to the interface by employing a Newton method for the optimization problems. We also calculate the drop's length L_i, width S_i and depth W_i as the semi-axes of the ellipsoid which has the same inertia tensor as the droplet. In addition, we monitor the orientation angle of the drop θ_x defined as the angle between the longest semi-axis L and the flow direction, i.e. the x-axis.

Our results are in very good agreement with experimental findings of Bentley and Leal [4], and of Ha and Leal [16], for the subcritical and supercritical deformation of droplets in planar four-roll mill flows as discussed in more detail in Section 7. They also show very good agreement with the experimental findings for the transient evolution in simple shear flow reported in Figs. 5 and 6 of Ref. [15], and are in excellent agreement with the corresponding computations by Pozrikidis included in Ref. [15].

Our computations are in excellent agreement with the analytical predictions of Wetzel and Tucker [36] for the transient evolution of droplets with zero interfacial tension in shear and planar extensional flows reported in Fig. 1 of the analytical study. In particular, we considered the evolution of a droplet with $\lambda = 3$ over the time interval $[0, 5]$ in simple shear flow, and a drop with $\lambda = 18.6$ over the time interval $[0, 6]$ in planar extensional flow, as in Fig. 1 of Ref. [36]. By employing $\Delta t = 0.1$ and our adaptive mesh reconstruction (described in the next section), our numerical results for the droplet's length, width and depth agree over at least four significant digits with those of the analytical study.

Our algorithm, based on the backward differentiation and diagonally implicit Runge–Kutta schemes, has very strong stability properties even for $\Delta t = \mathrm{O}(1)$. As test problems, we have monitored the deformation of initially spherical droplets with very dense grids under shear and extensional flows and for several capillary numbers and viscosity ratios. (In particular, we discretized the initial spherical droplets into $N_\mathrm{E} = 54, 66$ spectral elements and utilized $N_\mathrm{B} = 9$–12 basis points, resulting in a total number of spectral points $N = N_\mathrm{E} N_\mathrm{B}^2$ in the range 4374–9504). For both types of implicit schemes, the required time step Δt is *independent* of the grid spacing Δx and the capillary number Ca. Most importantly, the same is true when we define droplets with regular grids (i.e. $N_\mathrm{E} \geqslant 6$) which are employed for the study of interfacial deformation.

We emphasize that we can easily achieve sufficient accuracy even with large time steps by employing high-order formulas such as the third- and fourth-order backward differentiation and diagonally implicit Runge–Kutta schemes. As test problems, we have determined the evolution of the interfacial shape for various viscosity ratios λ and capillary numbers Ca, and for different time steps, usually $\Delta t = 10^{-3}, 10^{-2}, 10^{-1}$. In all cases, the high-order schemes mentioned above were able to produce a relative error $\leqslant 10^{-3}$ at all times. (The accuracy was verified by employing both our explicit and implicit algorithms.)

These test problems show that our implicit algorithm removes the penalty of small time steps associated with the Courant condition, Eq. (9), as expected due to the rigorous combination of the implicit formula with our Newton method.

Our implicit algorithm may also be employed to study physically stiff interfacial problems. We note that a stiff problem may be understood as the one involving two well separated time scales. To avoid numerical instability, any explicit algorithm needs to resolve successfully the small time scale (by employing small enough time steps) over the long period defined by the large time scale of the stiff problem. This results in a very large number of unnecessary time steps without any gain in accuracy [6,17]. On the other hand, all stiff interfacial problems are computationally simple for our algorithm owing to its (fully) implicit nature.

To show this issue, we have studied the dynamics of drop deformation near the critical conditions, i.e. near the flow rate at which equilibrium interfacial shapes cease to exist. As discussed in Refs. [5,24], near the critical point the drop evolves on two distinct time scales: a short one due to the surface tension relaxation and a long one which diverges to infinity at the critical point. In contrast to explicit algorithms where a small time step and a large number of steps are required to study this problem, our implicit algorithm produces stable and accurate results for the large time step of $\Delta t = 0.1$ as shown in Fig. 2a. The time evolution of the drop deformation D is shown in Fig. 2b for a subcritical and a supercritical capillary number. Both curves were produced by employing a time step $\Delta t = 0.1$ while we also include our results for $\Delta t = 1$. We note that even the large time step $\Delta t = 1$ results in a numerically stable (and quite accurate) evolution for both subcritical and supercritical conditions. (Note that for subcritical flow rates, after equilibrium is reached, the numerical accuracy does not depend on the time step Δt.)

An additional benefit of our implicit method is the fact that due to the linearization inherent in the involved Newton method, any non-linear boundary condition can be easily embodied into our algorithm and solved directly along with the interfacial evolution without the need for a post-solution treatment/correction. As an example, we refer to the non-linear relation between the contact line speed and the dynamic contact angle for the case of fluid volumes sliding on solid substrates [8,14,22].

The discussion so far in this section describes the properties of our fully-implicit algorithm independently of the boundary element method employed for its solution. The specific solution of our implicit algorithm utilized in our work is based on our spectral boundary element method [12,23] and thus it exploits all the benefits of the spectral methods, i.e. high-order interpolation with exponential convergence and numerical stability with increasing number of spectral points [18,21], along with the versatility of the boundary element method, i.e. the ability to handle the most complicated geometries. In addition, it is not affected by the disadvantage of the spectral methods used in volume discretization; namely, the requirement to deal with dense systems, because in boundary integral formulations the resulting systems are always dense, independent of the form of the discretization.

The exponential convergence in the numerical accuracy as the number of the employed spectral points $N = N_E N_B^2$ increases is clearly evident at the geometric properties of a given shape such as the interfacial curvature shown in Fig. 3. The exponential convergence of our spectral algorithm is in direct contrast to the common linear or quadratic convergence associated with low-order algorithms, e.g. [3,9,19,38].

The exponential convergence in the numerical accuracy is also evident for the dynamic evolution of the interfacial shape as shown in Fig. 4. We emphasize that the difference in the interfacial accuracy between our spectral algorithm and low-order methods is dramatic since the latter commonly achieve linear convergence by increasing the number of the employed grid points [9,20]. Therefore, to achieve a desired accuracy, our high-order method may not require as dense a numerical grid as low-order boundary element methods.

As a closure to this section, we note that one exception in our discussion of the limitations of the explicit boundary integral methods (presented earlier in Section 3) is the recent algorithm of Bazhlekov et al. [3]. By recognizing that the numerical instability associated with the Courant condition in a grid point $\boldsymbol{x}$ results mainly from the time discretization of the force jump, Eq. (5), in the grid points near $\boldsymbol{x}$, the authors implemented a multiple step time-integration scheme. In particular, the force jump was updated in accordance with the limitations imposed by the Courant condition while the interfacial velocity, and thus the interfacial shape, was updated in larger time intervals Δt (see Section III.E of the earlier study [3]). According to Ref. [3] the multiple step time-integration scheme permits utilizations of time steps 10 times larger than the stability limit for accurate calculations.

