A Lagrangian–Eulerian approach for the numerical simulation of free-surface flow of a viscoelastic material

Jocelyn Étienne [a,*], E.J. Hinch [a], Jie Li [b]

[a] *University of Cambridge, Department Applied Mathematics and Theoretical Physics, Wilberforce Road, Cambridge CB3 0WA, UK*
[b] *University of Cambridge, BP Institute & Engineering Department, Madingley Road, Cambridge CB3 0EZ, UK*

Abstract

A new numerical method specially adapted to the free-surface flow of viscoelastic material is proposed. It is based on a Lagrangian discretisation of the material and objective derivatives, which accounts well for the hyperbolic nature of these terms and goes well with a Lagrangian tracking of a time-evolving domain. Through the Arbitrary Lagrangian–Eulerian (ALE) formulation, the method can also be applied efficiently to solid-boundary problems, and is tested on the benchmark problem of the drag on a cylinder in a channel. The collapse of a column of Oldroyd-B fluid is then considered: under the action of surface tension, the column undergoes large deformation leading to the "beads-on-string" structure. Asymptotic results on the evolution of this structure are recovered in numerical simulations, and further features of this flow are exhibited.
© 2006 Elsevier B.V. All rights reserved.

1. Introduction

Free-surfaces and the extra-stress tensor are both quantities whose local properties at one given time essentially depend, not on the history of the flow at the point considered, but on its history along the material trajectory passing through this point and time. In one word, they are naturally understood from a Lagrangian point of view, moving along the flow, rather than from the observer's Eulerian point of view. This obvious remark has well-known, deep implications: the equations governing the evolution of such quantities are of hyperbolic nature, and thus the boundary conditions and numerical methods applied to them need to take into account this hyperbolic nature.

It is rather natural to try to take advantage of this common point between free-surface and viscoelastic flows, rather than having two separate difficulties, and thus to use a Lagrangian point of view for both of them. This is done in a very natural way for the free-surface, and several approaches have been pro-

posed for treating viscoelastic flow from a Lagrangian point of view. Rasmussen and Hassager [1,2] simulate creeping flows of viscoelastic fluids described by an integral model and with zero solvent viscosity using a Lagrangian integration of the memory function. Alternatively, fully Lagrangian formulation of the objective derivative has been proposed by Fortin and Esselaoui [3], which allows the simulation of flows of fluids described by a differential constitutive equation. A similar idea is used by Harlen et al. [4], where the simulation cleverly employs a piecewise constant rate of deformation to express the co-rotational derivative solely in terms of the deformation of a Lagrangian mesh. The idea of Fortin and Esselaoui can be seen as a generalisation of this procedure, were a local change-of-basis matrix tracks the deformation along material trajectories. Thus transformed, the governing equations of a moving-boundaries, viscoelastic flow (in the case of an Oldroyd-B fluid for the sake of simplicity) are presented in Section 2, and their discretisation in time in Section 3. A mixed finite elements space discretisation is introduced in Section 4, the particulars of the simulation of surface-tension driven free-surface are developed in Section 5.

It is well known that purely Lagrangian methods are not well suited when solid boundaries are involved, because the mesh is sheared by the flow and frequent remeshing is

---

 * Corresponding author. Tel.: +44 1223 337877; fax: +44 1223 765900.
 E-mail addresses: J.Etienne@damtp.cam.ac.uk (J. Étienne),
E.J.Hinch@damtp.cam.ac.uk (E.J. Hinch), jie@bpi.cam.ac.uk (J. Li).

thus needed to preserve the accuracy of the solution. Also, inflow and outflow boundaries are not naturally dealt with by Lagrangian methods. Thus, we introduce in Section 6 an Arbitrary Lagrangian–Eulerian (ALE) extension of the method, in which case the mesh is advected with an arbitrary advection velocity, which can locally differ from the material velocity. In the case when this advection of the mesh is zero, the algorithm can deal with fixed boundary problems, and the method is validated in Section 7 by comparison of the calculated drag on a cylinder in a channel flow with the values obtained by other authors (with codes specifically adapted to this latter fixed-boundary case).

Finally, we turn in Section 8 to a free-surface flow of both theoretical and practical interest, the collapse of a column of Oldroyd-B fluid. After the initial destabilisation of the column, elastic stresses build up in necking regions and control the rate of thinning. Thus, long threads of stretched material form, which connect drops of nearly relaxed fluid, a structure known as beads-on-string, with the thread radius exponentially decaying at a rate controlled by the Deborah number only [5–7]. Pioneer numerical work on this problem was done by Keunings [8], where the beads-on-string structure is obtained for a creeping flow of Oldroyd-B fluid by means of an axisymmetric simulation. The same technique is used by Bousfield et al. [9] in order to validate a one-dimensional model, with which they obtain the decay predicted by asymptotic theory over a short time range for a Deborah number of 2. The simulations presented in this article are axisymmetric and for Deborah numbers reaching 300. A good agreement with the asymptotic law is exhibited over a time equal to the relaxation time of the polymers.

2. Lagrangian–Eulerian formulation of the governing equations

Let us consider a viscoelastic material, composed of a solvent of viscosity, η_s, and polymers making a contribution to the solution viscosity, η_p, and having a relaxation time, λ, occupying at some reference time t_r the domain of space Ω_r. As displacement occurs, the material will occupy at time t a domain $\Omega(t)$, this domain is bounded by solid boundaries $\Gamma_D(t)$ and free boundaries $\Gamma_N(t)$.

The conservation of momentum and mass in the domain $\Omega(t)$ are expressed as:

$$Re \left(\frac{\partial}{\partial t} + \boldsymbol{u}.\nabla \right) \boldsymbol{u} = -\nabla p + (1 - \alpha)\nabla \cdot (2\mathbf{D}\boldsymbol{u}) + \nabla \cdot \boldsymbol{\sigma} \quad (1)$$

$$\nabla \cdot \boldsymbol{u} = 0 \quad (2)$$

where $\boldsymbol{u}$ is the velocity, p the pressure, $\mathbf{D}\boldsymbol{u}$ the rate-of-strain tensor $(\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^T)/2$ and $\boldsymbol{\sigma}$ is the extra-stress tensor. Here, we have assumed that the fluid has a constant density ϱ_0 and put $\alpha = \eta_p/(\eta_s + \eta_p)$, the fraction of polymer contribution to the total viscosity. The Reynolds number is therefore, $Re = \varrho_0 UR/(\eta_s + \eta_p)$, with R and U characteristic length and viscosity. Boundary conditions are

$$\boldsymbol{u}|_{\Gamma_D} = \boldsymbol{f}_D; \qquad \boldsymbol{\sigma}_{\text{tot}}\boldsymbol{n}|_{\Gamma_N(t)} = \boldsymbol{f}_N \quad (3)$$

where $\boldsymbol{\sigma}_{\text{tot}} = -p\boldsymbol{I} + 2(1 - \alpha)\mathbf{D}\boldsymbol{u} + \boldsymbol{\sigma}$ and initial conditions are provided at time $t = t_0$, $\boldsymbol{u}(t_0) = \boldsymbol{u}_0$.

The extra-stress tensor obeys the Oldroyd-B constitutive law:

$$\boldsymbol{\sigma} + De \overset{\triangledown}{\boldsymbol{\sigma}} = 2\alpha\mathbf{D}\boldsymbol{u} \quad (4)$$

where $\overset{\triangledown}{\boldsymbol{\sigma}}$ denotes the upper-convected objective derivative of $\boldsymbol{\sigma}$,

$$\overset{\triangledown}{\boldsymbol{\sigma}} = \left(\frac{\partial}{\partial t} + \boldsymbol{u} \cdot \nabla \right) \boldsymbol{\sigma} - (\nabla \boldsymbol{u})^T \boldsymbol{\sigma} - \boldsymbol{\sigma}\nabla \boldsymbol{u} \quad (5)$$

with the convention that

$$(\nabla \boldsymbol{u})_{ij} = \frac{\partial u_j}{\partial x_i}$$

and the Deborah number, $De = \lambda U/R$, is the non-dimensional relaxation time of the fluid. Because there is no inflow, no boundary conditions are necessary for $\boldsymbol{\sigma}$; only initial conditions $\boldsymbol{\sigma}(t_0) = \boldsymbol{\sigma}_0$.

