Numerical solution of the eXtended Pom-Pom model for viscoelastic free surface flows

C.M. Oishi[a], F.P. Martins[a], M.F. Tomé[b], J.A. Cuminato[b], S. McKee[c,*]

[a] *Departamento de Matemática, Estatística e Computação, Universidade Estadual Paulista, Presidente Prudente, Brazil*
[b] *Department of Applied Mathematics and Statistics, University of São Paulo, São Carlos, Brazil*
[c] *Department of Mathematics and Statistics, University of Strathclyde, Glasgow, UK*

A B S T R A C T

In this paper we present a finite difference method for solving two-dimensional viscoelastic unsteady free surface flows governed by the single equation version of the eXtended Pom-Pom (XPP) model. The momentum equations are solved by a projection method which uncouples the velocity and pressure fields. We are interested in low Reynolds number flows and, to enhance the stability of the numerical method, an implicit technique for computing the pressure condition on the free surface is employed. This strategy is invoked to solve the governing equations within a Marker-and-Cell type approach while simultaneously calculating the correct normal stress condition on the free surface. The numerical code is validated by performing mesh refinement on a two-dimensional channel flow. Numerical results include an investigation of the influence of the parameters of the XPP equation on the extrudate swelling ratio and the simulation of the Barus effect for XPP fluids.

© 2010 Elsevier B.V. All rights reserved.

1. Introduction

A quantitative understanding of polymeric flows is essential for many industrial processes. Thus, considerable effort has gone into the development of codes for the large array of complex constitutive models. An important early review of the numerical simulation of viscoelastic flows appeared in 1984 (Crochet et al. [21]); Owens and Phillips [42] brought together more recent advances while a collection of interesting problems were addressed by Walters and Webster [70].

Many codes based on a variety of numerical methods have been developed for rheological flows: finite element method (e.g. [14,15,24,26,32–34]); finite volume methods (e.g. [4,41,46,47,67,71]); finite difference methods (e.g. [16,22,62]); and mixed finite volume and finite element methods (e.g. [1,2,51,69]). These authors have restricted themselves to confined flow: less has been done for free surface flows although the Oldroyd B and the Upper-Convected Maxwell models of viscoelastic flows can be found in, for instance [12,25,43]. Nonetheless, not a great deal of work would appear to have been done in developing numerical methods for an important class of polymeric flows characterized by the Pom-Pom constitutive relationship, at least not for free surface flow problems. This model was originally proposed by McLeish and Larson [35] and applied by Inkson et al. [30] to model low density polyethylene melts in elongational and shear flows. It was used by Bishko et al. [11] to numerically study the transient flow of branched polymer melts in a planar 4:1 contraction. The original model suffered from certain weakness, for instance, discontinuous steady state solutions and unrealistic zero normal stress differences. To overcome these difficulties, Verbeeten et al. [64] proposed the improved Pom-Pom model (see also [18,49,54]). This improved formulation was called the eXtended Pom-Pom model (XPP) and various numerical methods have been suggested. In [65], the authors employed a finite element method to investigate low-density polyethylene melts using the XPP model. Following on from this, Verbeeten et al. [66] used the XPP model to solve planar contraction flow while Aboubacar et al. [3] applied the model to solve Poiseuille flow in a channel. A three-dimensional contraction flow using the multi-mode XPP model was treated by Sirakov et al. [52] while van Os and Phillips [63] considered the flow of the XPP fluid past a cylinder using a spectral element approach. Aguayo et al. [6] investigated 4:1 planar contraction flow using the XPP model; and in [5] he considered rounded-corner contraction. Recently, Inkson et al. [31] solved the models of XPP type using the spectral element method. Most recently, Russo and Phillips [50] studied extrudate swell behaviour of branched polymer melts using the multi-mode eXtended Pom-Pom model. A spectral element scheme was employed in space, while the temporal discretisation used

* Corresponding author.
E-mail addresses: cassiooishi@gmail.com (C.M. Oishi), fpmartins@fct.unesp.br (F.P. Martins), murilo@icmc.usp.br (M.F. Tomé), jacumina@icmc.usp.br (J.A. Cuminato), s.mckee@strath.ac.uk, smck@maths.strath.ac.u (S. McKee).

a second-order operator-integration-factor splitting scheme. The paper provides a clear and balanced overview of the subject. In summary, some numerical techniques have been successfully applied to solve viscoelastic flows using the Pom-Pom model. However, the simulation of free surface viscoelastic flows using the Pom-Pom constitutive model has received relatively little attention.

This paper is concerned with the development of an implicit finite difference algorithm capable of efficiently solving complex free surface flows using the single equation of the XPP model. The methodology extends previous work (see Oishi et al. [38,39]) for Newtonian free surface flows. The algorithm is described in some detail and partially validated by solving channel flow on a sequence of decreasing meshes. The paper then considers the influence of various parameters that characterize the model on the swelling ratio. Finally, we apply the code to simulate the Barus effect of XPP fluids.

2. Mathematical formulation

The governing equations for incompressible flows are the conservation of mass and momentum which can be written as

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

$$\rho \left(\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot (\mathbf{uu}) \right) = -\nabla p + \nabla \cdot \boldsymbol{\tau} + \rho \mathbf{g}, \tag{2}$$

where $\mathbf{u}$ is the velocity vector, t is the time, p is the pressure, ρ is the fluid density, $\mathbf{g}$ is the gravity field and $\boldsymbol{\tau}$ is the extra-stress tensor which is defined by an appropriate constitutive equation characterizing the fluid model.

In this work we are interested in simulating fluid flows that obey the single eXtended Pom-Pom (XPP) constitutive equation given by

$$f(\lambda, \boldsymbol{\tau})\boldsymbol{\tau} + \lambda_1 \overset{\nabla}{\boldsymbol{\tau}} + G_0(f(\lambda, \boldsymbol{\tau}) - 1)\mathbf{I} + \frac{\alpha}{G_0}(\boldsymbol{\tau} \cdot \boldsymbol{\tau}) = 2\mu_P \mathbf{D}, \tag{3}$$

where $\mathbf{D}$ is the rate of deformation tensor

$$\mathbf{D} = \frac{1}{2}[(\nabla \mathbf{u}) + (\nabla \mathbf{u})^T], \tag{4}$$

and the function $f(\lambda, \boldsymbol{\tau})$ is defined by

$$f(\lambda, \boldsymbol{\tau}) = 2\frac{\lambda_1}{\lambda_2} e^{Q_0(\lambda-1)} \left(1 - \frac{1}{\lambda} \right) + \frac{1}{\lambda^2} \left[1 - \frac{\alpha}{3G_0^2} tr(\boldsymbol{\tau} \cdot \boldsymbol{\tau}) \right]. \tag{5}$$

The parameter λ is given by

$$\lambda = \sqrt{1 + \frac{1}{3G_0} tr(\boldsymbol{\tau})}; \tag{6}$$

it is the backbone stretch (that is, it is directly coupled to the polymeric contribution in the XPP model). The upper convected derivative of a tensor $\boldsymbol{\tau}$ is defined by

$$\overset{\nabla}{\boldsymbol{\tau}} = \frac{\partial \boldsymbol{\tau}}{\partial t} + \nabla \cdot (\mathbf{u}\boldsymbol{\tau}) - [(\nabla \mathbf{u}) \cdot \boldsymbol{\tau} + \boldsymbol{\tau} \cdot (\nabla \mathbf{u})^T]. \tag{7}$$

Thus, the polymeric tensor $\boldsymbol{\tau}$ is defined by Eqs. (3)–(7).

The temporal constants of this model are λ_1 and λ_2 being, respectively, the orientation and backbone stretch relaxation times [3]. Moreover, $\mu_P = G_0\lambda_1$ and $QQ_0 = 2$, where G_0 is the linear relaxation modulus and Q is the number of arms at the backbone extremity of the Pom-Pom molecule. Additionally, the total viscosity of the fluid is given by $\mu = \mu_S + \mu_P$ (solvent and polymeric viscosities, respectively) while the parameter α controls the anisotropic drag: the model predicts a non-zero second normal stress difference provided $\alpha \neq 0$. To solve (1) and (2) it is usual to employ the so called EVSS transformation [48] which consists

of decomposing the extra-stress tensor into a sum of a Newtonian and a polymeric tensor as follows

$$\boldsymbol{\tau} = 2\mu_S \mathbf{D} + \mathbf{T}, \tag{8}$$

where μ_S is a solvent viscosity, $\mathbf{T}$ is a non-Newtonian extra-stress tensor characterizing the fluid model. Introducing Eq. (8) into Eqs. (2), (3), (5), (6) and (7) we obtain the transformed equations which are, upon nondimensionalization,

$$\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot (\mathbf{uu}) = -\nabla p + \frac{\beta}{Re}\nabla^2 \mathbf{u} + \nabla \cdot \mathbf{T} + \frac{1}{Fr^2}\mathbf{g}, \tag{9}$$

$$\frac{\partial \mathbf{T}}{\partial t} + \nabla \cdot (\mathbf{uT}) - [(\nabla \mathbf{u}) \cdot \mathbf{T} + \mathbf{T} \cdot (\nabla \mathbf{u})^\mathbf{T}]$$

$$= 2\xi \mathbf{D} - \frac{1}{We}\left\{ f(\lambda, \mathbf{T})\mathbf{T} + \xi[f(\lambda, \mathbf{T}) - 1]\mathbf{I} + \frac{\alpha}{\xi}\mathbf{T} \cdot \mathbf{T} \right\}, \tag{10}$$

$$f(\lambda, \mathbf{T}) = \frac{2}{\gamma}\left(1 - \frac{1}{\lambda} \right)e^{Q_0(\lambda-1)} + \frac{1}{\lambda^2}\left[1 - \frac{\alpha}{3\xi^2}tr(\mathbf{T} \cdot \mathbf{T}) \right], \tag{11}$$

$$\lambda = \sqrt{1 + \frac{1}{3\xi}|tr(\mathbf{T})|}, \tag{12}$$

where

$$\xi = (1 - \beta)(ReWe)^{-1}. \tag{13}$$

In these equations, the dimensionless numbers are

$$Re = \frac{\rho LU}{\mu}, \quad We = \frac{\lambda_1 U}{L}, \quad Fr = \frac{U}{\sqrt{gL}}, \quad \beta = \frac{\mu_S}{\mu}, \quad \gamma = \frac{\lambda_2}{\lambda_1}. \tag{14}$$

The symbols Re, We and Fr represent the Reynolds, Weissenberg and Froude numbers, respectively. These non-dimensional equations were obtained by using the following scaling variables: length (L), velocity (U) and gravity (g). In dimensionless form, the mass conservation Eq. (1) remains unchanged.

One feature of this fluid model is that both the Oldroyd-B and the UCM models emerge as special cases. Indeed, by taking the function $f(\lambda, \boldsymbol{\tau}) = 1$ and $\alpha = 0$ in equation (10) the Oldroyd-B model is recovered and if, in addition, we select $\beta = 0$ in (13) then the UCM model is obtained.

Thus one sees that in order to simulate the flow of a XPP fluid one needs to be able to solve the mass conservation equation (1) together with Eqs. (9)–(12) subject to appropriate initial and boundary conditions.

2.1. Initial and boundary conditions

In this work we considered four types of boundaries: prescribed inflows, outflows, rigid walls and moving free surfaces. The velocity, prescribed at an inflow, is given by

$$\mathbf{u} = \mathbf{u}_{inflow}, \tag{15}$$

while at an outflow the homogeneous Neumann condition is employed, namely,

$$\frac{\partial \mathbf{u}}{\partial n} = 0, \tag{16}$$

where n represents the direction of the outflow.