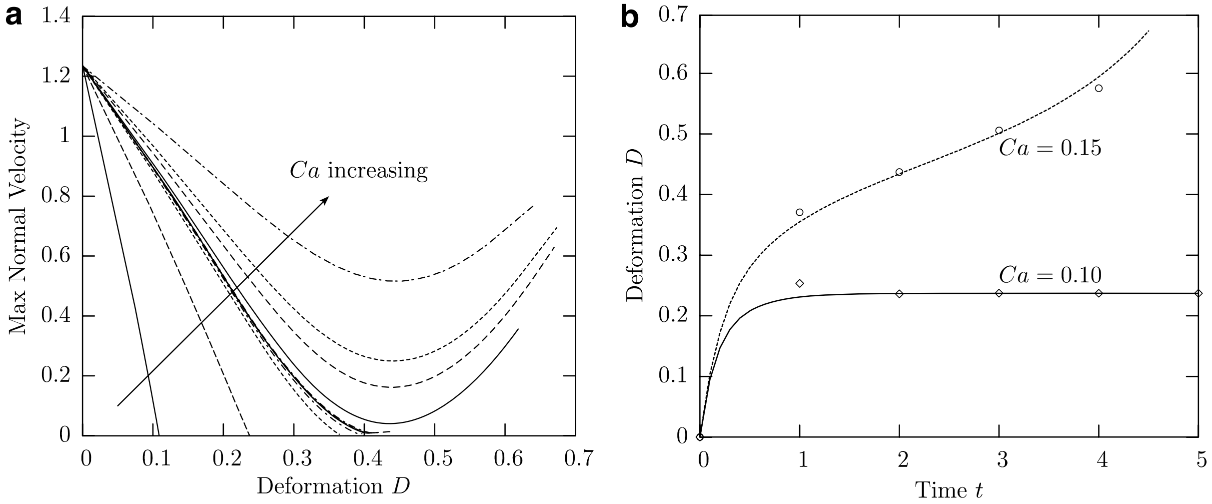

Fig. 2. (a) Dynamics near the critical point: maximum normal velocity versus deformation D for a droplet with viscosity ratio $\lambda = 0.5$ in a planar extensional flow $\boldsymbol{u}^{\infty} = G(x, -y, 0)$. The capillary number is $Ca = 0.05, 0.1, 0.13, 0.133, 0.134, 0.1341, 0.1342, 0.1345, 0.138, 0.15, 0.16, 0.2$. These curves were generated by employing the third-order diagonally implicit Runge–Kutta scheme (DIRK3) with $\Delta t = 0.1$. The accuracy was verified by employing smaller time steps with both our explicit and implicit algorithms. The critical capillary number (i.e. $Ca \approx 0.1342$) is in excellent agreement with experimental findings [4]. (b) Time evolution of the droplet deformation D for the subcritical $Ca = 0.1$ and the supercritical $Ca = 0.15$. Both curves were produced by employing DIRK3 with $\Delta t = 0.1$; also included as diamond and circular points the corresponding results for $\Delta t = 1$.

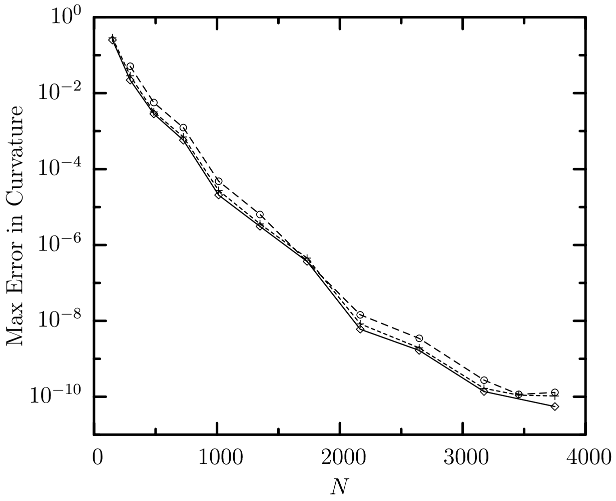

Fig. 3. The maximum absolute error in the computed curvature versus the number of spectral points $N = N_{\mathrm{E}} N_{\mathrm{B}}^{2}$ for different spheroids: ——, $a = b = c = 1$; - - - -, $a = b = 1, c = 0.4$; — — —, $a = 1, b = c = 0.4$. (a, b and c are the spheroid semi-axes). The exact value was used to determine the numerical error.

Thus, the algorithm of Bazhlekov et al. [3] seems to do a good work with respect to the limitations on the employed time step imposed by the Courant condition. However, it is unclear if that algorithm is completely unaffected by the Courant condition, i.e. if the employed time step is independent of the grid spacing Δx and the capillary number Ca as it happens with our methodology due to its rigorous combination of implicit formulas with our Newton method. In addition, we note that the algorithm of Bazhlekov, Anderson and Meijer is still affected by small physical time scales (such as those near critical conditions) and does not have the advantage of easy incorporation of non-linear boundary conditions.