$\Omega(t)$ can be related to Ω_r by the trajectories $t \mapsto X(\boldsymbol{x}_r, t)$ of material points of reference coordinates $\boldsymbol{x}_r \in \Omega_r$ as $\Omega(t) = X(\Omega_r, t)$. The material trajectories are tangent to the instantaneous velocity of the material $\boldsymbol{u}[X(\boldsymbol{x}_r, t), t]$,

$$\begin{cases} \dfrac{\partial X}{\partial t}(\boldsymbol{x}_r, t) = \boldsymbol{u} \circ X(\boldsymbol{x}_r, t), & t \in [0, T], \\ X(\boldsymbol{x}_r, t_r) = \boldsymbol{x}_r, \end{cases}$$

where $\boldsymbol{u} \circ X(\boldsymbol{x}_r, t)$ is understood as $\boldsymbol{u}[X(\boldsymbol{x}_r, t), t]$. Then the governing equations can be rewritten in the domain of reference Ω_r thanks to the mapping X:

$$Re \frac{\partial \boldsymbol{u} \circ X}{\partial t} = (-\nabla p + (1 - \alpha)\nabla \cdot (2\mathbf{D}\boldsymbol{u}) + \nabla \cdot \boldsymbol{\sigma}) \circ X, \quad (6a)$$

$$(\nabla \cdot \boldsymbol{u}) \circ X = 0 \quad (6b)$$

$$\boldsymbol{\sigma} \circ X + De \left(\frac{\partial \boldsymbol{\sigma} \circ X}{\partial t} - [(\nabla \boldsymbol{u})^T \boldsymbol{\sigma} + \boldsymbol{\sigma}\nabla \boldsymbol{u}] \circ X \right)$$
$$= 2\alpha(\mathbf{D}\boldsymbol{u}) \circ X. \quad (6c)$$

Note that X is the identity for $t = t_r$, and so composition by X simplifies everywhere except in time-derivative terms with a proper choice of t_r (and indeed, a different reference time can and will be used at each step of the numerical method). This new system has the double advantage of being written on a fixed domain Ω_r and of having reduced the material derivatives to partial derivatives in time, it may be discretised as is for instance in refs. [10,11]. In the momentum Eq. (6a), this process has removed any transport term from the equation, however, in the case of the constitutive Eq. (6c), non-elliptic terms remain. In order to go one step further, let us remark as Fortin and Esselaoui [3] that a matrix $\boldsymbol{R}$, such that $\partial \boldsymbol{R} \circ X/\partial t = -(\boldsymbol{R}(\nabla \boldsymbol{u})^T)(X(\boldsymbol{x}, t), t)$ can be seen as a change-of-basis tensor which keeps track of the deformation locally seen by the fluid between the reference time t_r and any time t. This corresponds to the fact that we have $\partial(\boldsymbol{R}\boldsymbol{\sigma}\boldsymbol{R}^T) \circ X/\partial t = (\boldsymbol{R}\overset{\triangledown}{\boldsymbol{\sigma}}\boldsymbol{R}^T) \circ X$.

This allows us to rewrite the problem:

$$\frac{\partial X}{\partial t} = \boldsymbol{u} \circ X \quad (7a)$$

$$\frac{\partial \boldsymbol{R} \circ \boldsymbol{X}}{\partial t} = -(\boldsymbol{R}(\nabla \boldsymbol{u})^{\mathrm{T}}) \circ \boldsymbol{X} \tag{7b}$$

$$Re\frac{\partial \boldsymbol{u} \circ \boldsymbol{X}}{\partial t} = (-\nabla p + (1-\alpha)\nabla \cdot (2\mathbf{D}\boldsymbol{u}) + \nabla \cdot \boldsymbol{\sigma}) \circ \boldsymbol{X} \tag{7c}$$

$$(\nabla \cdot \boldsymbol{u}) \circ \boldsymbol{X} = 0 \tag{7d}$$

$$\boldsymbol{\sigma} \circ \boldsymbol{X} + De\boldsymbol{R}^{-1}\frac{\partial(\boldsymbol{R}\boldsymbol{\sigma}\boldsymbol{R}^{\mathrm{T}}) \circ \boldsymbol{X}}{\partial t}\boldsymbol{R}^{-\mathrm{T}} = 2\alpha(\mathbf{D}\boldsymbol{u}) \circ \boldsymbol{X} \tag{7e}$$

$$\boldsymbol{X}(\boldsymbol{x}_{\mathrm{r}}, t_{\mathrm{r}}) = \boldsymbol{x}_{\mathrm{r}}, \qquad \boldsymbol{R}(t_{\mathrm{r}}) = \boldsymbol{I},$$

$$\boldsymbol{u} \circ \boldsymbol{X}|_{\Gamma_{\mathrm{D}}(t_{\mathrm{r}})} = \boldsymbol{f}_{\mathrm{D}} \circ \boldsymbol{X}, \boldsymbol{\sigma}_{\mathrm{tot}} \circ \boldsymbol{X} \cdot \boldsymbol{n}|_{\Gamma_{\mathrm{N}}(t_{\mathrm{r}})} = \boldsymbol{f}_{\mathrm{N}} \circ \boldsymbol{X}$$

3. Discretisation in time

Let $\{t_n = n\Delta t; n = 0, \ldots, N = T/\Delta t\}$ be a partition of the time interval $[0,T]$. With the choice of t_n for the reference time, for $n = 1, \ldots, N$, we can discretise system (7) in terms of $\Omega^n, \boldsymbol{u}^n, p^n$, $\boldsymbol{\sigma}^n$ the respective approximations of $\boldsymbol{u}, p$ and $\boldsymbol{\sigma}$ at time t_n, taking $u^0 = u_0$ and $\boldsymbol{\sigma}^0 = \boldsymbol{\sigma}_0$:

$$\Omega^n = \Omega^{n-1} + \Delta t\boldsymbol{u}^{n-1}(\Omega^{n-1}) \tag{8}$$

$$\boldsymbol{X}^{n-1}(\boldsymbol{x}) = \boldsymbol{x} - \Delta t\boldsymbol{u}^{n-1}(\boldsymbol{x}) \qquad \forall \boldsymbol{x} \in \Omega^n \tag{9}$$

$$\frac{\boldsymbol{I} - \boldsymbol{R}^n \circ \boldsymbol{X}^{n-1}}{\Delta t} = -(\boldsymbol{R}^{n-1}(\nabla \boldsymbol{u}^{n-1})^{\mathrm{T}}) \circ \boldsymbol{X}^{n-1} \qquad \text{in } \Omega^n \tag{10}$$

$$\frac{\boldsymbol{u}^n - \boldsymbol{u}^{n-1} \circ \boldsymbol{X}^{n-1}}{\Delta t} - (1-\alpha)\nabla \cdot 2\mathbf{D}\boldsymbol{u}^n - \nabla \cdot \boldsymbol{\sigma}^n + \nabla p^n = 0 \tag{11}$$

$$\nabla \cdot \boldsymbol{u}^n = 0 \qquad \text{in } \Omega^n \tag{12}$$

$$\boldsymbol{\sigma}^n + De\frac{\boldsymbol{\sigma}^n - [\boldsymbol{R}^{n-1}\boldsymbol{\sigma}^{n-1}(\boldsymbol{R}^{n-1})^{\mathrm{T}}] \circ \boldsymbol{X}^{n-1}}{\Delta t} = 2\alpha\mathbf{D}\boldsymbol{u}^n \qquad \text{in } \Omega^n \tag{13}$$

This choice of discretisation allows us to define a semi-implicit algorithm as follows.