On the solid stationary walls, the no-slip condition is used ($\mathbf{u} = \mathbf{0}$). On the moving free surfaces, surface tension forces are neglected so that the correct boundary conditions are (see Batchelor [9], p. 153):

$$\mathbf{n}^T \cdot \boldsymbol{\sigma} \cdot \mathbf{n} = 0, \tag{17}$$

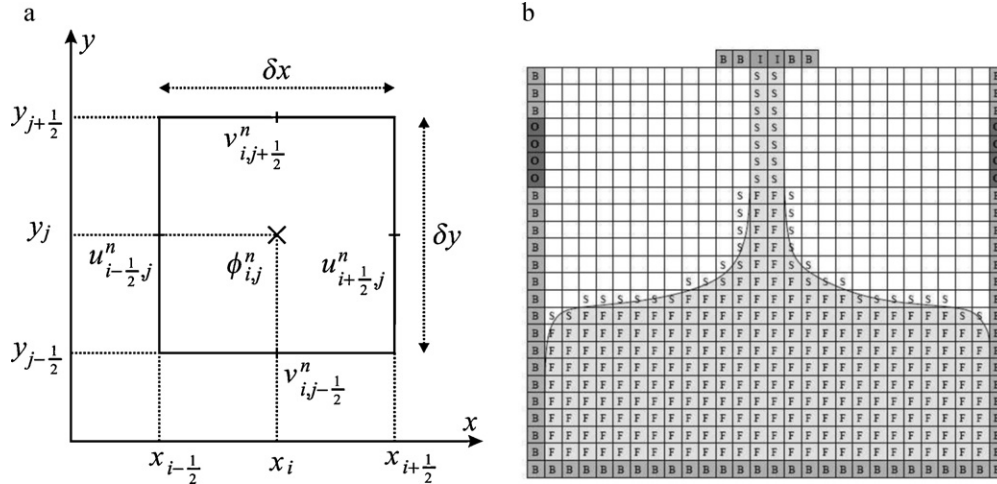$$\mathbf{m}^T \cdot \boldsymbol{\sigma} \cdot \mathbf{n} = 0, \tag{18}$$

a

b

Fig. 1. (a) Discrete variables in a staggered cell (i, j) and (b) illustration of cell type classification used.

where $\boldsymbol{\sigma} = -p\mathbf{I} + 2(\beta/Re)\mathbf{D} + \mathbf{T}$ is the dimensionless total stress tensor and $\mathbf{n}$ and $\mathbf{m}$ are, respectively, unit normal and the tangential vectors to the free surface. For two-dimensional Cartesian flows, we take $\mathbf{n} = (n_x, n_y)^T$ and $\mathbf{m} = (-n_y, n_x)^T$ so that conditions (17) and (18) can be rewritten as

$$p - \frac{2\beta}{Re}\left[n_x n_y\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right) + n_x^2\frac{\partial u}{\partial x} + n_y^2\frac{\partial v}{\partial y}\right]$$
$$= n_x^2 T^{xx} + 2n_x n_y T^{xy} + n_y^2 T^{yy}, \tag{19}$$

$$2n_x n_y\left(\frac{\partial v}{\partial y} - \frac{\partial u}{\partial x}\right) + (n_x^2 - n_y^2)\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right)$$
$$= \frac{Re}{\beta}[n_x n_y(T^{xx} - T^{yy}) + (n_y^2 - n_x^2)T^{xy}], \tag{20}$$

with obvious notation.

3. Numerical method

The numerical method used to obtain the solution of the governing equations is based on the Simplified-Marker-And-Cell formulation [8] (see also [28]) and employs the finite difference method on a staggered grid. Fig. 1(a) illustrates an example of a two-dimensional staggered cell where the velocity components are stored in the middle of the cell faces while the other variables, represented by the variable ϕ, are positioned at the cell centre.

In this work we shall treat flows with moving free surfaces so that a scheme to track the moving free surface and the fluid region is employed. For this, the cells in the mesh are classified as follows (for more details see Tomé and McKee [56]):

- EMPTY (**E**): cells that do not contain fluid;
- FULL (**F**): cells that contain fluid and do not have any face in contact with **E** cell faces;
- SURFACE (**S**): cells that contain fluid and have one or more faces in contact with **E** cell faces;
- INFLOW (**I**): cells that define an inflow;
- OUTFLOW (**O**): cells that define an outflow;
- BOUNDARY (**B**): cells that define the position and location of rigid walls.

This cell classification scheme facilitates the application of the different boundary conditions. Fig. 1(b) illustrates this classification

for a given instant of time. In this figure, for clarity, the **E** cells are represented by blank cells.

For the application of the boundary conditions at the free surface we follow the ideas of Tome and McKee [56]. We assume that the mesh is sufficiently fine so that the free surface can be locally approximated by a linear suface which can be horizontal or vertical or 45° sloped. An example of such surfaces is shown in Fig. 2.

The momentum equation (9) together with the mass conservation equation (1) are solved by a projection method to uncouple the velocity and pressure fields. The projection method was originally proposed by Chorin [17], and several modifications have been presented in the literature (e.g. [13,27] among many others). However, there are only a few papers dealing with free surface flows, for instance [39,45,68]. In this paper, we extend some ideas presented by Oishi et al. [39] for Newtonian free surface flows and apply these to the solution of the XPP model.

In many applications involving the flow of polymers, the Reynolds number is typically small ($Re < 1$), at least in parts of the spatial domain. Therefore, to avoid the parabolic stability restriction inherent in explicit schemes, the momentum equation (9) is integrated implicitly in time by the Crank-Nicolson method. In this case, the Navier-Stokes equations (9) and (1) may be rewritten as

$$\frac{\mathbf{u}^{(n+1)} - \mathbf{u}^{(n)}}{\delta t} + \nabla \cdot (\mathbf{uu})^{(n)} + \nabla p^{(n+1)}$$
$$= \frac{\beta}{2Re}[\nabla^2\mathbf{u}^{(n+1)} + \nabla^2\mathbf{u}^{(n)}] + \nabla \cdot \mathbf{T}^{(n+(1/2))} + \frac{1}{Fr^2}\mathbf{g}, \tag{21}$$

and

$$\nabla \cdot \mathbf{u}^{(n+1)} = 0, \tag{22}$$

where the term $\nabla \cdot \mathbf{T}^{(n+(1/2))}$ is treated as a source term and is calculated by

$$\nabla \cdot \mathbf{T}^{(n+(1/2))} = \frac{1}{2}[\nabla \cdot \mathbf{T}^{(n)} + \nabla \cdot \bar{\mathbf{T}}^{(n+1)}]. \tag{23}$$

The tensor $\bar{\mathbf{T}}^{(n+1)}$ is obtained by solving a hyperbolic equation using a Runge-Kutta method that will be described in Section 3.2. From now on, the upper indices (n) and $(n + 1)$ denote the fields at times $t = t_n$ and $t = t_n + \delta t$, respectively.

The projection method based on the Helmholtz-Hodge decomposition (see [23]) states that every smooth vector field can be decomposed as a sum of a gradient and a divergence-free vector field, i.e.,

$$\tilde{\mathbf{u}}^{(n+1)} = \mathbf{u}^{(n+1)} + \nabla\psi^{(n+1)}. \tag{24}$$
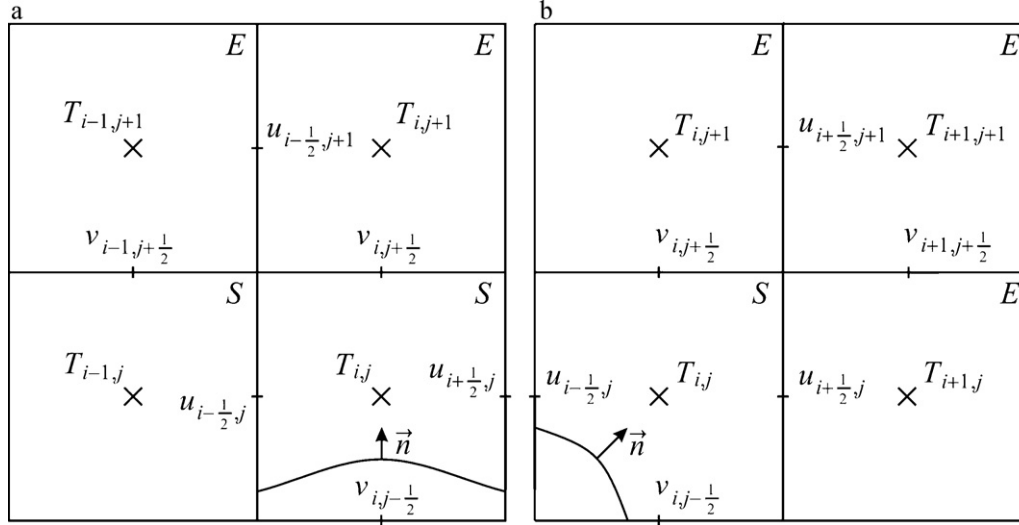
Fig. 2. Free surface intercepting two opposite faces (a) and two adjacent faces (b) of a **S** cell. In the case (a), $\vec{n}$ is assumed to be parallel to one of the coordinated axis, and we select $\vec{n} = (0, 1)^T$. For the case described in (b), the normal vector $\vec{n}$ is assumed to be at an angle of $45°$ with the coordinate axes, and we select $\vec{n} = (1/\sqrt{2}, 1/\sqrt{2})^T$.

To obtain the intermediate velocity $\tilde{\mathbf{u}}^{(n+1)}$, we approximate $p^{(n+1)}$ by $p^{(n)}$ in Eq. (21) and calculate $\tilde{\mathbf{u}}^{(n+1)}$ from

$$\frac{\tilde{\mathbf{u}}^{(n+1)} - \mathbf{u}^{(n)}}{\delta t} + \nabla \cdot (\mathbf{uu})^{(n)} + \nabla p^{(n)}$$
$$= \frac{\beta}{2Re}[\nabla^2 \tilde{\mathbf{u}}^{(n+1)} + \nabla^2 \mathbf{u}^{(n)}] + \nabla \cdot \mathbf{T}^{(n+(1/2))} + \frac{1}{Fr^2}\mathbf{g}. \quad (25)$$

The boundary conditions for $\tilde{\mathbf{u}}$ are the same as those imposed on $\mathbf{u}$. To enhance the stability of the Crank-Nicolson method the boundary conditions on rigid walls are dealt with implicitly (see Oishi et al. [40]).

Once $\tilde{\mathbf{u}}^{(n+1)}$ has been obtained we take the divergence of (24) and, upon imposing mass conservation on $\mathbf{u}^{(n+1)}$, we obtain the following Poisson equation for $\psi^{(n+1)}$

$$\nabla^2 \psi^{(n+1)} = \nabla \cdot \tilde{\mathbf{u}}^{(n+1)}. \quad (26)$$

The boundary conditions required for solving this Poisson equation are the homogeneous Neumann boundary conditions for rigid walls and inflows while homogeneous Dirichlet conditions are applied at outflows (see Tome and McKee [56]). The equations for $\psi^{(n+1)}$ on the free surface will be discussed in Section 3.1.

Having solved the Poisson equation (26) for $\psi^{(n+1)}$, the final velocity $\mathbf{u}^{(n+1)}$ is obtained from Eq. (24). The pressure is then computed by introducing (24) into (25) and, by comparing it with Eq. (21), we obtain the following equation

$$p^{(n+1)} = p^{(n)} + \frac{\psi^{(n+1)}}{\delta t} - \frac{\beta}{2Re}\nabla^2 \psi^{(n+1)}. \quad (27)$$

Once $\mathbf{u}^{(n+1)}$ and $p^{(n+1)}$ have been calculated, we are in a position to obtain the non-Newtonian extra-stress tensor $\mathbf{T}^{n+1}$ through the XPP constitutive equation (see Section 3.2).

3.1. Implicit calculation of the pressure on the free surface

Explicit MAC-type methods for solving a variety of viscoelastic free surface flows have been presented by Tomé et al. [55,58–60] (see also Paulo et al. [44]). In these papers, the pressure boundary condition on the free surface has been computed from Eq. (19) explicitly so that the boundary condition for the Poisson Eq. (26) has been $\psi^{(n+1)} = 0$ on the free surface cells (**S** cells). This procedure imposes a parabolic stability restriction on the time step size of the

form $\delta t < (Re/4)\delta^2$, where δ is the spatial mesh spacing (assuming a uniform grid). If, however, the problem involves low Reynolds number flow in any part of the flow region then the CPU time can be considerable. A methodology that allows one to overcome this often severe restriction for the specific problem of free surface flows was originally proposed by Oishi et al. [38]: here, the authors presented an implicit technique for solving low Reynolds number Newtonian flows. More recently, this idea has been extended to three-dimensional free surface flows (Oldroyd-B [39] and UCM [61]) and good results were reported. For these reasons, we follow the ideas in [39,61] and extend them to solve viscoelastic free surface flows of XPP fluids.