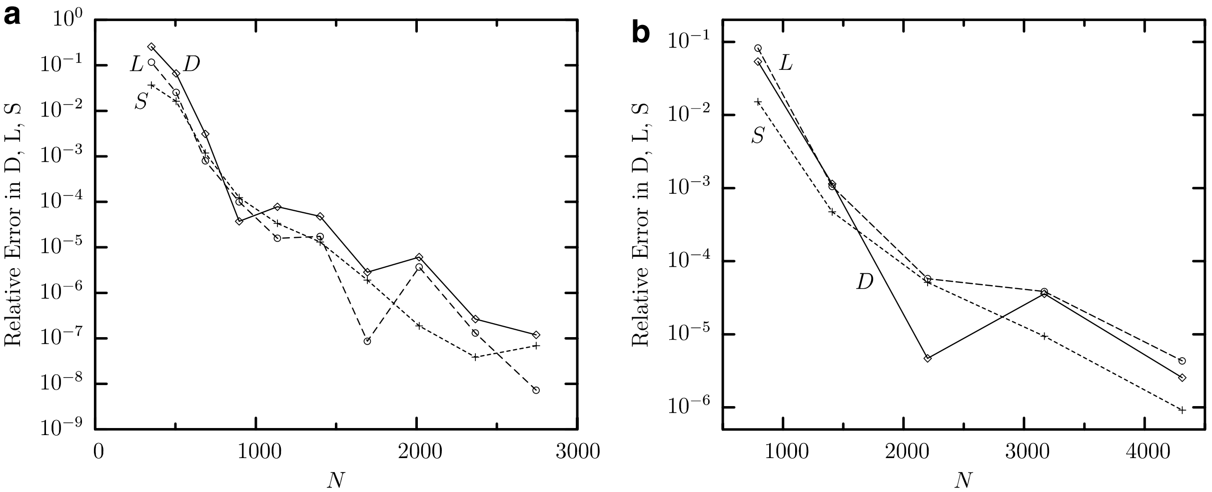

Fig. 4. Relative error in the computed deformation D, drop length L and width S versus the number of spectral points $N = N_E N_B^2$. (a) $Ca = 0.155$, $\lambda = 0.2$, at time $t = 0.5$ in a planar extensional flow. (Note that $D = 0.292$ while the droplet length has been increased by 38% and its width has been decreased by 24%.) The exponential convergence shown was generated by employing $N_E = 14$ spectral elements and varying the number of basis points N_B from 5 to 14 with $\Delta t = 0.01$. (b) $Ca = 0.25$, $\lambda = 0.056$, at time $t = 1$ in a planar extensional flow. (Note that $D = 0.5$ while the droplet length has been increased by 86% and its width has been decreased by 39%.) The exponential convergence shown was generated by employing $N_E = 22$ spectral elements and varying the number of basis points N_B from 6 to 14 with $\Delta t = 0.01$. (In both cases, the results for $N_B = 15$ were used to determine the numerical error.)

7. Dynamics of free-suspended droplets

In this section we consider the dynamics of free-suspended droplets at both subcritical and supercritical flow rates. The main purpose of this section is to establish the ability of our implicit algorithm to determine accurately the interfacial shape by utilizing large time steps independent of the Courant condition, Eq. (9). We also show the ability of our spectral boundary element method to accurately determine complicated interfacial morphologies including elongated shapes with pointed edges or interfacial necks.

7.1. Subcritical droplet evolution

In this subsection, we study the droplet evolution at subcritical capillary numbers where the drop reaches equilibrium after an initial transient evolution. In particular, we investigate the time evolution of the droplet length L, width S and deformation D for different viscosity ratios including low- and high-viscosity droplets. We also compare the equilibrium shapes from our numerical results with the experimental findings of Bentley and Leal [4]. To do this, we study droplet deformation in planar four-roll mill flows $\boldsymbol{u}^{\infty} = (u_x, u_y, 0)$ with

$$u_x = \frac{G}{2}[(1 + a_f)x + (1 - a_f)y]$$
$$u_y = \frac{G}{2}[(-1 + a_f)x - (1 + a_f)y] \tag{22}$$

for several viscosity ratios and capillary numbers reported in the experimental study.

The measurement of the droplet dimensions in the experimental work [4] was achieved via processing of rather low-resolution digital images which produced an average deviation of 5% with respect to measurements of high-resolution (film) photographs [31]. In all cases we have studied, our numerical results are in excellent agreement with the experimental findings (i.e. the two determinations are very close or within the 5% deviation) or in very good agreement with the experimental results (i.e. the two determinations are quite close).

In all cases presented in this subsection (as well as in Section 7.2), the initial shape at time $t = 0$ corresponds to a spherical droplet, i.e. we consider the problem where a steady flow is introduced into the system for all times $t > 0$. Our preference is to discretize this spherical shape into $N_E = 6$ equal-size spectral elements as shown in Fig. 1. The results we present are derived by employing $N_B = 10$ basis points and the third-order diagonally implicit Runge–Kutta scheme (DIRK3) with $\Delta t = 0.1$ (unless noted otherwise). The accuracy of our results was verified by employing smaller time steps (usually $\Delta t = 0.05, 0.02, 0.01$), basis points $N_B = 7, \ldots, 13$ as well as other implicit schemes. The three-dimensional droplet shapes presented in this subsection as well as in Section 7.2 show the actual spectral grid used for the interfacial solution.