The first step (8) is the explicit calculation of a first order approximation of the domain $\Omega(t_n)$, using the approximations Ω^{n-1} of $\Omega(t_{n-1})$ and $\boldsymbol{u}^{n-1}$ of $\boldsymbol{u}(t_{n-1})$. The second step (9) is the approximation of $\boldsymbol{X}(t_{n-1})$ for the reference time $t_{\mathrm{r}} = t_n$. It makes explicit the relation between the points of the new domain Ω^n and those of Ω^{n-1}, but is purely a matter of notation here, since step (8) already allows us to keep track of the trajectories of material points (namely, for a space-discretised algorithm, the image $\boldsymbol{X}^{n-1}(\boldsymbol{x}_i^n)$ of node i of the mesh of Ω^n will be by construction $\boldsymbol{x}_i^{n-1}$, the node i of the mesh of Ω^{n-1}).

The third step (10) is the discretisation of Eq. (7b), where we have used that $\boldsymbol{R}^n = \boldsymbol{I}$ because t_n is the reference time. Up to the first order in Δt, it can be inverted so as to provide an explicit formula for $\boldsymbol{R}^{n-1} \circ \boldsymbol{X}^{n-1}$, in terms of the velocity at the previous time-step and $\boldsymbol{X}^{n-1}$:

$$\boldsymbol{R}^{n-1} \circ \boldsymbol{X}^{n-1} = \boldsymbol{I} + \Delta t(\nabla \boldsymbol{u}^{n-1})^{\mathrm{T}} \circ \boldsymbol{X}^{n-1}. \tag{14}$$

The fourth step is the resolution of system (11) and (12), which is a generalised Stokes type problem for unknowns $\boldsymbol{u}^n$ and p^n. The choice of discretisation involves a term in $\boldsymbol{\sigma}^n$, which makes the scheme more implicit, but is an unknown at this stage. However, Eq. (13) gives an explicit formula for $\boldsymbol{\sigma}^n$ in terms of already calculated quantities and of $\boldsymbol{u}^n$:

$$\boldsymbol{\sigma}^n + \beta[\boldsymbol{R}^{n-1}\boldsymbol{\sigma}^{n-1}(\boldsymbol{R}^{n-1})^{\mathrm{T}}] \circ \boldsymbol{X}^{n-1} + 2\alpha(1-\beta)\mathbf{D}\boldsymbol{u}^n, \tag{15}$$

where

$$\beta = \frac{De}{De + \Delta t}. \tag{16}$$

Thus, the same technique as in ref. [12] can be used to rewrite (11) as

$$\frac{\boldsymbol{u}^n - \boldsymbol{u}^{n-1} \circ \boldsymbol{X}^{n-1}}{\Delta t} - (1-\alpha\beta)\nabla \cdot 2\mathbf{D}\boldsymbol{u}^n$$
$$- \beta\nabla \cdot [\boldsymbol{R}^{n-1}\boldsymbol{\sigma}^{n-1}(\boldsymbol{R}^{n-1})^{\mathrm{T}}] \circ \boldsymbol{X}^{n-1} + \nabla p^n = 0 \qquad \text{in } \Omega^n, \tag{17}$$

and the system composed of (17) and (12) can be solved for $\boldsymbol{u}^n$ and p^n in terms of known quantities.

Finally, the fifth step is the explicit calculation of $\boldsymbol{\sigma}^n$ through Eq. (15), using $\boldsymbol{u}^n$.

4. Discretisation in space and resolution method

We will now construct a finite element approximation of the sequence $(\Omega^n, \boldsymbol{u}^n, p^n, \boldsymbol{\sigma}^n)$. This construction is given for $\Omega(t) \in \mathbb{R}^2$ for the sake of simplicity, but extension to three dimensions is straightforward. Let $\mathcal{T}_h^0$ be a finite element triangular mesh of Ω_h^0, an approximation of Ω^0. Meshes $\mathcal{T}_h^n$ and approximate domains Ω^n will be obtained in the course of the algorithm, and we use finite element spaces based on these triangulation, which are continuous and quadratic for the velocity, continuous and linear for the pressure, and discontinuous and linear for the extra-stress tensor (see Fig. 1). Additionally, we require the velocity to satisfy the boundary condition on the solid boundary, the pressure to have zero mean and the stress tensor to be symmetric.

The sequence of approximated domains Ω^n and corresponding meshes $\mathcal{T}_h^n$ and finite element spaces can then be defined by induction: define mesh $\mathcal{T}_h^n$ as mesh $\mathcal{T}_h^{n-1}$ advected by the (continuous) velocity field $\boldsymbol{u}_h^{n-1}$. Then Ω_h^n is the domain described by $\mathcal{T}_h^n$. A quadratic approximation of the velocity allows the possibility for the mesh to have curved boundaries (isoparametric mesh of degree 2). However, a simpler implementation has been used where only the vertices of the triangles are advected, keeping straight-sided triangles.

This of course results in a suboptimal convergence rate, as the boundary of the domain is only represented by a piecewise linear approximation while the interior approximation is of second order. The convergence rate for Stokes problem with this combination is known to be 3/2 in H^1 norm, see, e.g. ref. [13]
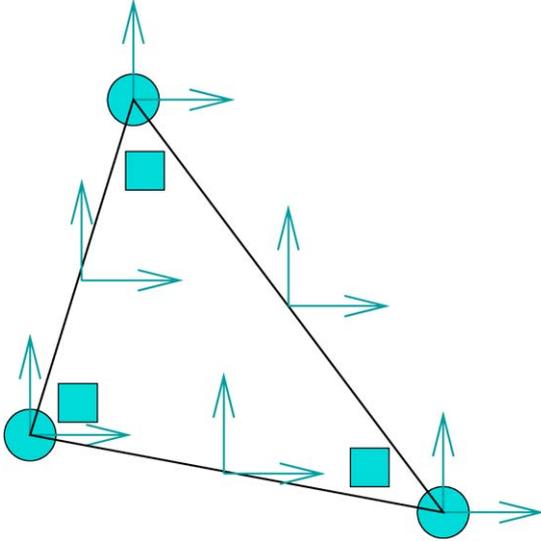
Fig. 1. The finite element approximation. Arrows, velocity components (P_2–C^0); circles, pressure (P_1–C^0); squares, extra-stress (P_1–C^{-1}).

for this result and numerical tests. The present algorithm also achieves a 3/2 convergence rate (see Fig. 2) for the calculation of the drag force on a cylinder in a Newtonian confined channel flow, in the setup presented in Fig. 4.

Let us now give the final algorithm in terms of the variational formulation of the problem in the finite element spaces at time t^n:

Algorithm.

Step 1: Let $\mathcal{T}_h^n$ be a triangulation with the same connectivity of $\mathcal{T}_h^{n-1}$ and vertices $x_i^n = x_i^{n-1} + \Delta t\, u_h^{n-1}(x_i^{n-1})$ for all $x_i^{n-1} \in \mathcal{T}_h^{n-1}$.

Step 2: Calculate explicitly:

$$R_h^{n-1}(x_i^{n-1}) = I + \Delta t(\nabla u_h^{n-1})^{\mathrm{T}}(x_i^{n-1}). \tag{18a}$$

Fig. 2. Relative error in the drag on a cylinder in a channel in Stokes flow as a function of spatial resolution h. The straight line gives the slope $h^{3/2}$. The reference drag value used is the one obtained by Hulsen et al. [22].