To apply this strategy, we consider two-dimensional free surface flows, and approximate Eq. (19) implicitly by

$$p^{(n+1)} - \frac{2\beta}{Re}\left[n_x n_y\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right) + \left(n_x^2\frac{\partial u}{\partial x} + n_y^2\frac{\partial v}{\partial y}\right)\right]^{(n+1)}$$
$$= (n_x^2 T^{xx} + 2n_x n_y T^{xy} + n_y^2 T^{yy})^{(n+1)}. \quad (28)$$

This equation couples the pressure, velocity and the non-Newtonian extra-stress tensor at the free surface. In previous works the authors decoupled the non-Newtonian stress field from the velocity and pressure fields by simply substituting $\mathbf{T}^{(n+1)}$ by $\mathbf{T}^{(n)}$. In this paper, we perform an additional step and compute an approximation $\bar{\mathbf{T}}^{(n+1)}$ by the explicit Euler method. The details of the calculation of $\bar{\mathbf{T}}^{(n+1)}$ will be given in Section 3.2. Thus, introducing $\bar{\mathbf{T}}^{(n+1)}$ into Eq. (28) we obtain

$$p^{(n+1)} - \frac{2\beta}{Re}\left[n_x n_y\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right) + \left(n_x^2\frac{\partial u}{\partial x} + n_y^2\frac{\partial v}{\partial y}\right)\right]^{(n+1)}$$
$$= (n_x^2 \bar{T}^{xx} + 2n_x n_y \bar{T}^{xy} + n_y^2 \bar{T}^{yy})^{(n+1)}. \quad (29)$$

To solve this equation we use (24) and (27) to generate new equations for the potential function on free surface cells which are now coupled with the Poisson Eq. (26). To illustrate this strategy, we will show how to obtain the equations for $\psi^{(n+1)}$ for the cases of two free surface orientations. For instance, let us consider the surface cell displayed in Fig. 2(a). For this cell we take $\mathbf{n} = (0, 1)^T$ in

which case equation (29) reduces to

$$p^{(n+1)} = \frac{2\beta}{Re}\left(\frac{\partial v}{\partial y}\right)^{(n+1)} + (\bar{T}^{yy})^{(n+1)}. \tag{30}$$

Now, imposing mass conservation (22) we get

$$\left(\frac{\partial v}{\partial y}\right)^{(n+1)} = -\left(\frac{\partial u}{\partial x}\right)^{(n+1)} \tag{31}$$

and, upon introducing (31) into (30), we obtain

$$p^{(n+1)} = -\frac{2\beta}{Re}\left(\frac{\partial u}{\partial x}\right)^{(n+1)} + (\bar{T}^{yy})^{(n+1)}. \tag{32}$$

Now, substituting the pressure from equation (27) into (32), we get

$$p^{(n)} + \frac{\psi^{(n+1)}}{\delta t} - \frac{\beta}{2Re}\nabla^2\psi^{(n+1)} = -\frac{2\beta}{Re}\left(\frac{\partial u}{\partial x}\right)^{(n+1)} + (\bar{T}^{yy})^{(n+1)}. \tag{33}$$

Finally, from Eq. (24) we have

$$\mathbf{u}^{(n+1)} = \tilde{\mathbf{u}}^{(n+1)} - \nabla\psi^{(n+1)} \tag{34}$$

which, when introduced into (33), produces the following equation for $\psi^{(n+1)}$:

$$\frac{\psi^{(n+1)}}{\delta t} - \frac{2\beta}{Re}\frac{\partial^2\psi^{(n+1)}}{\partial x^2} - \frac{\beta}{2Re}\nabla^2\psi^{(n+1)}$$
$$= -p^{(n)} - \frac{2\beta}{Re}\left(\frac{\partial\tilde{u}}{\partial x}\right)^{(n+1)} + (\bar{T}^{yy})^{(n+1)}. \tag{35}$$

Thus, for every free surface cell that has a normal vector defined by $\mathbf{n} = (0, 1)^T$, we have obtained one equation for $\psi^{(n+1)}$ associated with that cell.

For the case depicted in Fig. 2(b), we approximate the free surface by a 45^0 sloped surface so that the normal vector is $\mathbf{n} = (1/\sqrt{2}, 1/\sqrt{2})^T$. In this case, Eq. (29) reduces to

$$p^{(n+1)} = \frac{\beta}{Re}\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right)^{(n+1)} + \frac{1}{2}(\bar{T}^{xx} + 2\bar{T}^{xy} + \bar{T}^{yy})^{(n+1)}. \tag{36}$$

We now introduce $p^{(n+1)}$ from (27) and $\mathbf{u}^{(n+1)}$ from (24) into Eq. (36) to obtain

$$\frac{\psi^{(n+1)}}{\delta t} + \frac{2\beta}{Re}\frac{\partial^2\psi^{(n+1)}}{\partial y\,\partial x} - \frac{\beta}{2Re}\nabla^2\psi^{(n+1)}$$
$$= -p^{(n)} + \frac{\beta}{Re}\left(\frac{\partial\tilde{u}}{\partial y} + \frac{\partial\tilde{v}}{\partial x}\right)^{(n+1)} + \frac{1}{2}(\bar{T}^{xx} + 2\bar{T}^{xy} + \bar{T}^{yy})^{(n+1)}. \tag{37}$$

Again, for each surface cell which possesses the normal vector $\mathbf{n} = (1/\sqrt{2}, 1/\sqrt{2})^T$, we have obtained one equation involving the potential $\psi^{(n+1)}$, associated with that specific cell.

For more details of the derivation of other orientations, see Oishi et al. [38].

3.2. Calculation of the non-Newtonian extra-stress tensor for the XPP model

The non-Newtonian stress tensor $\mathbf{T}$ for the XPP model is computed from Eq. (10) by a second-order Runge-Kutta method as follows. From Eq. (10), we define

$$\mathbf{F}(\mathbf{u}, \mathbf{T}) = [(\nabla\mathbf{u})\cdot\mathbf{T} + \mathbf{T}\cdot(\nabla\mathbf{u})^T] + 2\xi\mathbf{D} - [\nabla\cdot(\mathbf{u}\mathbf{T})]$$
$$- \frac{1}{We}\left\{f(\lambda, \mathbf{T})\mathbf{T} + \xi(f(\lambda, \mathbf{T}) - 1)\mathbf{I} + \frac{\alpha}{\xi}(\mathbf{T}\cdot\mathbf{T})\right\}. \tag{38}$$

Then, $\mathbf{T}^{(n+1)}$ is obtained in two stages. First, an approximate $\bar{\mathbf{T}}^{(n+1)}$ is calculated by the explicit Euler method, namely,

$$\bar{\mathbf{T}}^{(n+1)} = \mathbf{T}^{(n)} + \delta t\,\mathbf{F}(\mathbf{u}^{(n)}, \mathbf{T}^{(n)}). \tag{39}$$

In the second stage we solve the XPP constitutive equation by the second order modified Euler method given by

$$\mathbf{T}^{(n+1)} = \mathbf{T}^{(n)} + \frac{\delta t}{2}[\mathbf{F}(\mathbf{u}^{(n)}, \mathbf{T}^{(n)}) + \mathbf{F}(\mathbf{u}^{(n+1)}, \bar{\mathbf{T}}^{(n+1)})]. \tag{40}$$

To compute Eqs. (39) and (40) the following equations are used:

$$F^{xx}(\mathbf{u}, \mathbf{T}) = 2\left(\frac{\partial u}{\partial x}T^{xx} + \frac{\partial u}{\partial y}T^{xy}\right)$$
$$- \left[\frac{\partial(uT^{xx})}{\partial x} + \frac{\partial(vT^{xx})}{\partial y}\right] + 2\xi\frac{\partial u}{\partial x}$$
$$- \frac{1}{We}\left\{f(\lambda, \mathbf{T})T^{xx} + \xi(f(\lambda, \mathbf{T}) - 1) + \frac{\alpha}{\xi}[(T^{xx})^2 + (T^{xy})^2]\right\}, \tag{41}$$

$$F^{yy}(\mathbf{u}, \mathbf{T}) = 2\left(\frac{\partial v}{\partial x}T^{xy} + \frac{\partial v}{\partial y}T^{yy}\right)$$
$$- \left[\frac{\partial(uT^{yy})}{\partial x} + \frac{\partial(vT^{yy})}{\partial y}\right] + 2\xi\frac{\partial v}{\partial y} - \frac{1}{We}$$
$$\times \left\{f(\lambda, \mathbf{T})T^{yy} + \xi(f(\lambda, \mathbf{T}) - 1) + \frac{\alpha}{\xi}[(T^{yy})^2 + (T^{xy})^2]\right\}, \tag{42}$$

$$F^{xy}(\mathbf{u}, \mathbf{T}) = \left(\frac{\partial v}{\partial x}T^{xx} + \frac{\partial u}{\partial y}T^{yy}\right)$$
$$- \left[\frac{\partial(uT^{xy})}{\partial x} + \frac{\partial(vT^{xy})}{\partial y}\right] + \xi\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right) - \frac{1}{We}$$
$$\times \left\{f(\lambda, \mathbf{T})T^{xy} + \frac{\alpha}{\xi}[T^{xy}(T^{xx} + T^{yy})]\right\}, \tag{43}$$

where from (11) we have

$$f(\lambda, \mathbf{T}) = \frac{2}{\gamma}\left(1 - \frac{1}{\lambda}\right)e^{Q_0(\lambda-1)}$$
$$+ \frac{1}{\lambda^2}\left\{1 - \frac{\alpha}{3\xi^2}[(T^{xx})^2 + 2T^{xy} + (T^{yy})^2]\right\} \tag{44}$$

and

$$\lambda = \sqrt{1 + \frac{1}{3\xi}|T^{xx} + T^{yy}|}. \tag{45}$$

3.2.1. Computation of the non-Newtonian extra-stress tensor on mesh boundaries

When solving Eqs. (41)–(43) in order to compute $\bar{\mathbf{T}}^{(n+1)}$ from (39) and $\mathbf{T}^{(n+1)}$ from (40), care should be taken when approximating the derivatives contained within the material derivative of Eqs. (41)–(43). It is known that first order upwinding can result in solutions that contain excessive diffusion while second order central difference approximations can lead to oscillatory solutions. To avoid these difficulties many researchers have been developing high order accurate stable upwind methods to approximate the convective terms of hyperbolic equations. In this work we employ CUBISTA (Convergent Universally Bounded Interpolation Scheme for the Treatment of Advection) [7]. This method requires that the values of the variable to be approximated, say φ, be approximated by using upstream (φ_U), downstream (φ_D) and remote upstream (φ_R) values with respect to the point at which the variable is defined
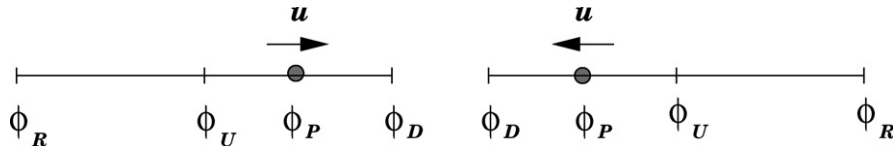
Fig. 3. Reference points used for the CUBISTA upwind scheme.

(see Fig. 3). Therefore, when computing the non-Newtonian extra-stress tensor in regions that are near the boundary, further values of the non-Newtonian stress tensor **T** are needed. These are obtained as follows:

Inflow boundaries: If the velocity at fluid entrance is constant then we follow the strategy of Crochet et al. [21] (see also Mompean and Deville [36], Tomé et al. [58]) and set **T** = **0**, while for fully developed flows prescribed by

$$u(y) = 4U\left(\frac{y}{L}\right)\left(1 - \frac{y}{L}\right), \qquad v = 0, \tag{46}$$

we impose the Oldroyd-B profile for **T**, namely,

$$T^{xx} = 2\frac{We}{Re}\left(1 - \frac{\lambda_2}{\lambda_1}\right)\left(\frac{\partial u}{\partial y}\right)^2, \quad T^{yy} = 0,$$

$$T^{xy} = \frac{1}{Re}\left(1 - \frac{\lambda_2}{\lambda_1}\right)\left(\frac{\partial u}{\partial y}\right). \tag{47}$$

Outflow boundaries: At fluid exit we employ homogeneous Neumann conditions (see Mompean [36], Tomé et al. [58])

$$\frac{\partial T^{xx}}{\partial n} = \frac{\partial T^{xy}}{\partial n} = \frac{\partial T^{yy}}{\partial n} = 0, \tag{48}$$

where n denotes the normal direction to the boundary.

Rigid walls: On these boundaries we use the no-slip condition (**u** = **0**) and compute **T** directly from Eq. (10). For instance, on a rigid wall parallel to the x-axis the tensor **T** is calculated from the equations

$$\frac{\partial T^{xx}}{\partial t} = 2\frac{\partial u}{\partial y}T^{xy} - \frac{1}{We}\left\{f(\lambda, \mathbf{T})T^{xx} + \xi\left[f(\lambda, \mathbf{T}) - 1\right]\right.$$
$$\left. + \frac{\alpha}{\xi}[(T^{xx})^2 + (T^{xy})^2]\right\}, \tag{49}$$

$$\frac{\partial T^{yy}}{\partial t} = -\frac{1}{We}\left\{f(\lambda, \mathbf{T})T^{yy} + \xi[f(\lambda, \mathbf{T}) - 1] + \frac{\alpha}{\xi}[(T^{xy})^2 + (T^{yy})^2]\right\}, \tag{50}$$

$$\frac{\partial T^{xy}}{\partial t} = \frac{\partial u}{\partial y}(T^{yy} + \xi) - \frac{1}{We}\left\{f(\lambda, \mathbf{T})T^{xy} + \frac{\alpha}{\xi}[T^{xy}(T^{xx} + T^{yy})]\right\}. \tag{51}$$

The equations for calculating the non-Newtonian extra-stress tensor on rigid walls parallel to the y-axis are obtained similarly.