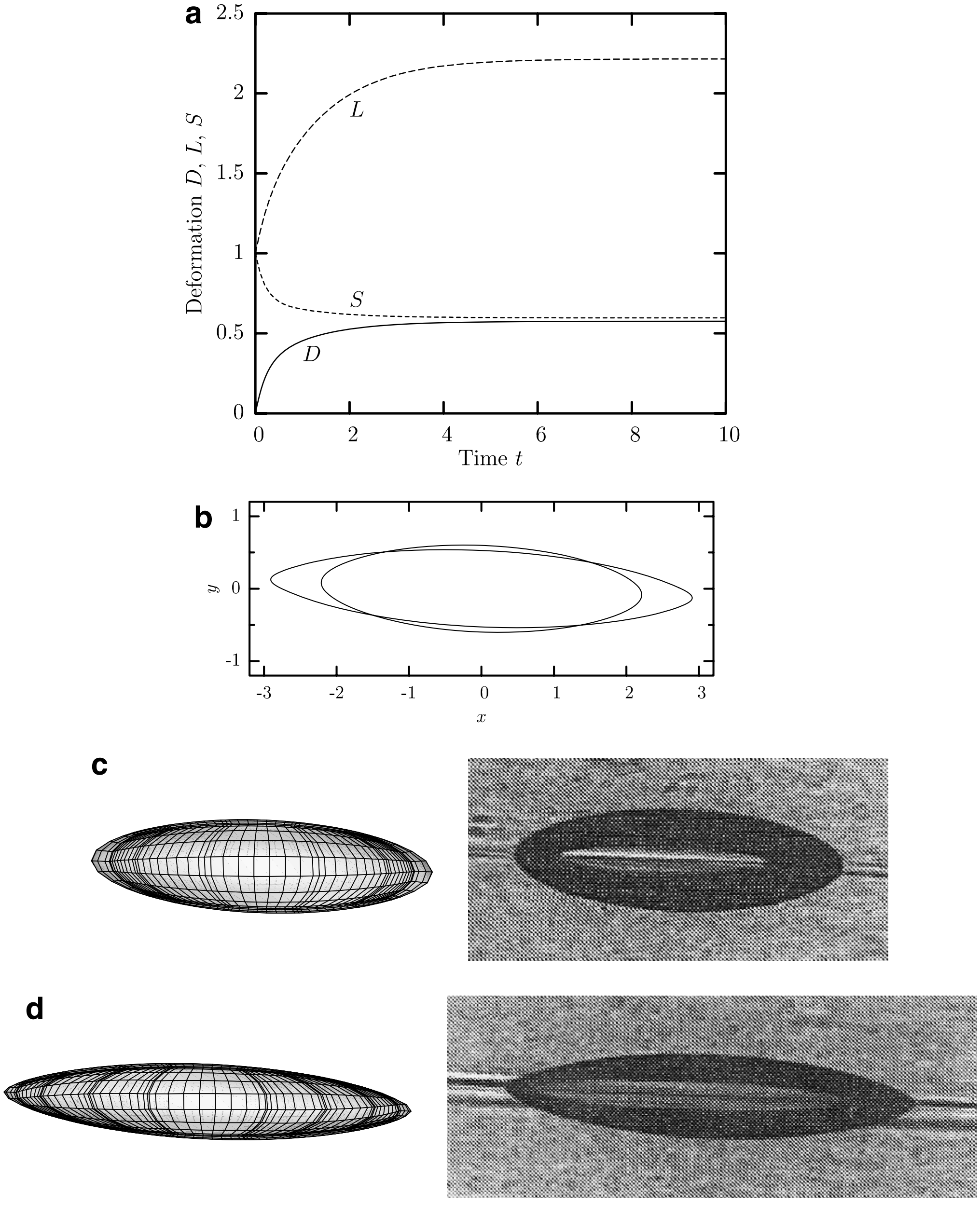

Fig. 5. Subcritical evolution of a droplet with $\lambda = 0.00108$ in a planar four-roll mill flow with $a_f = 0.8$. (a) Time evolution of the droplet deformation D and its major L and minor S semi-axes for $Ca = 0.263$. (b) Droplet profile at equilibrium for $Ca = 0.263, 0.321$. (c) Droplet shape at equilibrium for $Ca = 0.263$ from our computations and the experimental work of Bentley and Leal [4] reported in their Fig. 7. (d) As in (c) but for $Ca = 0.321$.

In strong flows, both at subcritical conditions presented in this subsection as well as at supercritical conditions presented in Section 7.2, the droplet may elongate significantly. Thus, the original spectral discretization $\{N_E, N_B\}$ may not be sufficient to describe a very deformed interfacial shape. To resolve this issue, we utilize the adaptive mesh reconstruction procedure presented in our earlier publication on our explicit interfacial spectral boundary element method [34]. This procedure divides or unites the spectral elements during the transient evolution based on their relevant length but keeps fixed the number of basis points, N_B, defined on each element. The goal of our adaptive mesh reconstruction is to produce a reasonable spectral element discretization of the interfacial shape with respect to the element's arc lengths and/or the variation of curvature on it. Thus, as relevant lengths of a spectral element are regarded its arc length L_1 and its curvature length L_2 along each curvilinear direction,

$$L_1 = \int_{\text{elem}} \mathrm{d}\ell, \quad L_2 = R_{\text{ref}} \int_{\text{elem}} |\nabla \cdot \boldsymbol{n}| \mathrm{d}\ell \tag{23}$$

where $\mathrm{d}\ell$ is the arc length and R_{ref} some reference radius of curvature. (For problems involving surfaces in close contact, another relevant length may be considered associated with the surfaces' gap [9,40].)

For example, the moderately deformed interfacial shapes shown in Fig. 6(*c*) are determined with $N_E = 14$ elements produced by division of each of the two edge elements of the original $N_E = 6$ discretization into five new elements during the transient evolution. The same happens for the more elongated shapes shown in Fig. 5 where more divisions of the edge elements were employed.

Fig. 5 shows the subcritical evolution of a low-viscosity droplet with $\lambda = 0.00108$ in a planar four-roll mill flow with $a_f = 0.8$. The time evolution of the droplet length L, width S and deformation D is shown in Fig. 5a, while Fig. 5b–d shows the profile and shape of the droplet at equilibrium for $Ca = 0.263, 0.321$. The equilibrium interfacial shapes and the corresponding values of the deformation $D = 0.575$ (for $Ca = 0.263$), the droplet length $L = 2.91$ (for $Ca = 0.321$) and orientation angle $\theta_x = -2.4, -2.5$ are in very good agreement with the experimental results of Bentley and Leal [4] reported in their Fig. 7, i.e. $D^{\text{BL}} = 0.531$, $L^{\text{BL}} = 2.70$ and $\theta_x^{\text{BL}} = -1.5, -2$.

For the low-viscosity droplet presented in Fig. 5, we note the appearance of pointed edges especially at the higher capillary number $Ca = 0.321$. Our spectral method along with its adaptive mesh procedure can accurately describe these tips without the need to accumulate a very large number of grid points near the edges.

The subcritical evolution of a high-viscosity droplet with $\lambda = 14.4$, for $a_f = 0.4$ and $Ca = 0.172$, is shown in Fig. 6. The time evolution of the droplet deformation D is shown in Fig. 6a while the interfacial profile and shape are shown in Fig. 6b and c. The equilibrium interfacial shapes and the corresponding values of the deformation $D = 0.329$ and orientation angle $\theta_x = -16.8$ are in very good agreement with the experimental results of Bentley and Leal [4] reported in their Fig. 13, i.e. $D^{\text{BL}} = 0.337$ and $\theta_x^{\text{BL}} = -19$.

For high-viscosity droplets, the time necessary to reach equilibrium is much longer compared to that for equal- and low-viscosity droplets. An estimate for this time may be given by the surface tension time scale incorporating the influence of the viscosity ratio, e.g. $\tau_\gamma = (1+\lambda)\mu a/\gamma = (1+\lambda)Ca\tau_f$. The term showing the influence of the viscosity ratio, i.e. $(1+\lambda)$, is only asymptotically correct at the extreme limits $\lambda \to 0$ and $\lambda \to \infty$; for intermediate values of λ, the influence of the viscosity ratio is more complicated and may be affected by the flow type and other parameters of the specific interfacial system.

As shown in Fig. 6a for the determination of the deformation of the high-viscosity droplet we employed several time steps, i.e. $\Delta t = 0.1, 1, 5, 25$. Observe that the first three time steps produce accurate results at all times. The larger time step $\Delta t = 25$ produces inaccurate results during the transient droplet evolution (see its prediction for time $t = 25$) and accurate prediction of the steady-state conditions where the accuracy of the computational results does not depend on the time step (but only on the space discretization). In addition, observe that even the large time step $\Delta t = 25$ produces stable evolution towards the correct equilibrium.

By studying our numerical results for different time steps, such as those included in Figs. 6a and 2b, the following question arises: for a specific subcritical evolution, what is the largest time step Δt where accurate solution is obtained at all times, especially far from equilibrium? Our numerical results suggest that when we employ as a time step a small fraction of the actual time T necessary to reach equilibrium, we always obtain accurate results. For example, for the case shown in Fig. 6a $T \approx 50$; in this case by utilizing $\Delta t = 0.1$ T or smaller, the numerical solution is accurate at all times, e.g. even for the first few time steps. On the other hand, close to equilibrium much higher time steps may successfully be employed.