Step 3: Calculate explicitly:

$$\tilde{\sigma}_h^{n-1}(x_i^n) = [R_h^{n-1}\sigma_h^{n-1}(R_h^{n-1})^{\mathrm{T}}](x_i^{n-1}), \tag{18b}$$

$$\tilde{u}_h^{n-1}(x_i^n) = u_h^{n-1}(x_i^{n-1}). \tag{18c}$$

Step 4: Solve the (linear) generalised Stokes problem: find u_h and p_h such that:

$$\left(\frac{u_h^{n-1} - \tilde{u}_h^{n-1}}{\Delta t}, v_h\right) + (1 - \alpha\beta)(2\mathbf{D}u_h^n, \mathbf{D}v_h)$$

$$- \int_{\Gamma_{\mathrm{N}}} f_{\mathrm{N}} \cdot v_h \, \mathrm{d}s + \beta(\tilde{\sigma}_h^{n-1}, \mathbf{D}v_h)$$

$$- (p_h^n, \nabla \cdot v_h) = 0, \tag{18d}$$

$$(\nabla \cdot u_h^n, q_h) = 0, \tag{18e}$$

for all v_h and q_h in the finite element spaces for velocity and pressure.

Step 5: Calculate explicitly $\sigma_h^n = \beta\tilde{\sigma}_h^{n-1} + 2\alpha(1 - \beta)\mathbf{D}u_h^n$.

Note that the operations in Eqs. (18a) and (18b) are merely scalar algebra on node-values of the fields, and do not involve any interpolation or cross-node communication. If the triangle vertices only are advected, interpolation is needed for midpoint nodal values in (18c). We use an augmented Lagrangian technique with a Uzawa iterative algorithm for the resolution of the Stokes-like problem in Step 4, as in ref. [14], which allows us to enforce exactly a zero-divergence of the velocity.

In the case of fixed, polygonal boundaries, the numerical analysis of this algorithm was made by Bensaada et al. in a recent article [15]. In particular, they show that optimal error bounds hold for the L^2 error on the extra-stress tensor and the H^1 error on the velocity, that is, the error tends to zero as $\Delta t + h^2$ subject to the condition that as h tends to zero, Δt tends to zero as $\Delta t \leq Ch^{d/2}$ for some arbitrary constant C and d the dimension of space containing the computational domain. Note that this condition is not a stability condition.

5. Case of free-surface flows

In the case of a flow with a free-surface $\Gamma_{\mathrm{F}}(t)$, the boundary condition at $\Gamma_{\mathrm{F}}(t)$ is set by the pressure jump across the surface, between the outer medium (supposedly of uniform pressure p_0) and the fluid. This pressure jump is due to surface tension, and can be expressed as

$$\sigma_{\mathrm{tot}}n_{|\Gamma_{\mathrm{F}}(t)} = \left(p_0 + \frac{\kappa}{\mathrm{Ca}}\right)n, \tag{19}$$

where κ is the local total curvature of the boundary $\Gamma_{\mathrm{F}}(t)$ and $\mathrm{Ca} = (\eta_{\mathrm{s}} + \eta_{\mathrm{p}})U/\gamma$ is the capillary number, based on the surface tension γ. In the discrete case, this raises the question of what the normal is at nodes of the boundary, where our approximation of the free-surface has corners. We choose to calculate a cube spline passing through all nodes on the free boundary, and then
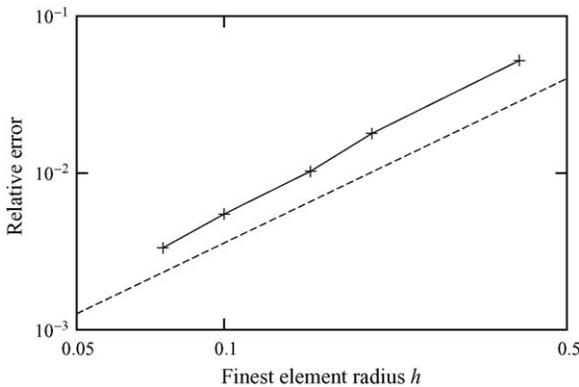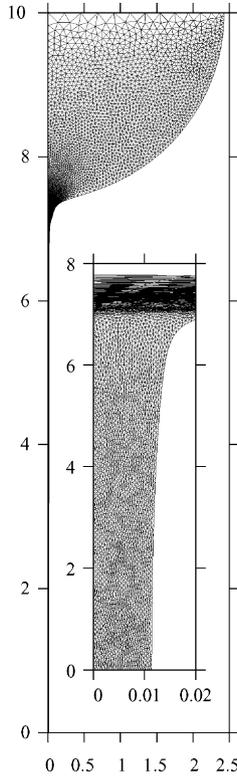
Fig. 3. Anisotropic mesh calculated for the beads-on-string problem (Section 8). The detail plot shows the triangles are adapted to the aspect ratio of the features of the fluid domain.

use this spline function to define the normal at nodes and to calculate the curvature, see, e.g. ref. [16].

As the domain deforms, the mesh deteriorates because of element stretching, which in turn affects the quality of the solution. Two strategies are possible, namely local re-arrangement of the mesh at each time-step [4] or occasional complete remeshing [2]. Both present the inconvenience that the solutions need to be interpolated onto the new mesh, an operation in which accuracy is not controlled and that may be unstable. Indeed, even if only edge-swaps are performed, the extra-stresses are in some sense interpolated, as the shape functions on which they are defined change, even if the node location and nodal values are kept the same. Thus, this operation is not altogether neutral on the solution, an equilibrium solution with one mesh will no longer be in equilibrium if edges are swapped. In the present case, the material undergoes very large deformations (see, e.g. Fig. 11), which means that large changes of connectivity in the mesh are necessary. In addition, the local aspect ratio of the final shape is extreme, and only anisotropic meshing can allow us to describe these geometries with a reasonably sized mesh. Thus, we opt for completely remeshing the domain at large time intervals, so as to reduce the number of interpolations,[1] which allows us to obtain a mesh well-adapted to the current domain shape.

An example of anisotropic mesh is shown in Fig. 3. Elements have an aspect ratio adapted to the local aspect ratio of the domain. This allows us to save a large number of nodes, and is more adapted to the solution we wish to calculate, as spanwise variations are much higher than variations along the stretched directions [17]. Thus, a metric depending on the local aspect ratio is calculated which prescribes the mesh step at every point of the domain in all directions through a matrix representation, the mesh generation itself being delegated to the free software BAMG [18].

6. Case of inflow and outflow boundaries

In the case when the fluid is flowing through part of the boundary, the mesh cannot be advected by the material velocity (at least in the neighbourhood of these boundaries), and the scheme has to be modified accordingly. A natural extension of the scheme proposed above is to use the Arbitrary Lagrangian–Eulerian method for mesh advection and the method of Lagrange–Galerkin (also called weak form of the method of characteristics) to account for the difference of mesh and material point advection. Let us introduce a partition of the boundary as $\partial\Omega = \Gamma_I \cup \Gamma_F \cup \Gamma_M(t)$, where Γ_I is the part of the boundary through which fluid is flowing, Γ_F the fixed part of the boundary and $\Gamma_M(t)$ its moving part. For the sake of simplicity, Γ_I and $\Gamma_M(t)$ are supposed to be non-adjacent.[2]

Let us introduce an arbitrary advection field $\boldsymbol{u}_a$, and the associated trajectories $t \mapsto A(\boldsymbol{x}_r, t)$ of points of reference coordinate $\boldsymbol{x}_r \in \Omega_r$,

$$\begin{cases} \dfrac{\partial A}{\partial t}(\boldsymbol{x}_r, t) = \boldsymbol{u}_a(A(\boldsymbol{x}_r, t), t), & t \in [0, T], \\ A(\boldsymbol{x}_r, t_r) = \boldsymbol{x}_r \end{cases}$$