3.3. Computational algorithm

We are now in a position to write down the algorithm for simulating the flow of a XPP fluid. It is supposed that at time $t = t_n$ the solenoidal velocity $\mathbf{u}^{(n)}$, the pressure field $p^{(n)}$, the non-Newtonian extra-stress tensor $\mathbf{T}^{(n)}$ are known. The solutions $\mathbf{u}^{(n+1)}$, $p^{(n+1)}$ and $\mathbf{T}^{(n+1)}$ are obtained by the following steps.

1. Compute the stress tensor on mesh boundaries according to the equations described in Section 3.2.1.
2. Calculate $\bar{\mathbf{T}}^{(n+1)}$ from Eq. (39) and then compute $\mathbf{T}^{(n+(1/2))}$ by Eq. (23).
3. Calculate the intermediate velocity $\tilde{\mathbf{u}}^{(n+1)}$ from Eq. (25) using the Crank-Nicolson scheme. The resulting linear systems are

solved by the Conjugate Gradient method with diagonal pre-conditioning.
4. Solve the Poisson equation (26) simultaneously with the equations obtained for $\psi^{(n+1)}$ from the application of the boundary conditions for the pressure on the free surface (see Section 3.1). The corresponding finite difference equations will generate a large nonsymmetric linear system which can be efficiently solved by the Bi-conjugate gradient method with SOR (BiCGstab-SOR) pre-conditioning.
5. Calculate the final velocity field $\mathbf{u}^{(n+1)}$ from Eq. (24).
6. Update the final pressure field $p^{(n+1)}$ using Eq. (27).
7. Calculate $\mathbf{T}^{(n+1)}$ using Eq. (40).
8. Move the free surface. In this last step, the velocity $\mathbf{u}^{(n+1)}$ is used to compute a new free surface by solving

$$\frac{d\mathbf{x}_P}{dt} = \mathbf{u}_P^{(n+1)} \tag{52}$$

for every particle P. The velocity $\mathbf{u}_P^{(n+1)}$ is found by performing bilinear interpolation using the four nearest neighbours of $\mathbf{u}^{(n+1)}$. The free surface is defined by a set of points that move with the fluid. A scheme for inserting and deleting particles is employed. Details on the free surface movement and particle insertion/deletion can be found in Tomé et al. [57].

The approximation of the equations contained in the algorithm above by the finite difference method is a somewhat obvious extention of those in [58] and so are not given here.

4. Time-step calculation

The Oldroyd-B solver of Tomé et al. [58] solves the momentum equation explicitly so that the time-step size is required to satisfy the restrictions

$$\delta t < \delta t_{VISC} = \frac{Re}{4}h^2, \tag{53}$$

$$\delta t < \delta t_{CFL} = \frac{h}{V_{max}}, \tag{54}$$

where V_{max} denotes the maximum of velocity in the x and y-directions. Condition (53) is a viscous restriction due to the explicit calculation of the momentum equations while (54) is the CFL condition. Therefore, if the Reynolds number is small ($Re \ll 1$) then condition (53) would lead to a very small time-step. One reason the Crank-Nicolson is being employed to solve the momentum equations is that we expect it to obey a less restrictive condition. We follow the procedure employed by Oishi et al. [39] (see also Tome and McKee [56]) and compute the time step by

$$\delta t = fact * \min\{fact_1 * \delta t_{VISC}, fact_2 * \delta t_{CFL}\}, \tag{55}$$

where $fact, fact_1, fact_2 > 0$. The constants $fact, fact_1, fact_2$ appear as a conservative measure since the true solenoidal velocities are not known at the begining of the calculation. The implementation of these inequalities follows the ideas of Tome and McKee [56].

In the calculations presented by the explicit Oldroyd-B solver of Tomé et al. [58] the constant $fact$ assumed the value of 0.2 while the constants $fact_1$ and $fact_2$ were assigned the value of 0.5. However, if the flow involves a low Reynolds number ($Re \ll 1$) then condition

(53) could be unnecessarily restrictive. Thus, we follow the procedure used by Oishi et al. [39] and set the value of the constant $fact_1 \gg 1$; for instance, in this paper we use $fact_1 = 10$. This more than compensates for the extra computations arising from using the Crank-Nicolson method to solve the momentum equations.

5. Verification of the numerical method

To verify the correctness of the numerical method presented in this paper, we simulated the flow of a XPP fluid in a two-dimensional channel of width L and length $5L$. At the channel entrance, the fully developed flow given by Eq. (46) was used while the non-Newtonian extra-stress tensor **T** assumed the Oldroyd-B profile of Eq. (47). At the channel exit a homogeneous Neumann condition was imposed for both the velocity and the non-Newtonian extra-stress tensor **T**. At the channel walls the velocity obeyed the no-slip condition while the non-Newtonian extra-stress tensor **T** was calculated from Eqs. (49)–(51).

The simulation started with the channel empty. Fluid was then injected through the entrance and the channel progressively filled. Initially, there was a free surface within the channel and on that free surface the boundary conditions imposed were the free surface stress conditions given by Eqs. (19) and (20).

The following input data were employed: $L = 1$, $U = 1$, $Re = 0.1$, $We = 2$, $\beta = 0.5$, $\alpha = 0.2$, $\gamma = 0.5$, $Q = 2.0$ and gravity was neglected. To study the convergence of the numerical method, channel flow was simulated on five meshes: M1 ($h = 0.2$, 5×25 cells), M2 ($h = 0.1$, 10×50 cells), M3 ($h = 0.05$, 20×100 cells), M4 ($h = 0.025$, 40×200 cells) and M5 ($h = 0.0125$, 80×400 cells). An analytic solution for this problem is not known so we compared the numerical solutions obtained on meshes M1, M2, M3 and M4 to the solution obtained on mesh M5 which, hereafter, we shall refer to as SOL_{EXACT}.

Channel flow was simulated on the meshes mentioned above until $t = 50$. At this time the results did not show any variation implying that steady state had been established.

Table 1
Errors obtained on meshes M1, M2, M3 and M4.

	$E(u)$	$E(T^{xx})$	$E(T^{xy})$	$E(T^{yy})$
M1	5.69×10^{-2}	5.96×10^{-2}	1.55×10^{-1}	7.62×10^{-2}
M2	1.52×10^{-2}	4.35×10^{-2}	1.68×10^{-2}	5.50×10^{-2}
M3	3.68×10^{-3}	1.00×10^{-2}	5.16×10^{-3}	1.62×10^{-2}
M4	7.39×10^{-4}	2.03×10^{-3}	9.85×10^{-4}	3.19×10^{-3}

The solution profiles obtained by the numerical method using the meshes mentioned above are displayed in Fig. 4 where we can observe that the solutions obtained on meshes M1, M2, M3 and M4 agree well with the SOL_{EXACT}. To quantify the convergence of the numerical method we calculated the relative errors using the l_2-norm by

$$||E||_2 = \sqrt{\frac{\sum_{i,j}(SOL_{EXACT} - SOL_{NUM})^2}{\sum_{i,j}(SOL_{EXACT})^2}}, \qquad (56)$$

where SOL_{NUM} denotes the solutions obtained on meshes M1–M4. Table 1 displays the calculated errors for the velocity u and also for the components of the tensor **T**. We can see in Table 1 that as the mesh is refined all the errors decreased indicating convergence of the algorithm.

5.1. The convergence of the free surface

To supply further evidence concerning the convergence of the numerical method, we simulated the time-dependent extrudate swell problem (for details see Section 6) using three meshes: M1 ($h = 0.1$), M2 ($h = 0.05$) and M3 ($h = 0.025$). With reference to Fig. 5 we used $L = 1$; the velocity at the channel entrance was given by Eq. (46) with $U = 1$ while the non-Newtonian extra-stress tensor **T** was
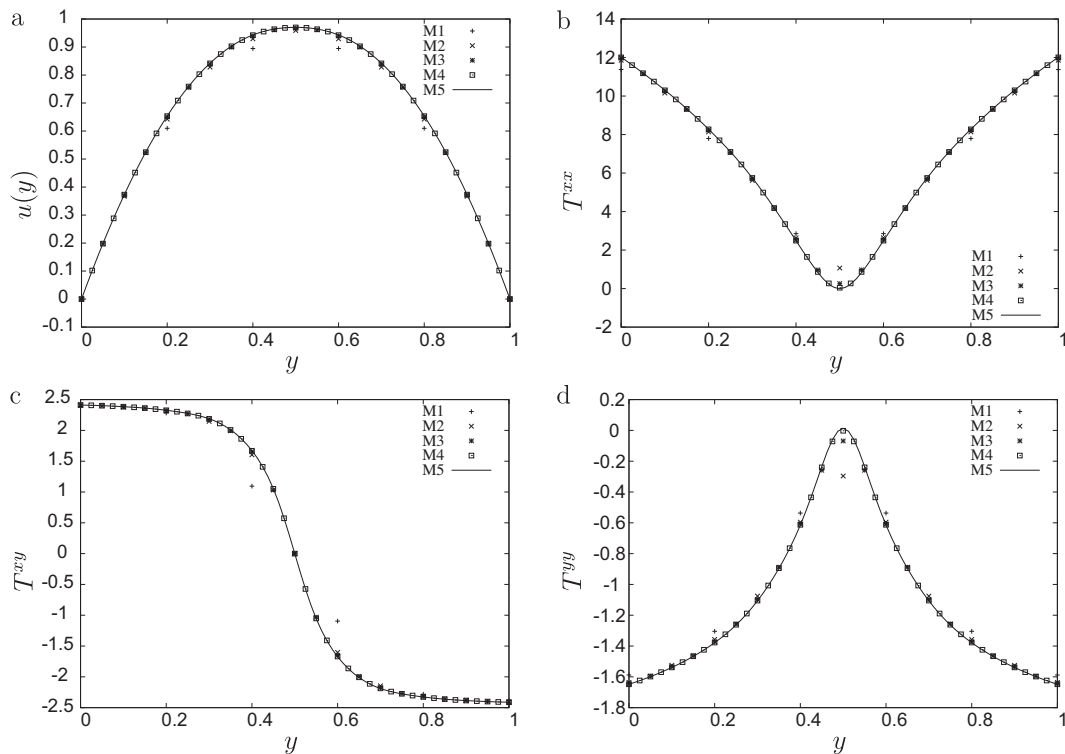

Fig. 4. Numerical solution of channel flow of a XPP fluid. Comparison of the numerical solutions obtained on meshes M1, M2, M3 and M4 with the numerical solution obtained on mesh M5. (a) u, (b) T^{xx}, (c) T^{xy}, (d) T^{yy}.
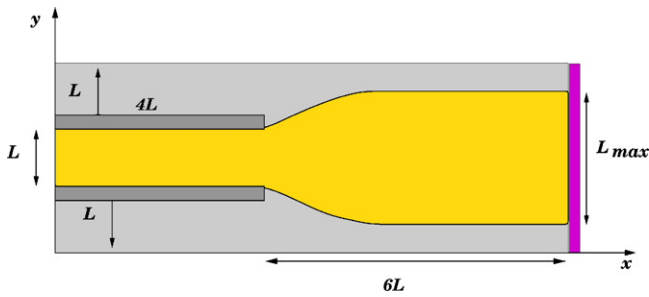
Fig. 5. Domain definition for the simulation of unsteady extrudate swell (pink surface represents an outflow boundary through which the fluid can flow). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of the article.)

defined by Eq. (47). In this study we considered the XPP model with the following parameters: $\alpha = 0.1$, $Re = 0.05$, $We = 10$, $\beta = 0.5$, $\gamma = 0.8$, $Q = 8.0$.

The fluid surface obtained in these simulations at times $t = 10$, $t = 15$ and $t = 50$ are displayed in Fig. 6. From this figure, we can observe that the free surface profiles obtained on meshes M1 and M2 approach the free surface profile obtained using the finer mesh M3. This result indicates the convergence of the numerical method for time-dependent moving free surface flows.