These conclusions may be employed for the efficient and accurate determination of slow dynamic processes such that of the subcritical evolution of high-viscosity droplets. In particular, we have the ability to start with a small enough time step so that we determine accurately an initial fast evolution; then the time step of our implicit method can be increased gradually as the evolution becomes less rapid; near equilibrium quite large time steps can be employed to cover any desired time period. (As an example, we have implemented this

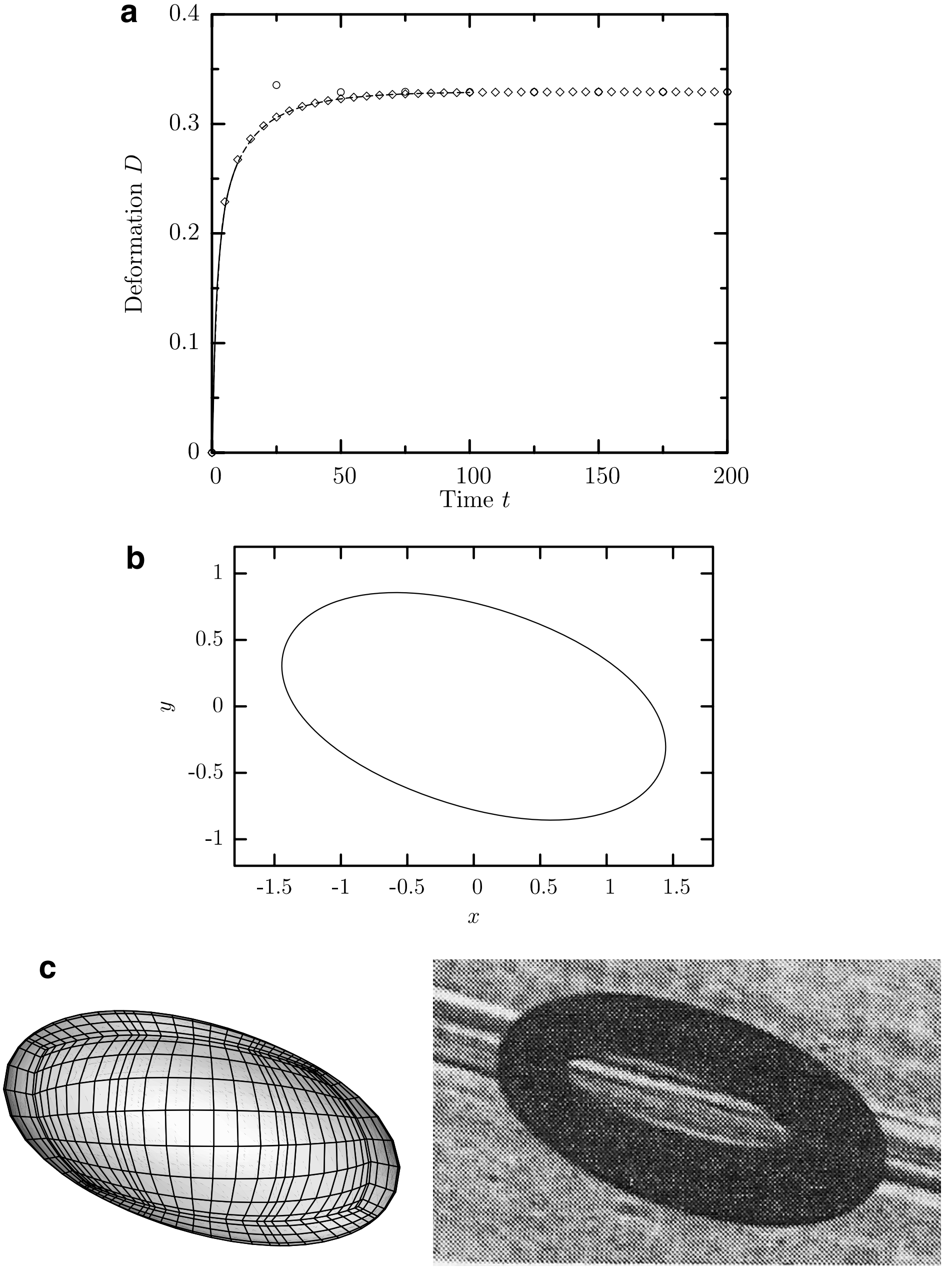

Fig. 6. Subcritical evolution of a droplet with $\lambda = 14.4$ in a planar four-roll mill flow with $a_f = 0.4$ and for $Ca = 0.172$. (a) Time evolution of the droplet deformation D for different time steps: ——, $\Delta t = 0.1$ in $[0, 20]$; ----, $\Delta t = 1$ in $[0, 100]$; ◇, $\Delta t = 5$; ○, $\Delta t = 25$. (Note that for $\Delta t = 5, 25$ our computations cover the time period $[0, 500]$ even though we present results only in $[0, 200]$.) The steady-state shape was also determined by employing one large time step with $\Delta t = 10^4$ and the implicit Euler method. (b) Droplet profile at equilibrium. (c) Droplet shape at equilibrium from our computations and the experimental work of Bentley and Leal [4] reported in their Fig. 13.

procedure for the determination of the evolution of the high-viscosity droplet shown in Fig. 6.) We emphasize that this procedure may be employed for the subcritical drop evolution at any viscosity ratio.

As an additional issue involving subcritical droplet evolution, we note that the equilibrium shape can be directly calculated by employing our implicit spectral method with a very large time step. This shows that in this case, our implicit method reverses to our Newton method for equilibrium shapes, as expected. In this case any implicit scheme may be employed for the accurate determination of the equilibrium conditions. The most efficient way is to implement the implicit Euler method as we have done for the high-viscosity droplet shown in Fig. 6.

As a closure to this subsection, we emphasize that the employed spectral discretization is sufficient to produce a small enough relative error (e.g. $\leqslant 10^{-3}$); however it is not necessary for this. In most cases, a smaller number of spectral elements and/or a smaller number of basis points may also achieve the desired accuracy. As an example, we mention that, for the case $Ca = 0.263$ shown in Fig. 5, identical time evolution and equilibrium shape was achieved by employing $N_{\mathrm{E}} = 14$ spectral elements instead of $N_{\mathrm{E}} = 22$ elements shown in Fig. 5c. (The coarser discretization was produced by setting a higher limit for element division resulting in a single division of the two edge elements during the transient evolution.)

7.2. Supercritical droplet evolution

In this subsection, we study the droplet evolution at supercritical capillary numbers to show the ability of our implicit method to determine efficiently and accurately elongated droplet shapes. Similarly to Section 7.1,