This advection field will be used instead of the physical velocity $\boldsymbol{u}$ to advect the mesh vertices. For clarity, mesh vertices will be denoted $\boldsymbol{a}_i$ instead of $\boldsymbol{x}_i$ in the sequel, since their trajectories are now the curves $t \mapsto A(\boldsymbol{x}_r, t)$ instead of the curves $t \mapsto X(\boldsymbol{x}_r, t)$. In order to have $\Gamma_M(t) = A(\Gamma_M(t_r), t)$, a necessary and sufficient condition is that $\boldsymbol{u}_a \cdot \boldsymbol{n}_{|\Gamma_M(t)} = \boldsymbol{u} \cdot \boldsymbol{n}_{|\Gamma_M(t)}$. If in addition $\boldsymbol{u}_a$ vanishes on Γ_I, and is tangent to Γ_F, we obtain that $\Omega(t) = A(\Omega(t_r), t)$ for any arbitrary choice of $\boldsymbol{u}_a$ in the interior of the domain. However, due to the difference between $\boldsymbol{u}_a$ and $\boldsymbol{u}$, the material derivative now becomes

$$\left(\frac{\partial}{\partial t} + \boldsymbol{u} \cdot \nabla\right) \boldsymbol{u} \circ A = \frac{\partial \boldsymbol{u} \circ A}{\partial t} + (\boldsymbol{u} - \boldsymbol{u}_a) \cdot \nabla \boldsymbol{u} \circ A$$

$$= \frac{\partial \boldsymbol{u} \circ A \circ \tilde{X}}{\partial t} \circ \tilde{X}^{-1}, \tag{20}$$

with $\tilde{X}$ such that $A \circ \tilde{X} = X$. The method of characteristics [19] can thus be used to discretise this term. Note that $\tilde{X}(\Omega_r, t)$ is in general not included in $\Omega(t)$, e.g. if there is inflow at $\boldsymbol{x}_r$, then at earlier times the advected point $\tilde{X}(\boldsymbol{x}_r, t)$ will not have arrived in the domain. This illustrates the hyperbolic nature of the transport

---

[1] Another reason for the common choice of local re-arrangement of the mesh is that interpolation between unstructured meshes is a difficult operation from an algorithmic point of view. However, the localisation technique developed for the method of characteristics (detailed in the end of Section 6) allows us to do this with a high cost-efficiency.

[2] Adjacent inflow and moving boundaries can be dealt with by considering separately the directions of advection.

term, which requires upwind information. Since there will nearly always be streamwise variations of the extra-stress upstream of the inflow boundary, it is necessary to provide an analytic expression (or tabulated values) for the extra-stress of the incoming fluid for all points that reach or cross the inflow boundary in a time Δt. However, in the case of a Poiseuille inflow, there is no streamwise variation of the extra-stress in the neighbourhood of the inflow boundary, so if $\tilde{X}(\boldsymbol{x}_r, \tau)$ is before the boundary, the value of $\sigma_{ij}(\tilde{X}(\boldsymbol{x}_r, \tau))$ can be taken to be $\sigma_{ij}(\tilde{X}(\boldsymbol{x}_r, \tau'))$ with τ' such that $\tau < \tau' < t_r$ and $\tilde{X}(\boldsymbol{x}_r, \tau') \in \Gamma_I$ (and this point is easily determined, since trajectories are straight lines close to the inflow boundary).

Thus, the modifications in Algorithm (18) consist of replacing Step 1 by the three following steps:

Step 1a: Determine some mesh advection field $\boldsymbol{u}_{a,h}$ within the above conditions.
Step 1b: Let $\mathcal{T}_h^n$ be an isoparametric triangulation of degree k, with the connectivity of $\mathcal{T}_h^{n-1}$ and nodes $\boldsymbol{a}_i^n = \boldsymbol{a}_i^{n-1} + \Delta t \boldsymbol{u}_{a,h}^{n-1}(\boldsymbol{a}_i^{n-1})$.
Step 1c: Calculate the characteristic feet [approximation of $X(\boldsymbol{a}_i^n, t^{n-1})$] with: $\boldsymbol{x}_i^{n-1} = \boldsymbol{a}_i^{n-1} + \Delta t (\boldsymbol{u}_{a,h}^{n-1} - \boldsymbol{u}_h^{n-1})(\boldsymbol{a}_i^{n-1})$.

Steps 2–5 are only changed in that velocity-advected nodes $\boldsymbol{x}_i^n$ are replaced everywhere by the newly defined $\boldsymbol{a}_i^n$ nodes, and in that the points $\boldsymbol{x}_i^{n-1}$ are generally not nodes of the mesh $\mathcal{T}_h^{n-1}$: thus the operations in Eqs. (18b) and (18c) are now interpolations except when $\boldsymbol{u}_{a,h}^{n-1} - \boldsymbol{u}_h^{n-1}$ vanishes. It is thus interesting, wherever possible, to have $\boldsymbol{u}_a = \boldsymbol{u}$ and to localise areas where interpolations take place only in the vicinity of inflow or outflow boundaries, where the extra stress tensor is a smooth function.

In order to interpolate the finite element fields at points $\boldsymbol{x}_i^{n-1}$, it is necessary to identify the mesh element to which they belong. This localisation problem is not straightforward for an unstructured, locally refined mesh and an efficient procedure needs to be developed. We use an approach known as quadtree, in which the rectangular bounding box of the domain is divided recursively in regular rectangular subgrids, which are in turn divided in the same way until they cover a fixed number of triangles [20]. The point $\boldsymbol{x}_i^{n-1}$ can efficiently be localised in the quadtree, and then only a small, fixed number of triangles remain to be examined. With this algorithm, the cost of localising any point in the mesh is logarithmic in terms of the total number of elements in the mesh. In practice, the computational cost of this implementation of the method of characteristics is found to be much less than the cost of solving a 2D Stokes problem on the same mesh. This implementation is embedded as part of the open-source free software RHEOLEF, which provides a general purpose C++ finite elements library [21].

7. Validation of the approach for a fixed boundary problem

The algorithm (18) developed in this paper is tailored for free-surface flow problems, and is also well adapted for transient flows, especially when convection is dominant. There is no question that it is not very well-suited for dealing with fixed-boundary flows; nevertheless, we want to show briefly that algorithm (18) can cope with fixed boundaries up to reasonably realistic Deborah numbers, so that it is an interesting approach for problems involving both a free-surface and some fixed boundaries: thus we present here calculations of a well-studied, stationary problem for which a proper comparison with results from the literature is possible.

In this test, we consider the flow of an Oldroyd-B fluid in a plane channel of infinite spanwise dimension, blocked by an infinitely long cylinder disposed in the spanwise direction, at equal distance of the channel walls and of diameter half the channel width. The flow far from the cylinder is supposed to be undisturbed established flow, perpendicular to the cylinder axis and parallel to the walls. The fixed computational domain Ω is as described in Fig. 4, the boundary conditions for the velocity being no-slip boundary conditions $\boldsymbol{u} = \boldsymbol{0}$ at the wall boundary Γ_D and cylinder Γ_C, symmetry conditions $u_y = 0$ and $\partial u_x / \partial y = 0$ at the axis Γ_A. At the inflow and outflow boundary, an established Poiseuille flow is imposed $u_x = 3U(R^2 - y^2)/2R^2$, $u_y = 0$, and for the inflow boundary only, the corresponding analytical solution of the Oldroyd-B constitutive law is used for providing inflow conditions on $\boldsymbol{\sigma}$. A time-marching approach is used, that is, we start from an artificial initial condition and let the system evolve a permanent flow. The initial condition for a given De number was taken as the steady solution obtained in a previous simulation for a lower De number, and for a Newtonian fluid for the first simulation.

In order to compare with published numerical simulations results, we take the flow parameters as $Re = 0$ and $\alpha = 0.41$. The comparison is based on the value of the non-dimensional drag force C_d exerted by the fluid on the cylinder,

$$C_d = \boldsymbol{e}_x \cdot \int_{\Gamma_C} \boldsymbol{\sigma}_{tot} n_{\Gamma_C} \, \mathrm{d}s$$
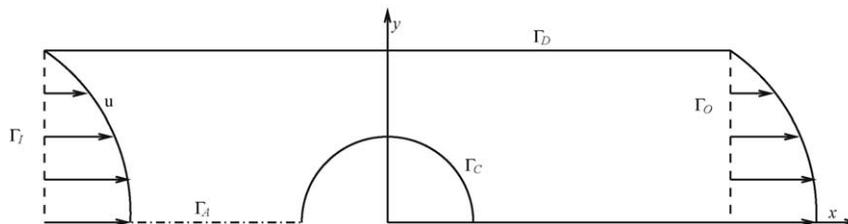
Fig. 4. Domain and boundary conditions for the problem of the cylinder in a channel.