In addition, a comparison was performed between the spectral method of Russo and Phillips [50] and the Marker and Cell approach of this paper. The extrudate swell problem was simulated for the XPP model with the following parameters: $\alpha = 0.025$, $Re = 1$,
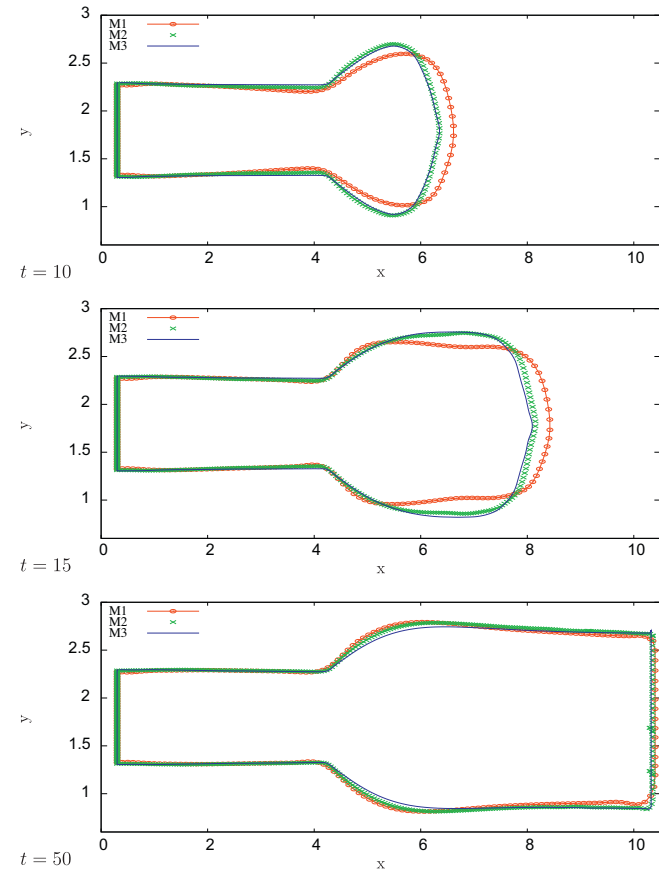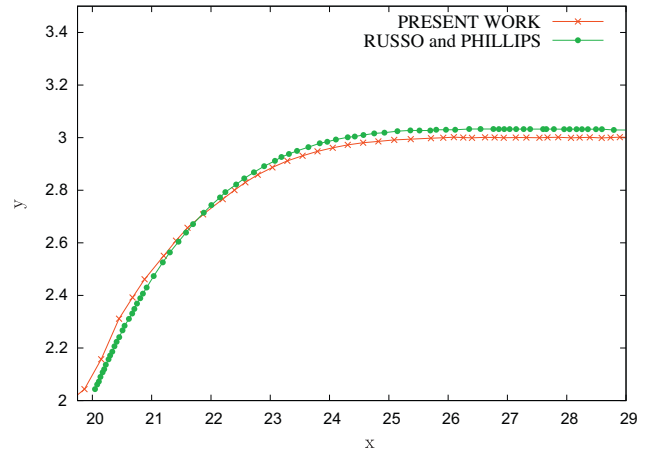

Fig. 7. Comparison between present computation and the numerical method of Russo and Phillips [50] for the time-dependent extrudate swell using the XPP model.

$We = 1$, $\beta = 0.11111111$, $\gamma = 1.0$ and $Q = 4.0$. The domain over which the problem was solved was the same as that used by Russo and Phillips [50] with the die exit at (nondimensional) $x = 20$. One can see from Fig. 7 that the profile of the free surface obtained by the method of this paper exhibits similar qualitative behaviour. Since the non-dimensional scaling in the Russo and Phillips paper is not specified, and the two methods are quite different, these results, we would argue, are quite good and display reasonable qualitative agreement.

5.2. Efficiency of the linear solvers

In this work, two linear solvers were employed: the Conjugate Gradient method with diagonal pre-conditioning or Jacobi pre-conditioner (CG-Jacobi) and the Bi-Conjugate Gradient Stabilized with a SOR pre-conditioner (BiCGstab-SOR). The CG-Jacobi was used to solve the symmetric linear system resulting from the discretization of the momentum equations for calculating the intermediate velocity $\tilde{\mathbf{u}}^{(n+1)}$ while the BiCGstab-SOR was applied to invert the nonsymmetric linear system for the pressure ψ (see the computational algorithm in Section 3.3).

The effect of grid refinement for solving the extrudate swell problem is illustrated in Table 2 with a study on the efficiency of the preconditioners for three meshes: M1, M2 and M3 (see Section 5.1). Table 2 displays the number of equations ($Neq(\cdot)$) for $\tilde{\mathbf{u}}$ ($\tilde{u}$ and $\tilde{v}$)) and ψ, the number of iterations ($Nit(\cdot)$), and the rate of increase in the number of iterations compared to the increase in the number of equations. This is estimated from the following:

$$N_i(\cdot) = \frac{Tit(M_{i+1})/Tit(M_i)}{Teq(M_{i+1})/Teq(M_i)}, \quad i = 1, 2. \tag{57}$$

In Eq. (57) we define Teq and Tit as, respectively, the total numbers of equations and iterations used for solving the linear system arising from the implicit formulation in each mesh. To simplify our study, we chose a specific model by setting $\alpha = 0.1$, $Re = 0.05$, $We = 10$, $\beta = 0.5$, $\gamma = 0.8$, $Q = 8.0$.

It is evident from this table that the preconditioners Jacobi, for calculating $\tilde{\mathbf{u}}$, and SOR, to solve the nonsymmetric linear system

Fig. 6. Free surface profiles obtained in the simulation of the time-dependent extrudate swell using the XPP model. Results shown on meshes M1, M2 and M3 at selected times. XPP model parameters employed were: $\alpha = 0.1$, $Re = 0.05$, $We = 10$, $\beta = 0.5$, $\gamma = 0.8$, $Q = 8.0$.

Table 2
Influence of mesh refinement on CG-Jacobi/BiCGstab-SOR methods for solving the extrudate swell problem.

Mesh	$Neq(\tilde{\mathbf{u}})$	$Nit(\tilde{\mathbf{u}})$	$N_i(\tilde{\mathbf{u}})$	$Neq(\psi)$	$Nit(\psi)$	$N_i(\psi)$
M1	3,234	31	–	1,698	13	–
M2	13,224	66	0.52	6,674	24	0.47
M3	51,048	126	0.49	26,024	44	0.47

Table 3
Performance study of the linear solvers employed in the implicit methodology. Input data used: $\alpha = 0.1$, $Re = 0.05$, $We = 10$, $\beta = 0.5$, $\gamma = 0.8$, $Q = 8.0$.

Methods	$Nit(\tilde{\mathbf{u}})$	$Nit(\psi)$	CPU time (in hours)
CG/BiCG-Jacobi	154	597	50.4
CG-Jacobi/BiCGstab-SOR	126	44	18.1

for ψ, are sensitive to the number of equations. Furthemore the number of iterations increases as the grid is refined.

In particular, we see that the number of iterations appears to increase linearly with the number of equations. From Table 2 we observe that the number of equations for mesh M2 is roughly 4 times that of M1 and the number of equations for mesh M3 is again roughly 4 times that of M2. This is so both for the linear system arising from the discretization of the momentum equations ($\tilde{u}$ – column 2 of Table 2) as well as that arising from the discretization of the equation for the pressure ψ (column 5 of Table 2). As for the number of iterations we see from columns 3 and 6 of Table 2 that this number doubles with each mesh refinement. In columns 4 and 7 we present the ratio calculated from (57) which show that the iteration count increases linearly with the number of equations with a constant of proportionality roughly equal to 0.5.

To study the efficiency of these solvers Table 3 provides a comparison between the solvers employed in this work and the solvers used in our previous work (see [39,61]) for the extrudate swell problem. We simulated the extrudate swell using the same domain employed in Section 5.1, with the mesh M3.

We considered the solution at time $t = 45$ s and displayed, in Table 3, the number of iterations taken by the linear solvers as well as the total CPU time of the entire simulation. From Table 3 we can see that the number of iterations required for the solution of the linear systems for $\tilde{u}$ and $\tilde{v}$ was slightly smaller for the CG-Jacobi than for the CG without pre-conditioning. Thus, in this case the use of a simple pre-conditioner led to modestly improved convergence. On the other hand, Table 3 shows that the number of iterations required by the preconditioned BiCGstab-SOR to solve the nonsymmetric linear system was remarkably reduced when compared with the BiCG-Jacobi method. Therefore, the application of an efficient pre-conditioner was essential for solving the nonsymmetrical linear system. Finally, Table 3 shows that the CPU time taken using the solvers CG-Jacobi/BiCGstab-SOR was substantially less than the CPU time for the solvers CG/BiCG-Jacobi.

In these numerical experiments, the convergence criterion for the linear solvers was $\epsilon = 10^{-10}$ while the relaxation parameter in the SOR pre-conditioner was $\omega = 1.8$. The results were obtained on a computer with $4 \times$ AMD Opteron 844 / 1.8 GHz processor and 8 Gbytes RAM running Linux.

6. Numerical simulation of unsteady extrudate swell

The unsteady extrudate swell problem consists of a jet of viscous fluid exiting a capillary of width L where, due to the normal stress differences, the jet swells and its width expands to a maximum L_{max} (see Fig. 5). The amount of swell can be measured by computing the swelling ratio S_r given by

$$S_r = \frac{L_{max}}{L}. \tag{58}$$

It is known that for Newtonian fluids, the jet does not suffer large swelling ratios and for low Reynolds number axisymmetric jets the maximum S_r is of order 13% (see Bird et al. [10]). However, for viscoelastic fluids, the jet swell can be very large and the swelling ratio S_r can attain values above 100% (e.g. [19]). This problem has many applications so that considerable effort has been employed to develop techniques to simulate the extrudate swell of complex fluids (e.g. [20,44,50,53,55,58]).

To demonstrate that the implicit technique presented in this paper can cope with the complex flows obtained from using the Pom-Pom model we applied it to simulate this unsteady extrudate swell problem.

We considered a 2D-channel with width L and length $4L$ and an outflow boundary positioned at a distance $6L$ from the channel exit. A domain size of $10L \times 3L$ was employed (see Fig. 5). On the channel entrance, walls and outflow, the boundary conditions were the same as those employed in the previous section. On the moving free surface, the boundary conditions were those described in Section 2.1, namely (19) and (20).

In the results presented next, we employed $L = 1$, $U = 1$, $Re = 0.05$ and used a mesh spacing $h = 0.05$ in all simulations. As the Reynolds number is small ($Re \ll 1$) we anticipate that a length of $3L$ was sufficient for the flow to develop inside the channel. The time step was automatically generated subject to the restrictions given in Section 4.

To verify the robustness of the numerical method we investigated the effect of the Pom-Pom parameters on the extrudate swelling ratio (S_r). We perfomed a number of simulations with various values of the parameters α, β, γ, Q and We.
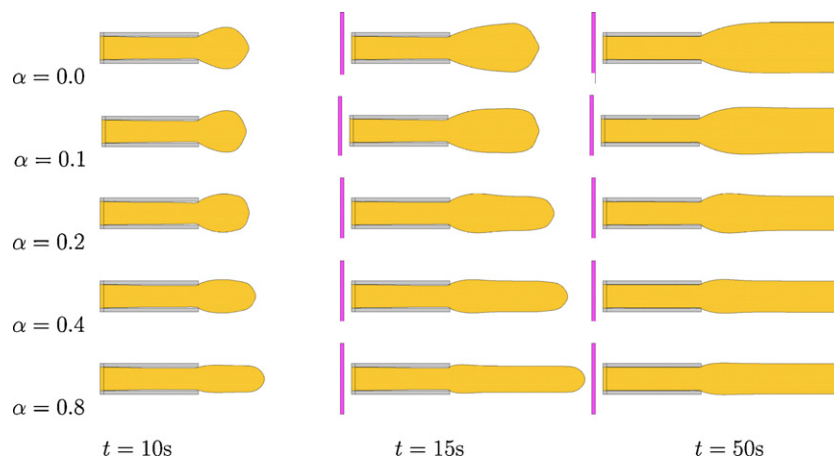

Fig. 8. Numerical simulation for the extrudate swell of a XPP fluid using the implicit formulation. Fluid flow visualization for different values of α at selected times. $Re = 0.05$, $We = 10$, $\beta = 0.5$, $\gamma = 0.8$, $Q = 8.0$.
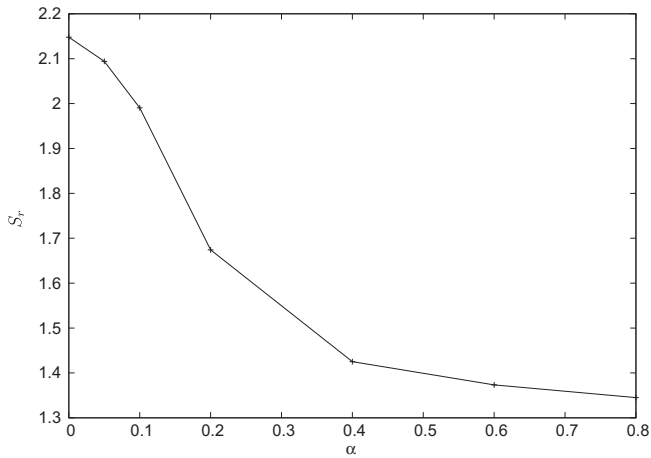
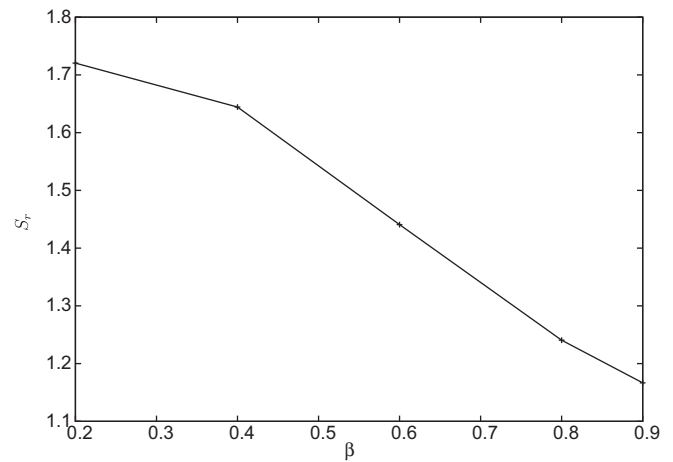**Fig. 9.** Extrudate swelling ratios S_r obtained as function of α.