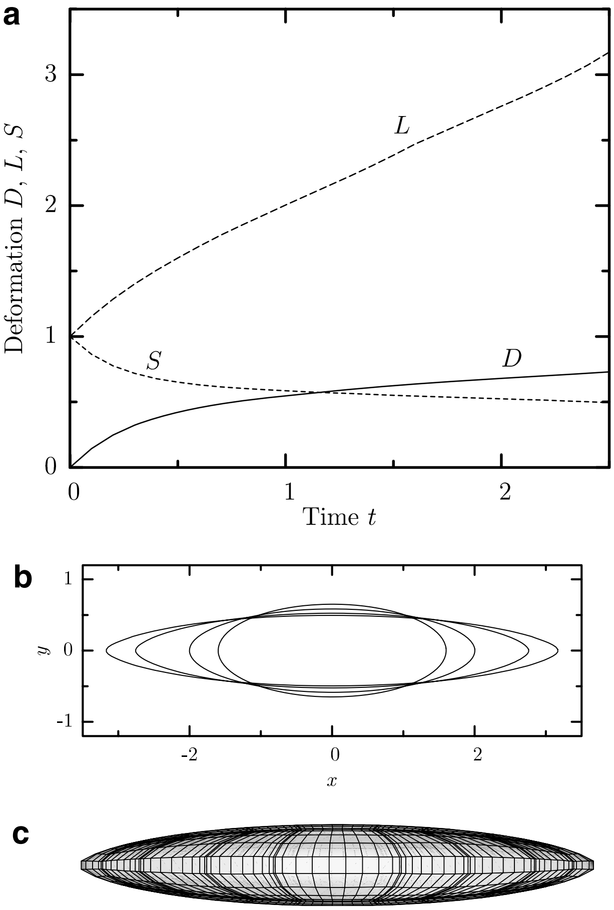

Fig. 7. Supercritical evolution of a droplet with $\lambda = 0.005$ in a planar planar extensional flow for $Ca = 0.303$. (a) Time evolution of the droplet deformation D and its major L and minor S semi-axes. (b) Droplet profile at times $t = 0.5, 1, 2, 2.5$. (c) Droplet shape at $t = 2.5$.

we investigate the time evolution of the droplet length L, width S and deformation D for different viscosity ratios. We also compare our numerical results with the experimental results of Ha and Leal [16] who studied drop deformation in a planar extensional flow $\boldsymbol{u}^\infty = G(x, -y, 0)$.

Fig. 7 shows the supercritical evolution of a low-viscosity droplet with $\lambda = 0.005$ in a planar extensional flow with $Ca = 0.303$. The time evolution of the droplet length L, width S and deformation D is shown in Fig. 7a. In addition, Fig. 7b shows the droplet profile for several times while the droplet shape at time $t = 2.5$ is shown in Fig. 7c. We note that our numerical results are in excellent agreement with the experimental findings of Ha and Leal [16] reported in their Fig. 5a.

As the drop elongates with time, tips appear at the edges of the drop as shown in the droplet profiles in Fig. 7b. At time $t = 2.5$ the droplet length is $L \approx 3.2$ and thus the droplet shape included in our Fig. 7c closely corresponds to the first photo included in Fig. 5a of Ref. [16]. Observe that our numerical shape is in excellent agreement with the experimental photo. We note that the accurate determination of tip formation in three-dimensional interfaces constitutes a challenging computational problem.

We consider now the transient evolution of a droplet with $\lambda = 0.209$ for the supercritical capillary number $Ca = 0.163$ shown in Fig. 8a. Note that after the initial fast increase of the droplet length, the rate of elongation slows down, while later a very sharp increase in the droplet length is observed. Fig. 8b and c shows the

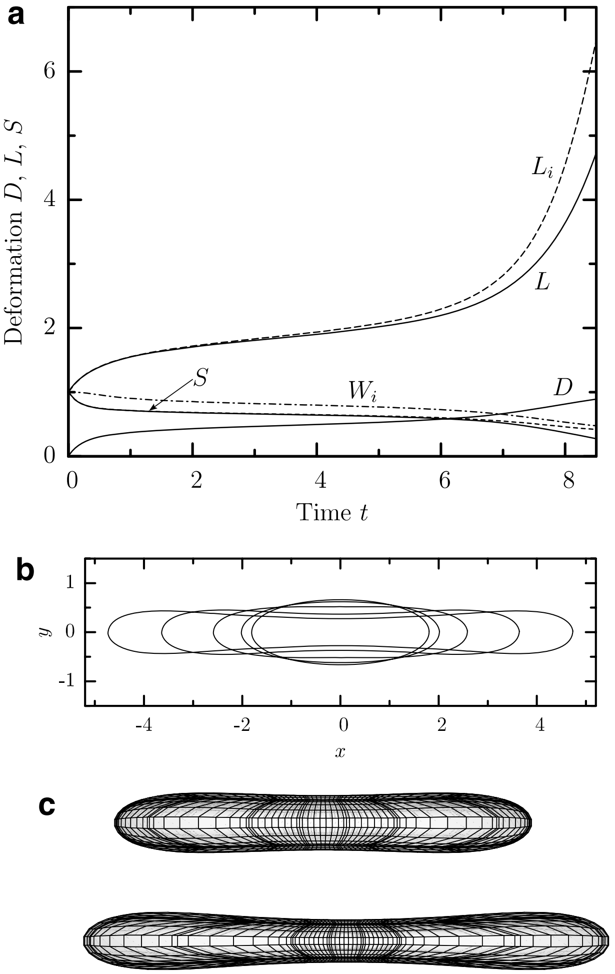

Fig. 8. Supercritical evolution of a droplet with $\lambda = 0.209$ in a planar extensional flow for $Ca = 0.163$. (a) Time evolution of the droplet deformation D and its major L and minor S semi-axes. Also included are the drop's lengths determined as the semi-axes of the ellipsoid which has the same inertia tensor as the droplet: ——, L_i; - - - -, S_i; –·–, W_i. (b) Droplet profile at times $t = 3, 5, 7, 8, 8.5$. (c) Droplet shape at times $t = 7.7, 8.2$.

droplet profile and shape at several times. Observe that later in time a thin neck is created in the middle of the drop as shown in the drop profile for $t = 8.5$ included in Fig. 8b. Our numerical results are thus consistent with the experimental findings of Ha and Leal [16] reported in their Fig. 2a.

The three-dimensional drop shape for $t = 8.2$ included in Fig. 8c shows three narrow rows of (four) spectral elements in the middle of the droplet. To reduce the computational cost, our algorithm can unite them into one row of elements around the waist of the droplet. In this experiment, we have avoided to do this, to show clearly that our implicit method is not affected by the Courant condition when an accumulation of grid points occurs locally on the drop interface.

In Fig. 8a we also include the time evolution of the drop's lengths, L_i, S_i and W_i, determined as the semi-axes of the ellipsoid which has the same inertia tensor as the droplet. A comparison of these lengths with the droplet's true longest and shortest semi-axes, L and S, reveals that for this case the quantities based on the inertia tensor overestimate the actual droplet dimensions at high elongation. This difference is more severe for the length L_i which overestimates rather significantly the drop's length L. On the other hand, the width S_i agrees rather well with the drop's width S until very late in the transient evolution where a thin neck is created in the middle of the drop.