Table 1
Drag coefficient C_d as a function of the Deborah number De

De	Alves et al. [31]	Hulsen et al. [22]	Dou and Phan-Thien [32]	Algorithm (18)
0		132.36	131.81	132.33
0.01	132.34			132.31
0.025	132.21			132.16
0.05	131.79			131.75
0.1	130.34	130.36	129.72	130.22
0.2	126.62	126.63	126.41	126.53
0.3	123.20	123.19	123.52	123.41
0.4	120.60	120.60	121.56	
0.5	118.83	118.84	120.58	

It was considered that convergence had been reached when all three conditions were met:

$$\frac{||\boldsymbol{u}_h^{n+1} - \boldsymbol{u}_h^n||_{L^2}}{||\boldsymbol{u}_h^{n+1}||_{L^2}} < 10^{-6},$$

$$\frac{||\boldsymbol{\sigma}_h^{n+1} - \boldsymbol{\sigma}_h^n||_{L^2}}{||\boldsymbol{\sigma}_h^{n+1}||_{L^2}} < 10^{-5},$$

$$\frac{C_{d,h}^{n+1} - C_{d,h}^n}{C_{d,h}^{n+1}} < 10^{-6}.$$

The only purpose of these simulations being to validate our numerical approach, a single mesh has been used. The mesh is fixed: the advection field of the mesh $\boldsymbol{u}_a = 0$ everywhere. For $De \leq 0.3$ and on this mesh, the method reaches convergence; however, for higher De convergence was not reached. As the method is primarily designed for moving boundary problems rather than flows past obstacles, this range of convergence is satisfactory. Other works in literature obtain results for up to $De = 1.8$ [22], however, even in recent papers, authors disagree on the quantitative results from $De = 0.5$, and there is no evidence that solutions exist for large Deborah number flows of Oldroyd-B fluids past obstacles [23]. It should also be noted that, due to high shear in the boundary layer close to the cylinder, the actual Deborah number characteristic of this boundary layer is actually approximately 10 times larger. Throughout the range of convergence, algorithm (18) gives results, which compare well with results from the literature, see Table 1 and Fig. 5. Fig. 6 shows the three components of the extra-stress tensor around the cylinder.

Although the simulation of this type of flow is not a strong point of algorithm (18), it is shown to be sufficiently robust to simulate viscoelastic flow in a classical test with fixed boundaries, and the results compare well with those given in the literature.

8. The beads-on-string problem

We consider the flow of a viscoelastic fluid, which initially forms an infinite cylindrical column of constant radius R of fully relaxed fluid at rest. At time $t = 0$, a small-amplitude disturbance of period L is applied to the free-surface, which will trigger the collapse of the column. These conditions are known to produce
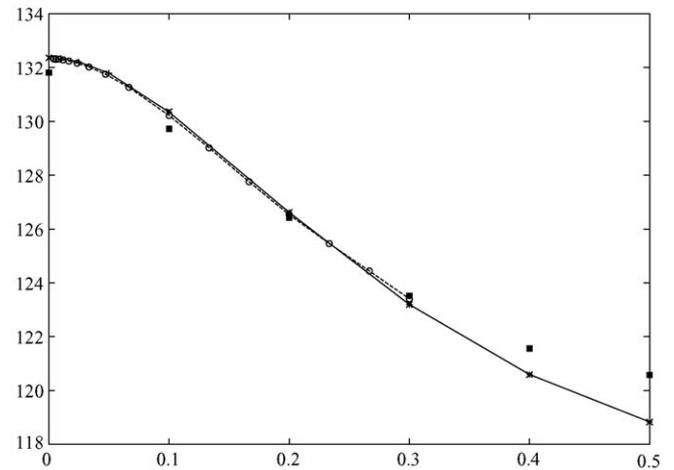
Fig. 5. Drag coefficient C_d as a function of the Deborah number De. (+) Alves et al. [31]; (×) Hulsen et al. [22]; (■) Dou and Phan-Thien [32]; (○) present algorithm.

the well-known beads-on-string structure, where spherical drops form at intervals along the liquid column and remain connected by threads of uniform radius before the final break-up occurs when polymers in the threads are fully stretched.

This phenomenon may be described by the Oldroyd-B model, as the balance involved in the thread-thinning regime is between capillary forces and the build-up of elastic forces. This model only fails to predict the later stages of the flow, when polymers reach their maximal stretching. Analytical investigations of slender cylinders of Oldroyd-B fluids [5–7] have determined the rate of thinning of these cylinders, namely

$$r = r_0 \exp\left(-\frac{t}{3De}\right), \tag{21}$$
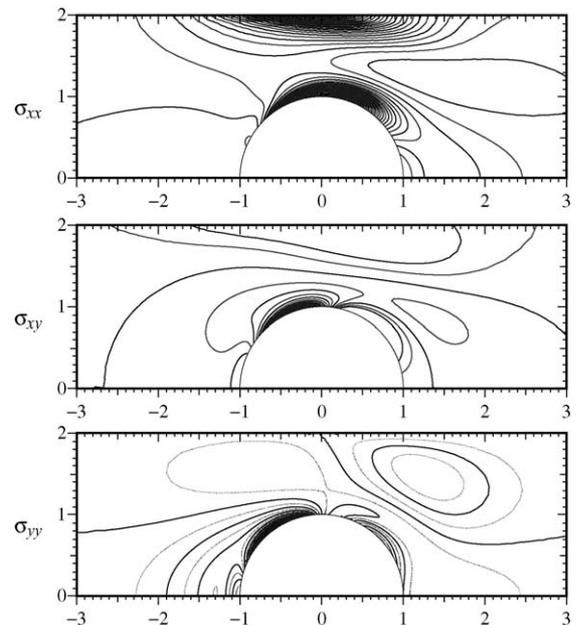
Fig. 6. Isocontours of the extra-stress tensor components in permanent flow around a cylinder in a closed plane channel, for $De = 0.2$. Solid lines are at 0.5 non-dimensional units interval, dotted lines (for σ_{yy} only) at 0.25 interval.
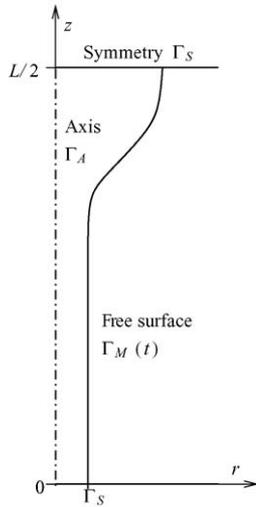
Fig. 7. Domain and boundary conditions for the beads-on-string simulations.

$t = 2De/3 \simeq 63.2$ $t = De \simeq 94.9$ $t = 3De/2 \simeq 142.4$

Fig. 8. Collapse of the column for $Oh = 3.16$, $De = 94.9$; $t = 2De/3 \simeq 63.2$, $t = De \simeq 94.9$, $t = 3De/2 \simeq 142.4$.

a prediction which has been confirmed by experiments [24,25]. It has also been obtained in numerical simulations of this problem using one-dimensional models based on a slender cylinder approximation [9,26,27], but to the best of our knowledge was never simulated in its original variables.

The flow presents an axial symmetry, thus simulations can be reduced to two dimensions only. It is also periodic in the z direction with symmetry planes at $L/2$ intervals. The computational domain is shown in Fig. 7. The flow is governed by Eqs. (1), (2), (4) with boundary conditions as follows:

$$u_z = 0, \qquad \sigma_{\text{tot}} e_z = 0, \qquad \text{on } \Gamma_S, \tag{22}$$

$$u_r = 0, \qquad \sigma_{\text{tot}} e_r = 0, \qquad \text{on } \Gamma_A, \tag{23}$$

$$\sigma_{\text{tot}} n = \frac{1}{\text{Ca}} \kappa n, \qquad \text{on } \Gamma_M(t). \tag{24}$$

Since there is no fixed boundary in this problem, the purely. Lagrangian approach is well adapted to it. The mesh is thus directly advected with the velocity field.