Fig. 11. Extrudate swelling ratios S_r obtained as a function of β.

6.1. Influence of α

The extrudate swell depends on the first normal stress difference (N_1) while in the XPP model the parameter α is the coefficient in front of the second normal stress difference (N_2). Therefore, we might anticipate that the smaller the parameter α is, the greater would be the swelling ratio S_r. To verify this hypothesis we simulated the unsteady extrudate swell for increasing values of the parameter α while the remaining parameters were held fixed. We used $We = 10$, $\beta = 0.5$, $\gamma = 0.8$, $Q = 8.0$ and $\alpha = 0.0$, 0.05, 0.1, 0.2, 0.4, 0.6, 0.8. Fig. 8 displays the fluid flow visualization at selected times while Fig. 9 shows the variation of the extrudate swelling ratio (S_r) as function of α. We can see from Fig. 9 that the extrudate swelling decreases as α increases; the maximum swelling ratio was approximately 2.15 for $\alpha = 0$ and the smallest swelling ratio was approximately 1.35 when $\alpha = 0.8$. Despite a high Weissenberg number ($We = 10$), we note that the swelling ratio was rather modest when $\alpha = 0.8$.

6.2. Influence of β

The parameter $0 \leq \beta \leq 1$ is associated with the amount of Newtonian solvent. A value of β close to zero corresponds to highly entangled systems (highly elastic fluids) while a value of β near 1 corresponds to dilute or less-entangled solutions (almost a Newto-

nian fluid). More details about the significance of the parameter β in the XPP model can be found in Aboubacar et al. [3].

To analyze the influence of the solvent contribution on the swelling ratio S_r, the extrudate swell was simulated using the following data: $\beta = 0.2$, 0.4, 0.6, 0.8, 0.9 and $We = 10.0$, $\alpha = 0.01$, $\gamma = 0.3$, $Q = 8.0$. Fig. 10 illustrates the fluid flow visualization obtained for different values of β. We observe that the extrudate swelling ratio increases when the polymeric solution becomes more concentrated as we can see clearly from Fig. 11. Indeed, we observe that S_r is decreasing linearly when $\beta > 0.4$. Thus, the largest extrudate swelling ratio was obtained for the smallest value of β employed. Conversely, the largest value for β (almost a Newtonian fluid) resulted in the smallest value for S_r.

6.3. Influence of γ

In the XPP model, the parameter $0 \leq \gamma \leq 1$ represents the ratio between the relaxation time of the stretch of the backbone (λ_1) and the orientation relaxation time (λ_2). Thus, this parameter is related to the degree of entanglement of the melt. High values of γ corresponds to molecules with relatively short backbone lengths while small values of γ corresponds to highly entangled backbone configurations.

To investigate the influence of $\gamma \in (0, 1)$ on the extrudate swelling ratio of a XPP fluid we performed a number of simulations
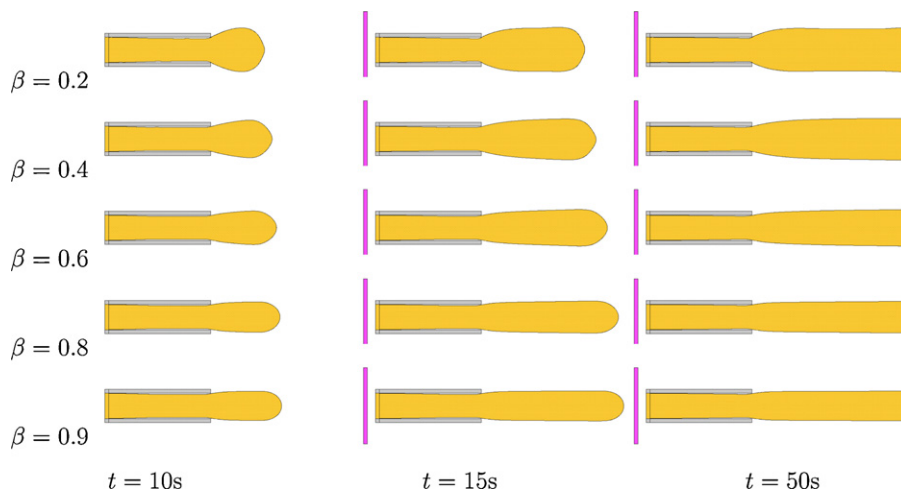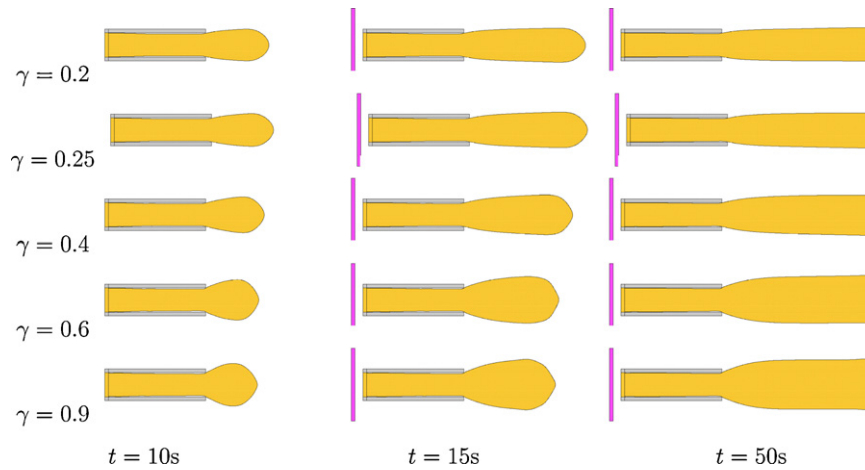

Fig. 10. Numerical simulation for the extrudate swell of a XPP fluid using the implicit formulation. Fluid flow visualization for different values of β at selected times. $Re = 0.05$, $We = 10$, $\alpha = 0.01$, $\gamma = 0.3$, $Q = 8$.

Fig. 12. Numerical simulation for the extrudate swell of a XPP fluid using the implicit formulation. Fluid flow visualization for different values of γ at selected times. $Re = 0.05$, $We = 10$, $\alpha = 0.01$, $\beta = 0.5$, $Q = 8.0$.

of unsteady extrudate swell for various values of γ. The parameters $We = 10$, $\alpha = 0.01$, $\beta = 0.5$, $Q = 8.0$ were kept fixed while the parameter γ assumed the following values 0.2, 0.25, 0.4, 0.6, 0.8, 0.9.

Fig. 12 displays the time evolution of the extrudate swell for each value of γ. We can see that as γ increases the extrudate swell also increases. This is quantified in Fig. 13 where the extrudate swelling ratios are given as a function of γ. We note that for $\gamma \in [0, 0.6]$ the extrudate swelling ratio S_r grows linearly and monotonically with γ. In short, the greatest swelling ratio occurred for the largest value of γ.

6.4. Influence of Q

The parameter Q represents the number of arms at each end of the backbone of the polymeric Pom-Pom molecule and consequently affects the level of entanglement. Thus one might expect the level of entanglement to become larger with increasing Q.

To investigate the effect of this parameter on the extrudate swell we performed various simulations with increasing values of Q. The data employed were $We = 10$, $\alpha = 0.01$, $\beta = 0.5$, $\gamma = 0.3$, and $Q = 1, 2, 4, 7, 11, 15, 20$.
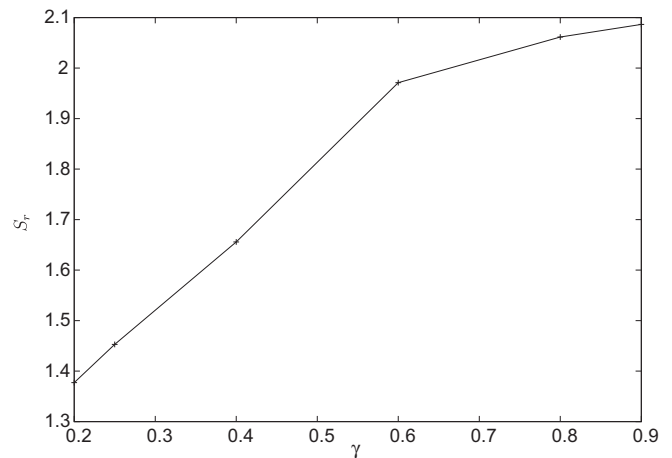

Fig. 13. Extrudate swelling ratios S_r as a function of γ.

The numerical results obtained are summarized in Figs. 14 and 15. The fluid flow visualization of the results at selected times is shown in Fig. 14 while Fig. 15 displays the extrudate swelling ratio S_r obtained with the implicit tech-
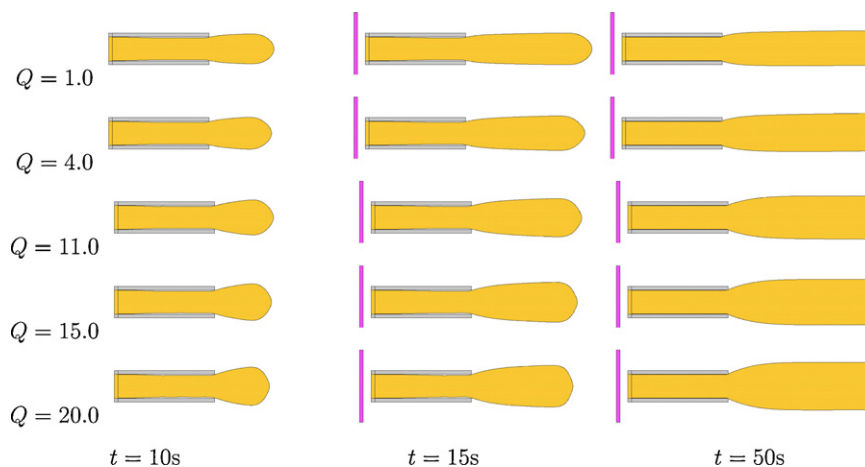

Fig. 14. Numerical simulation for the extrudate swelling of a XPP fluid using the implicit formulation. Fluid flow visualization for different values of Q at selected times. $Re = 0.05$, $We = 10$, $\alpha = 0.01$, $\beta = 0.5$, $\gamma = 0.3$.
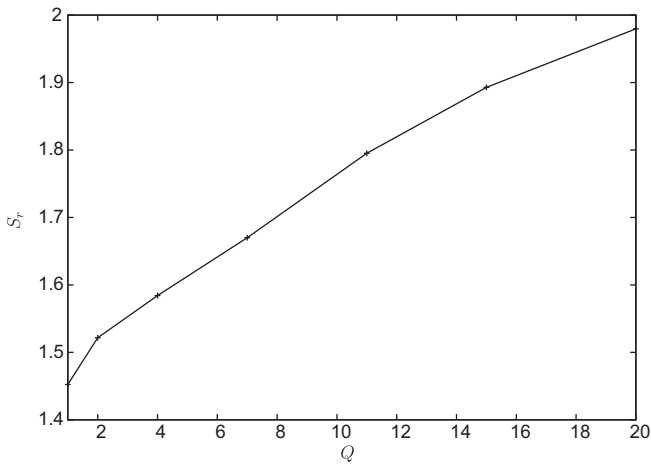
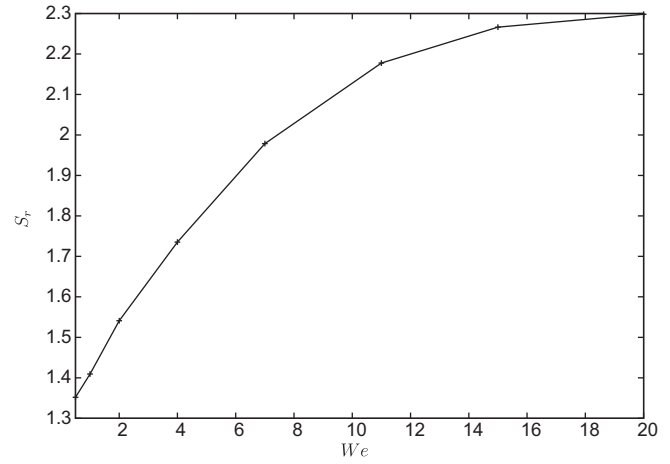**Fig. 15.** Extrudate swelling ratio S_r as a function of Q.