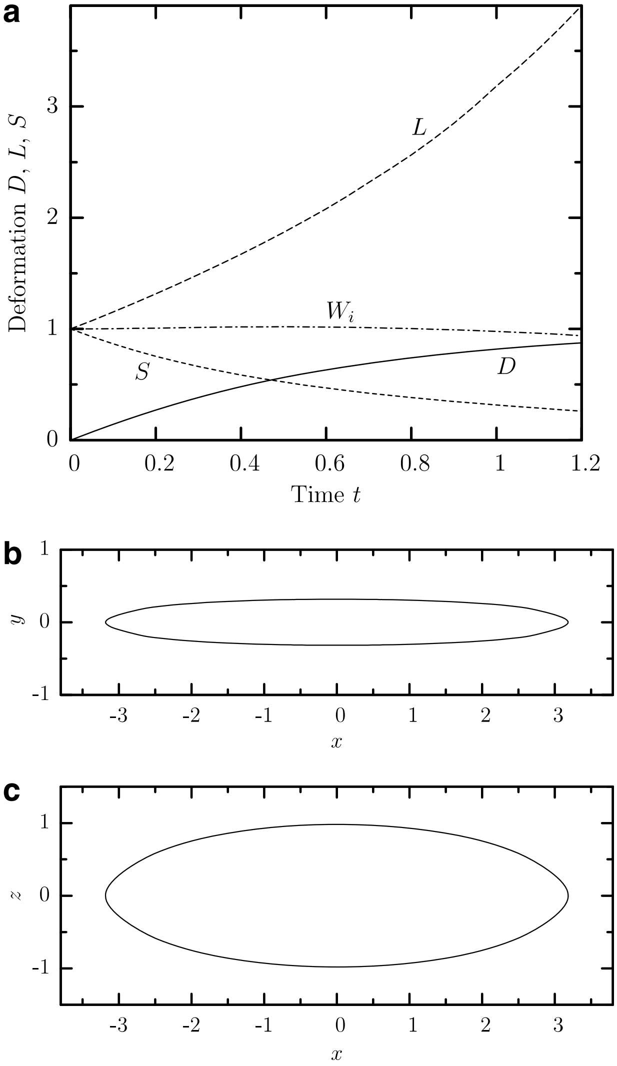

Fig. 9. Supercritical evolution of a droplet with $\lambda = 0.209$ in a planar extensional flow for $Ca = 1.55$. (a) Time evolution of the droplet deformation D, length L, width S and depth W_i. (The curves for L_i and S_i coincide with those for L and S, respectively.) (b) Droplet profile along the plane $z = 0$ at time $t = 1$. (c) As in (b) but for the plane $y = 0$.

We emphasize that this conclusion applies to all extended shapes we have presented so far in this section, at both subcritical and supercritical conditions. Our results for the rest cases of this section show that it is the length L_i which deviates from the true drop length L. In particular, for elongated shapes with pointed edges or tips, like those for the low-viscosity drops presented in Figs. 5 and 7, L_i underestimates the drop length. On the other hand, for elongated shapes with rounded edges like those for the moderate- and high-viscosity drops presented in Figs. 6 and 8, L_i overestimates the true drop length L. Both variations are consistent with the average nature of the inertia tensor whose determination involves the entire droplet surface.

Thus, one should be careful when using the lengths based on the inertia tensor to represent the actual drop's dimensions. On the other hand, the two ways to calculate the drop's length (e.g. discussed at the beginning of Section 6) may be used to determine until which extension the droplet shape can be approximated with an ellipsoid. For example, for the case depicted in Fig. 8, this happens for times $t \lesssim 3$ where the inertia tensor lengths, L_i and S_i, agree with the true droplet lengths, L and S.

Considering again Fig. 8a, we observe that while the drop's length and width start to increase or decrease immediately after the initiation of the flow, the drop's depth W_i is practically constant at early times and then shows a slow decrease. This points out to the dynamics of this specific flow which are different in the x, y and z directions. At supercritical flow rates, a planar extensional flow $\boldsymbol{u}^\infty = G(x, -y, 0)$ tends to rapidly increase the drop length in the x-direction, decrease its width in the y-direction, while initially may keep unchanged the drop depth in the z-direction.

To investigate further this issue, we have considered droplets in higher flow rates. In Fig. 9 we present the transient evolution of a droplet with $\lambda = 0.209$ for the supercritical capillary number $Ca = 1.55$. This flow rate corresponds to $Ca = 10Ca_\mathrm{c}$ based on our determination of the critical capillary number $Ca_\mathrm{c} \approx 0.155$ (which is in agreement with the experimental findings of Bentley and Leal [4] reported in their Fig. 28). As shown in Fig. 9a, while the droplet's length and width show a significant change, the depth W_i of the droplet is practically constant until $t \approx 0.8$. This results in the creation of temporary lamellar shapes as reported in Ref. [10]. By comparing the drop's profiles shown in Fig. 9 with those presented in Fig. 8, it is obvious that the increased flow rate reported in Fig. 9 (i.e. $Ca = 1.55$) is accompanied with the appearance of pointed edges which are in distinct contrast to the rounded ends found for $Ca = 0.163$.

8. Conclusions

In this paper we have developed an efficient, fully-implicit, three-dimensional interfacial dynamics algorithm based on a mathematically rigorous combination of implicit formulas with our Jacobian-free Newton method. The resulting algorithm has several desirable properties which are unique since the introduction of the boundary integral equations for interfacial dynamics in Stokes flow 30 years ago by Acrivos and coworkers [28,37]. First, our implicit interfacial algorithm removes the penalty of small time steps associated with the Courant condition, Eq. (9), by making the employed time step Δt independent of the space discretization Δx_min and the capillary number Ca. In addition, physical stiffness does not affect our implicit algorithm. Furthermore, due to the linearization inherent in the involved Newton method, any non-linear boundary condition can be easily embodied into our algorithm and solved directly along with the interfacial evolution without the need for a post-solution treatment/correction.

In this work the numerical solution of our implicit algorithm is achieved through the spectral boundary element method [12,23] which makes the combined methodology the only available three-dimensional high-order implicit interfacial algorithm. However, we emphasize that our fully-implicit time integration for interfacial dynamics in Stokes flow may be combined with any existing boundary integral algorithm.

Our fully-implicit methodology constitutes a new approach for the study of interfacial dynamics in Stokes flow of a broad range of physical systems including multiphase flows, drops, bubbles, capsules and biological cells. In addition, our methodology is not restricted to Stokes flow only but it can be directly employed to any system involving deformable interfaces governed by boundary integral equations such as the thermocapillary or electro-magnetic interfacial motion, and solidification. Thus, our algorithm is particularly suited for the efficient study of a variety of physical problems in regular- and small-scale systems, such as porous media, microfluidics and biomechanics.

Acknowledgements

This work was supported in part by the National Science Foundation. Acknowledgment is made to the Donors of the American Chemical Society Petroleum Research Fund for partial support of this research. Some computations were performed on multiprocessor supercomputers provided by the National Center for Supercomputing Applications (NCSA) in Illinois.

References

[1] M. Abramowitz, I.A. Stegun (Eds.), Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables, Dover, 1972.

[2] R. Alexander, Diagonally implicit Runge–Kutta methods for stiff O.D.E.'s, SIAM J. Numer. Anal. 14 (1997) 1006–1021.