The capillary time-scale of the problem is $T = \sqrt{R^3 \varrho / \gamma}$ and yields a characteristic velocity $U = R/T$. With this choice of scaling, the inverse of the Reynolds number and capillary number take both the same value $(\eta_s + \eta_p)/\sqrt{\varrho R \gamma}$, which is called the Ohnesorge number Oh. The Ohnesorge and Deborah numbers, along with the ratio of polymeric viscosity α are thus the only parameters of the study. In order to match the conditions of previous one-dimensional approaches, we choose the ratio of polymer viscosity to total viscosity $\alpha = 3/4$ and consider an initial disturbance of amplitude εR, with $\varepsilon = 10^{-3}$, over a period $L = 20R$, so that the initial non-dimensional radius is $r(z, t_0) = 1 + \varepsilon \cos(2\pi z/20)$. Let us compare the flow obtained with a small surface tension, $Oh = 3.16$ and $De = 94.9$ and with a 10 times larger surface tension, $Oh = 1$ and $De = 300$. Fig. 8 presents the time evolution of the free-surface for $Oh = 3.16$, $De = 94.9$. At first the jet has an essentially Newtonian behaviour, since the polymers are initially fully relaxed, but as the column radius is reduced elastic stresses build up and lead to the clearly recognisable bead-on-string structure, with connecting threads
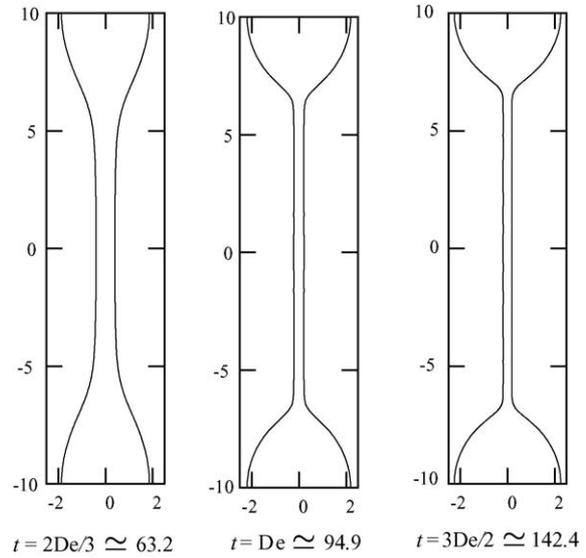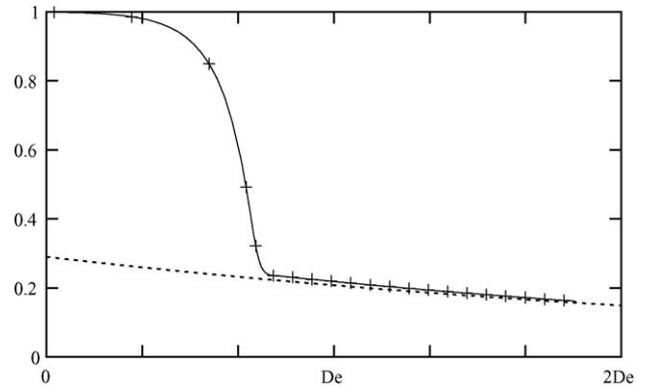
Fig. 9. The inertial and elastic pinching regimes for $Oh = 3.16$, $De = 94.9$, $L = 10$ vs. non-dimensional time. (+) Minimum radius of the liquid bridge; (- - -) analytical solution $r_0 \exp(-t/3De)$ with prefactor $r_0 = 0.29$.

of uniform radius. This radius is plotted against time in Fig. 9, and compared to the analytical law with good agreement, the coefficient being found as $r_0 = 0.29$. There is also a very good agreement between the present simulations and previous one-dimensional simulations by Li and Fontelos [28], as shown in Fig. 10. In addition to a cross-validation of our results, this indicates that the one-dimensional model does retain the essentials of the mechanism of formation of the beads-on-string structure. This is important, as 1D calculations are much faster than 2D ones and thus allow one to explore a wider range of parameters.

Now if we increase the role of surface tension by a factor 10 (Fig. 11, $Oh = 1$, $De = 300$), the fluid has relatively more inertia during the initial stages of the flow, when the elastic forces are

Fig. 10. Comparison between 1D calculations (dashed lines) [28] and 2D calculations (solid lines) at $t = 4De/3$.
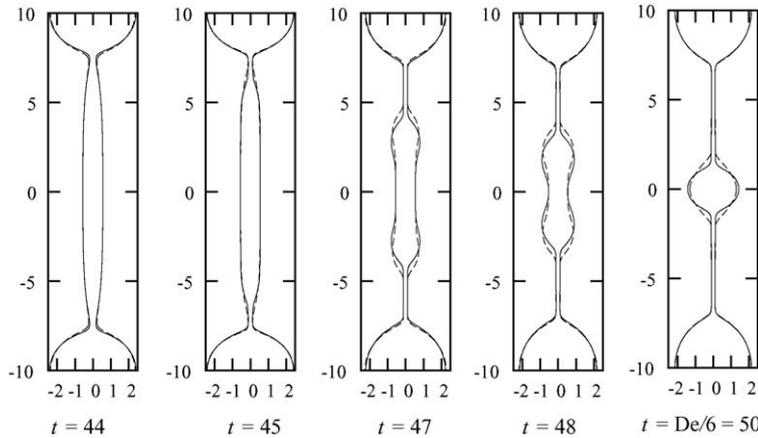
Fig. 11. Collapse of the column for $Oh = 1$, $De = 300$. Solid lines, at non-dimensional time t as indicated; dashed lines, at time $t - \delta t$ with $\delta t = 0.5$, are included to exhibit the dynamics of the process.
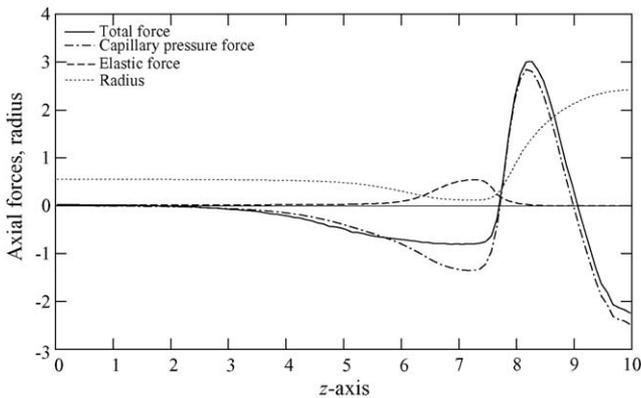
Fig. 12. Axial tension force (non-dimensional) in the fluid column for $Oh = 1$, $De = 300$ at $t = 45$ relative to $z = 0$. The total axial force at point z is $\int_{0 < x_z < z} \nabla \cdot \sigma_{\text{tot}} \, d\mathbf{x}$, the elastic force is $\int_r \int_\theta \sigma_{zz} \, d\mathbf{x}$ and the capillary pressure force, $2\pi(r(z) - r(0)/\text{Ca}) - \int_r \int_\theta [p - p(z = 0)] \, d\mathbf{x}$.

negligible; and the relatively lower viscosity is not sufficient to slow the capillary driven flow. In addition, the elastic reaction is slow, so that the effect of capillary pressure dominates and sharp necking regions form close to the drops, at each end of bulgy-shaped bridges, which is typical of the pinching of Newtonian jets. However, as the stretching increases in these necks, the elastic stress eventually builds up and slows down the necking, as shown in Fig. 12. It should be noted, however, that the capillary pressure remains dominant everywhere else. It is also seen on this figure that, due to capillary pressure, a net total force exists between the neck and the liquid bridge, which will quickly chase the fluid from the bridge into a central droplet, still connected to the main drops by a thread. This phenomenon, referred to as recoil, had been observed in 1D numerical simulations [26,28].