Fig. 17. Results of the extrudate swelling ratio S_r of a XPP fluid for different values of We.

nique described in this paper. It can be seen in Fig. 15 that the extrudate swelling ratio increases as the number of arms grows.

6.5. Influence of We

We now examine the influence of the Weissenberg number on the extrudate swelling ratio of XPP fluids. This parameter is related to the viscoelasticity of the fluid and it is anticipated that the extrudate swelling ratio might well be an increasing function of We. To verify this fact, we simulated the time-dependent extrudate swell for the following values of We: 0.5, 1, 2, 4, 7, 11, 15, 20.

The results of these simulations are given in Figs. 16 and 17. Fig. 16 displays the fluid flow configuration at selected times while Fig. 17 plots the swelling ratio obtained in these simulations. We can see from Fig. 17 that the swelling ratio is, indeed, an increasing function of We that appears to tend to an asymptotic value of around 2.3.

7. Simulation of the Barus effect: the influence of viscosity

The study in the previous section provided some indication as to how the various parameters might be chosen so that substantial elastic effects might be exhibited. This section is concerned with

Table 4
Data used in the XPP model to simulate the Barus effect with gravity.

α	β	γ	Q	We
0.01	0.55	0.8	15	20

finding a particular selection of parameters which will not only provide a large swelling ratio, but will also (when gravity is included) produce the so-called Barus effect [29]. This is an elastic memory effect which reduces the swelling ratio, beyond the capillary outlet, to the original diameter (or even less) of the fluid when it was flowing in the tube.

On this occasion we have employed a channel of length $10L$ through which fluid flows (cf. Fig. 5 with $3L$ replaced by $4L$ and $10L$ replaced by $12L$); the length $10L$ was chosen to ensure steady state flow for a range of Reynolds numbers prior to the emergence of the jet. The jet then travels a distance of $12L$ before reaching an outflow (displayed in pink where continuative outflow boundary conditions are applied). Gravity acts vertically downwards with $g = 9.81 \, \mathrm{m\,s^{-2}}$. A mesh size of (40×220)-cells was used ($h = 0.1$). The data employed in the XPP model are displayed in Table 4. The scaling for the velocity was $U = 0.5 \, \mathrm{m\,s^{-1}}$ and the length scale was $L = 0.01 \, \mathrm{m}$.
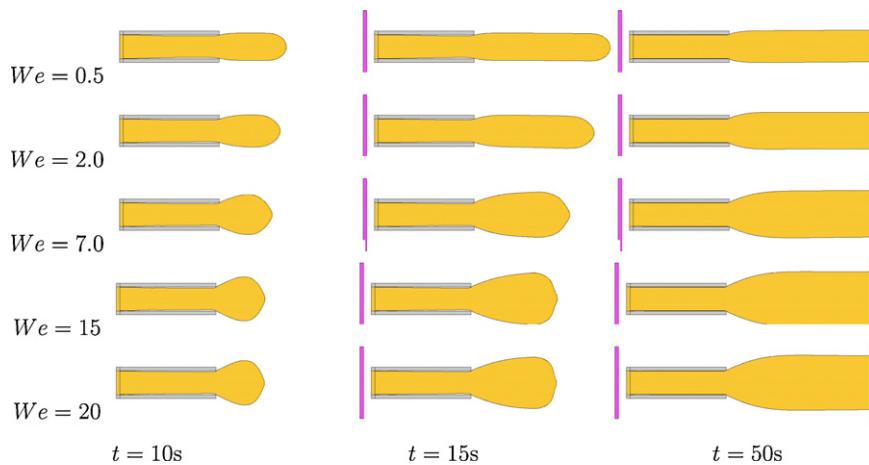
Fig. 18.

Fig. 16. Numerical simulation for the extrudate swell of a XPP fluid using the implicit formulation. Fluid flow visualization for different values of We at selected times. $Re = 0.05$, $\alpha = 0.01$, $\beta = 0.5$, $\gamma = 0.8$, $Q = 8.0$.
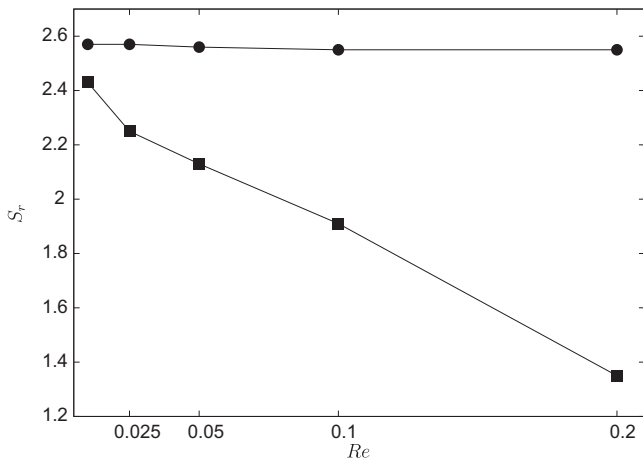
Fig. 18. Variation of the swelling ratio S_r against Re. With gravity (squares) and without gravity (circles).

Fig. 20. Numerical simulation of the Barus effect with gravity included. Fluid flow visualization at $t = 0.60$ s.

To observe the influence of viscosity on the Barus effect we performed five simulations with the Reynolds number taking the values of 0.2, 0.1, 0.05, 0.025 and 0.01. We have also performed five additional simulations using the same input data except that gravity has now been set to zero. The difference in Fig. 17 is substantial and perhaps a little surprising. For zero gravity, and the special choice of parameters adduced from the previous section, the swelling ratio is indeed large and is independent of the Reynolds number. This is not the case when we switch gravity on. As the Reynolds increases (but still remains very small) we observe that the swelling ratios are greatly reduced, as can be seen from the fluid flow visualizations in Figs. 19, 20 and 21. To avoid short wavelenght pertubations on th free surface in this simulation we employed a filter described in Mangiavacchi et al. [37].
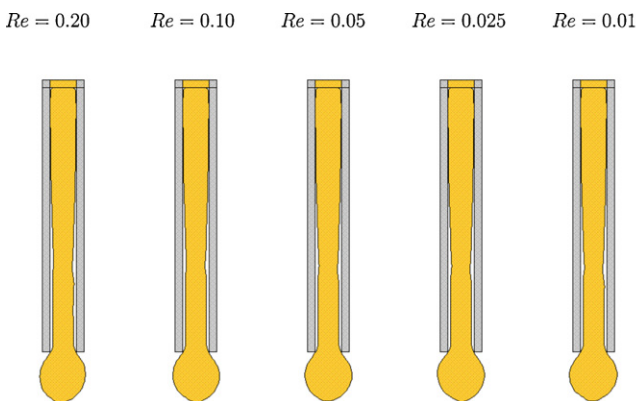

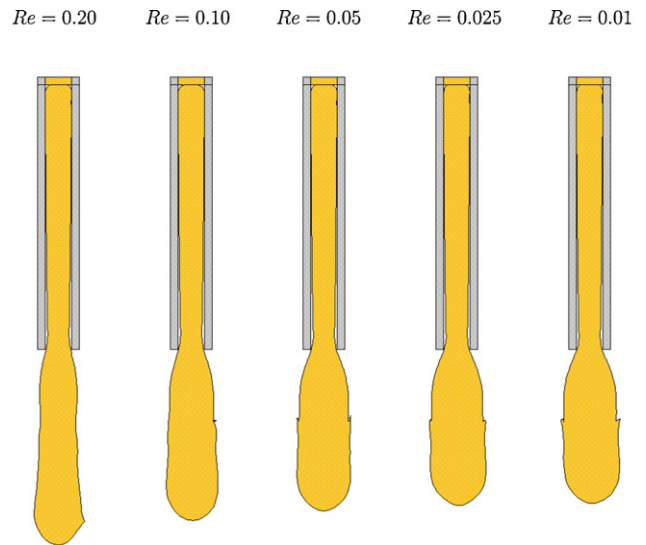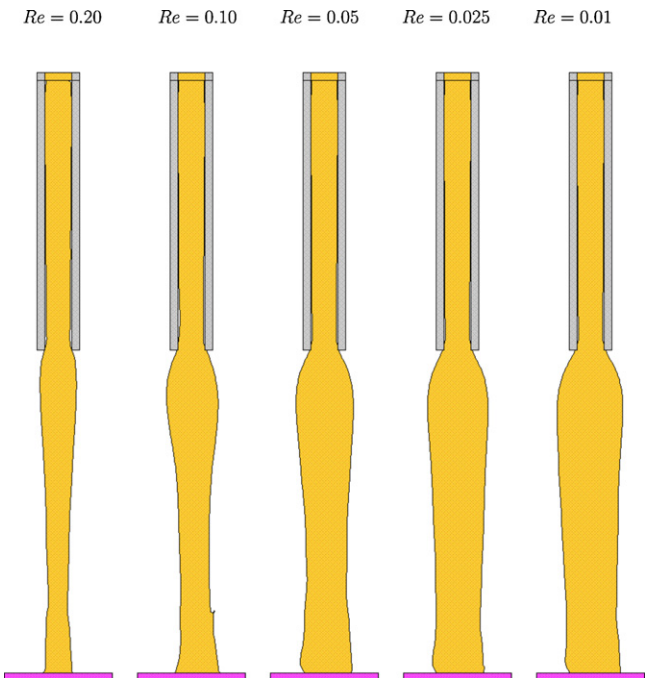**Fig. 21.** Numerical simulation of the Barus effect with gravity included. Fluid flow visualization after the jet has entered the outflow boundary. Results shown at times: $t = 0.78$ s ($Re = 0.20$), $t = 0.88$ s ($Re = 0.10$), $t = 0.96$ s ($Re = 0.05$), $t = 1.06$ s ($Re = 0.025$), $t = 1.12$ s ($Re = 0.01$).

8. Concluding remarks

This work has been concerned with an implicit numerical technique for simulating two-dimensional viscoelastic free surface flows. The viscoelastic model employed was the eXtended Pom-Pom (XPP) model. The solution strategy for the flow equations (conservation of mass and momentum) was essentially based on a projection method.

Fig. 19. Numerical simulation of the Barus effect with gravity included. Fluid flow visualization at $t = 0.34$ s.

First the equations were nondimensionalised. An intermediate fluid velocity was calculated by a Crank-Nicolson scheme and the resulting linear system solved by conjugate gradients with a diagonal preconditioner. A Poisson equation using implicit boundary conditions was solved for a velocity potential which then allowed the divergence free updated velocity to be calculated. The non-Newtonian extra-stress tensor was calculated by a second order Runge-Kutta method employing this updated velocity. An updated pressure was then able to be calculated explicitly. Finally, the new position of the moving free surface, defined by the virtual marker particles, was obtained by solving $d\mathbf{x}_p/dt = \mathbf{u}_p$ using Euler's method for each particle.

The algorithm was partially validated by solving channel flow on five different meshes and results, showing the convergence of the free surface location, were presented by simulating extrudate swell using mesh refinement. Extrudate swell from a capillary was then computed and a reasonably comprehensive sensitivity analysis was performed on all the parameters that characterize the XPP model to determine how they influence the extrudate swelling ratio. Results were obtained for Weissenberg numbers up to 20, but the code appeared to suffer from numerical instability thereafter. Finally, armed with this knowledge, we were able to exhibit the Barus effect when the swelling jet is subject to gravity effects in the direction of the flow; we also showed that gravity could play a significant role in reducing the swelling ratio.

Acknowledgments

Appendix A. Supplementary Data

Supplementary data associated with this article can be found, in the online version, at doi:10.1016/j.jnnfm.2010.11.001.