[3] I.B. Bazhlekov, P.D. Anderson, H.E.H. Meijer, Nonsingular boundary integral method for deformable drops in viscous flows, Phys. Fluids 16 (2004) 1064–1081.

[4] B.J. Bentley, L.G. Leal, An experimental investigation of drop deformation and breakup in steady, two-dimensional linear flows, J. Fluid Mech. 167 (1986) 241–283.

[5] J. Blawzdziewicz, V. Cristini, M. Loewenberg, Critical behavior of drops in linear flows. I. Phenomenological theory for drop dynamics near critical stationary states, Phys. Fluids 14 (2002) 2709–2718.

[6] J.C. Butcher, Numerical Methods for Ordinary Differential Equations, Wiley, 2003.

[7] C. Canuto, M.Y. Hussaini, A. Quarteroni, T.A. Zang, Spectral Methods in Fluid Dynamics, Springer, 1998.

[8] R.G. Cox, The dynamics of the spreading of liquids on a solid surface. Part 1. Viscous flow, J. Fluid Mech. 168 (1986) 169–194.

[9] V. Cristini, J. Blawzdziewicz, M. Loewenberg, An adaptive mesh algorithm for evolving surfaces: simulations of drop breakup and coalescence, J. Comput. Phys. 168 (2001) 445–463.

[10] V. Cristini, R.W. Hooper, C.W. Macosko, M. Simeone, S. Guido, A numerical and experimental investigation of lamellar blend morphologies, Ind. Eng. Chem. Res. 41 (2002) 6305–6311.

[11] R.H. Davis, Buoyancy-driven viscous interaction of a rising drop with a smaller trailing drop, Phys. Fluids 11 (5) (1999) 1016–1028.

[12] P. Dimitrakopoulos, J.J.L. Higdon, On the displacement of three-dimensional fluid droplets from solid surfaces in low-Reynolds-number shear flows, J. Fluid Mech. 377 (1998) 189–222.

[13] S.O. Fatunla, Numerical Methods for Initial Value Problems in Ordinary Differential Equations, Academic Press, 1988.

[14] R.T. Foister, The kinetics of displacement wetting in liquid/liquid/solid systems, J. Colloid Interf. Sci. 136 (1990) 266–282.

[15] S. Guido, M. Villone, Three-dimensional shape of a drop under simple shear flow, J. Rheol. 43 (2) (1998) 395–415.

[16] J.W. Ha, L.G. Leal, An experimental study of drop deformation and breakup in extensional flow at high capillary number, Phys. Fluids 13 (2001) 1568–1576.

[17] E. Hairer, S.P. Norsett, G. Wanner, Solving Ordinary Differential Equations, vol. I and II, Springer, 1993.

[18] M.Y. Hussaini, T.A. Zang, Spectral Methods in Fluid Dynamics, Ann. Rev. Fluid Mech. 19 (1987) 339–367.

[19] M. Loewenberg, E.J. Hinch, Numerical simulation of a concentrated emulsion in shear flow, J. Fluid Mech. 321 (1996) 395–419.

[20] M. Loewenberg, E.J. Hinch, Collision of two deformable drops in shear flow, J. Fluid Mech. 338 (1997) 299–315.

[21] Y. Maday, A.T. Patera, Spectral element methods for the incompressible Navier–Stokes Equations, in: A.K. Noor, J.T. Oden (Eds.), State of the Art Surveys in Computational Mechanics, ASME, 1989.

[22] A. Mazouchi, C.M. Gramlich, G.M. Homsy, Time-dependent free surface Stokes flow with a moving contact line. I. Flow over plane surfaces, Phys. Fluids 16 (2004) 1647–1659.

[23] G.P. Muldowney, J.J.L. Higdon, A spectral boundary element approach to three-dimensional Stokes flow, J. Fluid Mech. 298 (1995) 167–192.

[24] Y. Navot, Critical behavior of drop breakup in axisymmetric viscous flow, Phys. Fluids 11 (1999) 990–996.

[25] C. Pozrikidis, Boundary Integral and Singularity Methods for Linearized Viscous Flow, Cambridge University Press, Cambridge, 1992.

[26] C. Pozrikidis, Interfacial dynamics for stokes flow, J. Comput. Phys. 169 (2001) 250–301.

[27] C. Pozrikidis (Ed.), Modeling and Simulation of Capsules and Biological Cells, Chapman & Hall, 2003.

[28] J.M. Rallison, A. Acrivos, A numerical study of the deformation and burst of a viscous drop in an extensional flow, J. Fluid Mech. 89 (1978) 191–200.

[29] J.M. Rallison, A numerical study of the deformation and burst of a viscous drop in general shear flows, J. Fluid Mech. 109 (1981) 465–482.

[30] J.M. Rallison, The deformation of small viscous drops and bubbles in shear flows, Ann. Rev. Fluid Mech. 16 (1984) 45–66.

[31] H.A. Stone, B.J. Bentley, L.G. Leal, An experimental study of transient effects in the breakup of viscous dropls, J. Fluid Mech. 173 (1986) 131–158.

[32] H.A. Stone, Dynamics of drop deformation and breakup in viscous fluids, Ann. Rev. Fluid Mech. 26 (1994) 65–102.

[33] H.A. Stone, A.D. Stroock, A. Ajdari, Engineering flows in small devices: microfluidics toward a lab-on-a-chip, Ann. Rev. Fluid Mech. 36 (2004) 381–411.

[34] Y. Wang, P. Dimitrakopoulos, A three-dimensional spectral boundary element algorithm for interfacial dynamics in Stokes flow, Phys. Fluids 18 (1–16) (2006) 082106.

[35] Y. Wang, W.R. Dodson, P. Dimitrakopoulos, Dynamics of multiphase flows via spectral boundary elements and parallel computations, in: A. Deane, G. Brenner, A. Ecer, D. Emerson, J. McDonough, J. Periaux, N. Satofuka, D. Tromeur-Dervout (Eds.), Parallel Computational Fluid Dynamics: Theory and Applications, Proceedings of the Parallel CFD 2005 Conference, College Park, MD, USA, (May 24–27, 2005), Elsevier, 2006.

[36] E.D. Wetzel, C.L. Tucker III, Drop deformation in dispersions with unequal viscosities and zero interfacial tension, J. Fluid Mech. 426 (2001) 199–228.

[37] G.K. Youngren, A. Acrivos, On the shape of a gas bubble in a viscous extensional flow, J. Fluid Mech. 76 (1976) 433–442.

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

[39] A.Z. Zinchenko, R.H. Davis, Shear flow of highly concentrated emulsions of deformable drops by numerical simulations, J. Fluid Mech. 455 (2002) 21–62.

[40] A.Z. Zinchenko, R.H. Davis, A multipole-accelerated algorithm for close interaction of slightly deformable drops, J. Comput. Phys. 207 (2005) 695–735.