9. Conclusions

In this paper, we present an approach for free-surface flows of viscoelastic fluids in which the Lagrangian point of view is extensively used in order to describe the free-surface displacement and the material and objective derivatives. This method

is thus highly specialised for this type of flow, but through the Arbitrary Lagrangian–Eulerian technique, it can also handle problems in an Eulerian setup, and it is shown that it manages reasonably well with the difficult problem of drag on a cylinder.

In the case of free-surface flows, the method allows us to investigate problems with extremely large deformations and elastic stresses, and is validated by comparison with an analytic result (which has itself been verified experimentally). Anisotropic meshing allows us to deal with geometries where local aspect ratios are highly inhomogeneous.

Finally, the Lagrangian discretisation is intrinsically well suited for transport terms: it is known to have good properties when the material derivative dominates (see, e.g. ref. [29] for an example of a high Reynolds Newtonian flow), and also it is appropriate for free-surface viscoelastic flow. This was demonstrated for instance by Rasmussen and Hassager [30], who carry out three-dimensional simulations of the creeping flow of an upper-convected Maxwell fluid described by an integral model in a setup similar to ours. The high Deborah number free-surface calculations presented here confirm that this is also true for inertial flow of materials described by a differential constitutive equation. The Lagrangian–Eulerian approach presented here is thus an interesting technique for problems, such as the break-up of high speed jets of viscoelastic material, in which solid boundaries have little effect but where very large deformations occur, with beads-on-string structures appearing, which is an important issue for industry, for instance in inkjet printing.

References

[1] H.K. Rasmussen, O. Hassager, Simulation of transient viscoelastic flow, J. Non-Newtonian Fluid Mech. 46 (1993) 289–305.
[2] H.K. Rasmussen, Time-dependent finite-element method for the simulation of three-dimensional viscoelastic flow with integral models, J. Non-Newtonian Fluid Mech. 84 (1999) 217–232.

[3] M. Fortin, D. Esselaoui, A finite-element procedure for viscoelastic flows, Int. J. Numer. Methods Fluids 7 (10) (1987) 1035–1052.

[4] O.G. Harlen, J.M. Rallison, P. Szabo, A split Lagrangian–Eulerian method for simulating transient viscoelastic flows, J. Non-Newtonian Fluid Mech. 60 (1995) 81–104.

[5] A.V. Bazilevskii, S.I. Voronkov, V.M. Entov, A.N. Roshkov, Orientational effects in the decomposition of streams and strands of diluted polymer solutions, Sov. Phys. DokL 26 (1981) 333–335.

[6] M. Renardy, A numerical study of the asymptotic evolution and breakup of Newtonian and viscoelastic jets., J. Non-Newtonian Fluid Mech. 59 (2–3) (1995) 267–282.

[7] V.M. Entov, E.J. Hinch, Effect of a spectrum of relaxation times on the capillary thinning of a filament of elastic liquid, J. Non-Newtonian Fluid Mech. 72 (1997) 31–53.

[8] R. Keunings, An algorithm for the simulation of transient viscoelastic flows with free surfaces, J. Comput. Phys. 62 (1986) 199–220.

[9] D.W. Bousfield, R. Keunings, G. Marucci, M.M. Denn, Nonlinear analysis of the surface tension driven breakup of viscoelastic filaments, J. Non-Newtonian Fluid Mech. 21 (1986) 79–97.

[10] F.G. Basombrio, Flows of viscoelastic fluids treated by the method of characteristics, J. Non-Newtonian Fluid Mech. 39 (1991) 17–54.

[11] P. Wapperom, R. Keunings, V. Legat, The backward-tracking lagrangian particle method for transient viscoelastic flows, J. Non-Newtonian Fluid Mech. 91 (2000) 273–295.

[12] P. Saramito, Efficient simulation of nonlinear viscoelastic fluid flows, J. Non Newtonian Fluid Mech. 60 (1995) 199–223.

[13] E. Bansch, K. Deckelnick, Optimal error estimates for the stokes and Navier–Stokes equations with slip-boundary condition, M2AN 33 (5) (1999) 923–938.

[14] N. Roquet, P. Saramito, An adaptive finite element method for Bingham fluid flows around a cylinder, Corn-put, Methods Appl. Mech. Eng. 192 (2003) 3317–3341.

[15] M. Bensaada, D. Esselaoui, P. Saramito, Approximation of time-dependent viscoelatic fluid flows with the Lagrange–Galerkin method, Numer. Math., submitted for publication.

[16] J. Li, M. Hesse, J. Ziegler, A.W. Woods, An Arbitrary Lagrangian–Eulerian method for moving-boundary problems and its application to jumping over water, J. Comp. Phys. 308 (2005) 289–314.

[17] H. Borouchaki, P.L. George, F. Hecht, P. Laug, E. Saltel, Delaunay mesh generation governed by metric specifications. Part I: algorithms, Finite Elem. Anal. Des. 25 (1997) 61–83.

[18] F. Hecht, bamg: bidimensional anisotropic mesh generator, Technical report, INRIA, Rocquencourt, France, 1997 (http://www-rocql.inria.fr/gamma/cdrom/www/bamg).

[19] O. Pironneau, Finite Elements Methods for Fluids, John Wiley & Sons, Chichester, 1989.

[20] J. Étienne, Simulation numérique d'écoulements gravitaires à fortes différences de densité, Application aux avalanches, Ph.D. thesis, INP Grenoble, September 2004.

[21] P. Saramito, N. Roquet, J. Étienne, rheolef user's manual, Technical report, LMC-IMAG, Grenoble, France, 2003 (http://www-lmc.imag.fr/lmc-edp/Pierre.Saramito/rheolef).

[22] M.A. Hulsen, R. Fattalb, R. Kupfermanc, Flow of viscoelastic fluids past a cylinder at high weissenberg number: stabilized simulations using matrix logarithms, J. Non-Newtonian Fluid Mech. 127 (2005) 27–39.

[23] Y. Fan, Limiting behavior of the solutions of a falling sphere in a tube filled with viscoelastic fluids, J. Non-Newtonian Fluid Mech. 110 (2–3) (2003) 77–102.

[24] R.F. Liang, M.R. Mackley, Rheological characterization of the time and strain dependence for polyisobutylene solutions, J. Non-Newtonian Fluid Mech. 52 (1994) 387–405.

[25] S.L. Anna, G.H. McKinley, Elasto-capillary thinning and breakup of model elastic liquids, J. Rheol. 45 (2001) 115–138.

[26] H.-C. Chang, E.A. Demekhin, E. Kalaidin, Iterated stretching of viscoelastic jets, Phys. Fluids 11 (7) (1999) 1717–1737.

[27] C. Clasen, J. Eggers, M. Fontelos, J. Li, G.H. McKinley, The beads-on-string structure of viscoelastic threads, J. Fluid Mech. 556 (2006) 283–308.

[28] J. Li, M.A. Fontelos, Drop dynamics on the beads-on-string structure for viscoelastic jets: a numerical study, Phys. Fluids 15 (2003) 922–937.

[29] J. Étienne, E.J. Hopfinger, P. Saramito, Numerical simulations of high density ratio lock-exchange flows, Phys. Fluids 17 (3) (2005) 036601.

[30] H.K. Rasmussen, O. Hassager, Three-dimensional simulations of viscoelastic instability in polymeric filaments, J. Non-Newtonian Fluid Mech. 82 (1999) 189–202.

[31] M.A. Alves, F.T. Pinho, P.J. Oliveira, The flow of viscoelastic fluids past a cylinder: finite-volume high-resolution methods, J. Non-Newtonian Fluid Mech. 97 (2001) 207–232.

[32] H.-S. Dou, N. Phan-Thien, The flow of an Oldroyd-B fluid past a cylinder in a channel: adaptive viscosity vorticity (DAVSS-w) formulation, J. Non-Newtonian Fluid Mech. 87 (1999) 47–73.