References

[1] M. Aboubacar, H. Matallah, H.R. Tamaddon-Jahromi, M.F. Webster, Highly elastic solutions for Oldroyd-B and Phan-Thien/Tanner fluids with a finite volume/element method: planar contraction flows, J. Non-Newton. Fluid Mech. 103 (2002) 65–103.
[2] M. Aboubacar, H.R. Tamaddon-Jahromi, M.F. Webster, Time-dependent algorithms for viscoelastic flow: bridge between finite-volume and finite-element methodology, in: Second MIT Conference on Computational Fluid and Solid Mechanics, 2003, pp. 815–818.
[3] M. Aboubacar, J.P. Aguayo, P.M. Phillips, T.N. Phillips, H.R. Tamaddon-Jahromi, B.A. Snigerev, M.F. Webster, Modelling Pom-Pom type models with high-order finite volume schemes, J. Non-Newton. Fluid Mech. 126 (2005) 207–220.
[4] A. Afonso, P.J. Oliveira, F.T. Pinho, M.A. Alves, The log-conformation tensor approach in the finite-volume method framework, J. Non-Newton. Fluid Mech. 157 (2009) 55–65.
[5] J.P. Aguayo, H.R. Tamaddon-Jahromi, M.F. Webster, Extensional response of the Pom-Pom model through planar contraction flows for branched polymer melts, J. Non-Newton. Fluid Mech. 134 (2006) 105–126.
[6] J.P. Aguayo, P.M. Phillips, T.N. Phillips, H.R. Tamaddon-Jahromi, B.A. Snigerev, M.F. Webster, The numerical prediction of planar viscoelastic contraction flows using the Pom-Pom model and higher-order finite volume schemes, J. Comput. Phys. 220 (2007) 586–611.
[7] M.A. Alves, P.J. Oliveira, F.T. Pinho, A convergent and universally bounded interpolation scheme for the treatment of advection, Int. J. Numer. Meth. Fluids 41 (2003) 47–75.
[8] A.A. Amsden, F.H. Harlow, A simplified MAC technique for incompressible fluid flow calculations, J. Comput. Phys. 6 (1970) 332–335.
[9] G.K. Batchelor, An Introduction to Fluid Dynamics, Cambridge, 1970.
[10] R.B. Bird, R.C. Armstrong, O. Hassager, Dynamics of Polymeric Liquids, vol. I, Willey, 1987.
[11] G.B. Bishko, O.G. Harlen, T.C.B. McLeish, T.M. Nicholson, Numerical simulation of the transient flow of branched polymer melts through a planar contraction using the Pom-Pom model, J. Non-Newton. Fluid Mech. 82 (1999) 255–273.
[12] A. Bonito, M. Picasso, M. Laso, Numerical simulation of 3D viscoelastic flows with free surfaces, J. Comput. Phys. 215 (2006) 691–716.
[13] D.L. Brown, R. Cortez, M.L. Minion, Accurate projection methods for the incompressible Navier-Stokes equations, J. Comput. Phys. 168 (2001) 464–499.
[14] G.C. Buscaglia, A finite element analysis of rubber coextrusion using a power-law model, Int. J. Numer. Meth. Eng. 36 (1993) 2143–2156.
[15] Z. Cai, C.R. Westphal, An adaptive mixed least-squares finite element method for viscoelastic fluids of Oldroyd type, J. Non-Newton. Fluid Mech. 159 (2009) 72–80.
[16] H.C. Choi, J.H. Song, J.Y. Yoo, Numerical-simulation of the planar contraction flow of a Giesekus fluid, J. Non-Newton. Fluid Mech. 29 (1988) 347–379.
[17] A.J. Chorin, Numerical solution of the Navier-Stokes equations, J. Comput. Phys. 2 (1968) 745–762.
[18] N. Clemeur, R.P.G. Rutgers, B. Debbaut, On the evaluation of some differential formulations for the Pom-Pom constitutive model, Rheol. Acta 42 (2003) 217–231.
[19] M. Cloitre, T. Hall, C. Mata, D.D. Joseph, Delayed-die swell and sedimentation of elongated particles in wormlike micellar solutions, J. Non-Newton. Fluid Mech. 79 (1998) 157–171.
[20] M.J. Crochet, R. Keunings, Finite element analysis of die-swell of a highly elastic fluid, J. Non-Newton. Fluid Mech. 10 (1982) 339–356.
[21] M.J. Crochet, A.R. Davies, K. Walters, Numerical Simulation of Non-Newtonian Flow, Elsevier, Amsterdam, 1984.
[22] H. Demir, Numerical modelling of viscoelastic cavity driven flow using finite difference simulations, Appl. Math. Comput. 166 (2005) 64–83.
[23] F.M. Denaro, On the applications of the Helmholtz-Hodge decomposition in projection methods for incompressible flows with general boundary conditions, Int. J. Numer. Meth. Fluids 43 (2003) 43–69.
[24] Y. Fan, R.I. Tanner, N. Phan-Thien, Galerkin/least-square finite-element methods for steady viscoelastic flows, J. Non-Newton. Fluid Mech. 84 (1995) 233–256.
[25] J. Fang, R.G. Owens, L. Tacher, A. Parriaux, A numerical study of the SPH method for simulating transient viscoelastic free surface flows, J. Non-Newton. Fluid Mech. 139 (2006) 68–84.
[26] V. Ganvir, A. Lele, R. Thaokar, B.P. Gautham, Simulation of viscoelastic flows of polymer solutions in abrupt contractions using an arbitrary Lagrangian Eulerian (ALE) based finite element method, J. Non-Newton. Fluid Mech. 143 (2007) 157–169.
[27] J.L. Guermond, P. Minev, J. Shen, An overview of projection methods for incompressible flows, Comput. Method Appl. M 195 (2006) 6011–6045.
[28] F.H. Harlow, J.E. Welch, Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface, Phys. Fluids 8 (1965) 2182–2189.
[29] I. Hori, S. Okubo, On the normal stress effect and the Barus effect of polymer melts, J. Rheol. 24 (1980) 39–53.
[30] N.J. Inkson, T.C.B. McLeish, O.G. Harlen, D.J. Groves, Predicting low density polyethylene melt rheology in elongational and shear flows with "Pom-Pom" constitutive equations, J. Rheol. 43 (1999) 873–896.
[31] N.J. Inkson, T.N. Phillips, R.G.M. van Os, Numerical simulation of flow past a cylinder using models of XPP type, J. Non-Newton. Fluid Mech. 156 (2009) 7–20.
[32] R. Keunings, M.J. Crochet, Numerical simulation of the flow of a viscoelastic fluid through an abrupt contraction, J. Non-Newton. Fluid Mech. 14 (1984) 279–299.
[33] J.M. Kim, C. Kim, J.H. Kim, C. Chung, K.H. Ahn, S.J. Lee, High-resolution finite element simulation of 4:1 planar contraction flow of viscoelastic fluid, J. Non-Newton. Fluid Mech. 129 (2005) 23–37.
[34] X.L. Luo, R.I. Tanner, A streamline element scheme for solving viscoelastic flow problems. Differential constitutive-equations, J. Non-Newton. Fluid Mech. 21 (1989) 179–199.
[35] T.C.B. McLeish, R.G. Larson, Molecular constitutive equations for a class of branched polymers: the Pom-Pom polymer, J. Rheol. 42 (1998) 81–110.
[36] G. Mompean, M. Deville, Unsteady finite volume simulation of Oldroyd-B fluid through a three-dimensional planar contraction, J. Non-Newton. Fluid Mech. 72 (1997) 253–279.
[37] N. Mangiavacchi, A. Castelo, M.F. Tomé, J.A. Cuminato, M.L.B. de Oliveira, S. McKee, An effective implementation of surface tension using Marker and Cell method for axisymmetric and planar flows, SIAM J. Sci. Comput. 26 (2005) 1340–1368.
[38] C.M. Oishi, J.A. Cuminato, V.G. Ferreira, M.F. Tomé, A. Castelo, N. Mangiavacchi, S. McKee, A stable semi-implicit method for free surface flows, Trans. ASME J. Appl. Mech. 73 (2006) 940–947.
[39] C.M. Oishi, M.F. Tomé, J.A. Cuminato, S. McKee, An implicit technique for solving 3D low Reynolds number moving free surface flows, J. Comput. Phys. 227 (2008) 7446–7468.
[40] C.M. Oishi, J.A. Cuminato, J.Y. Yuan, S. McKee, Stability of numerical schemes on staggered grids, Numer. Lin. Alg. Appl. 15 (2008) 945–967.
[41] P.J. Oliveira, F.T. Pinho, G.A. Pinto, Numerical simulation of non-linear elastic flows with a general collocated finite-volume method, J. Non-Newton. Fluid Mech. 79 (1998) 1–43.
[42] R.G. Owens, T.N. Phillips, Computational Rheology, World Scientific Publishing Company, 2002.

[43] M. Pasquali, L.E. Scriven, Free surface flows of polymer solutions with models based on the conformation tensor, J. Non-Newton. Fluid Mech. 108 (2002) 363–409.

[44] G.S. de Paulo, M.F. Tomé, S. McKee, A marker-and-cell approach to viscoelastic free surface flows using the PTT model, J. Non-Newton. Fluid Mech. 147 (2007) 149–174.

[45] B. Perot, R. Nallapati, A moving unstructured staggered mesh method for the simulation of incompressible free-surface flows, J. Comput. Phys. 184 (2003) 192–214.

[46] N. Phan-Thien, R.I. Tanner, Viscoelastic finite volume method, Rheol. Ser. 8 (1999) 331–359.

[47] T.N. Phillips, A.J. Willians, Viscoelastic flow through a planar contraction using a semi-Lagrangian finite volume method, J. Non-Newton. Fluid Mech. 87 (1999) 215–246.

[48] D. Rajagopalan, R. Amstrong, R. Brown, Finite element methods for calculation of steady viscoelastic flow using constitutive equations with newtonian viscosity, J. Non-Newton. Fluid Mech. 36 (1990) 159–192.

[49] P. Rubio, M.H. Wagner, LDPE melt rheology and the Pom-Pom model, J. Non-Newton. Fluid Mech. 92 (2000) 245–259.

[50] G. Russo, T.N. Phillips, Numerical prediction of extrudate swell of branched polymer melts, Rheol. Acta 49 (2010) 657–676.

[51] T. Sato, S.M. Richardson, Explicit numerical simulation of time-dependent viscoelastic flow problems by a finite element/finite volume method, J. Non-Newton. Fluid Mech. 51 (1994) 249–275.

[52] I. Sirakov, A. Ainser, M. Haouche, J. Guillet, Three-dimensional numerical simulation of viscoelastic contraction flows using the Pom-Pom differential constitutive model, J. Non-Newton. Fluid Mech. 126 (2005) 163–173.

[53] R.I. Tanner, A theory of die-swell, J. Polym. Sci. 8 (1970) 2067–2078.

[54] R.I. Tanner, On the congruence of some network and Pom-Pom models, Koren-Aust. Rheol. J. 18 (2006) 9–14.

[55] M.F. Tomé, J.L. Doricio, A. Castelo, J.A. Cuminato, S. McKee, Solving viscoelastic free surface flows of a second-order-fluid using a marker-and-cell approach, Int. J. Numer. Meth. Fluids 53 (2007) 599–627.

[56] M.F. Tomé, S. McKee, GENSMAC: a computational marker-and-cell method for free surface flows in general domains, J. Comput. Phys. 110 (1994) 171–186.

[57] M.F. Tomé, A. Castelo, J.A. Cuminato, J. Murakami, R. Minghim, C.F. Oliveira, S. McKee, Numerical simulation of axisymmetric free surface flows, J. Comput. Phys. 157 (2000) 441–472.

[58] M. Tomé, N. Mangiavacchi, J.A. Cuminato, A. Castelo, S. McKee, A finite difference technique for simulating unsteady viscoelastic free surface flows, J. Non-Newton. Fluid Mech. 106 (2002) 61–106.

[59] M.F. Tomé, L. Grossi, A. Castelo, J.A. Cuminato, S. McKee, K. Walters, Die-swell, splashing drop and a numerical technique for solving the Oldroyd-B model for axisymmetric free surface flows, J. Non-Newton. Fluid Mech. 141 (2007) 148–166.

[60] M.F. Tomé, A. Castelo, V.G. Ferreira, S. McKee, A finite difference technique for solving the Oldroyd-B model for 3D-unsteady free surface flows, J. Non-Newton. Fluid Mech. 154 (2008) 179–206.

[61] M.F. Tomé, R.A. Silva, C.M. Oishi, S. McKee, Numerical solution of the Upper-Convected Maxwell model for three-dimensional free surface flows, Commun. Comput. Phys. 6 (2009) 367–395.

[62] D. Trebotich, P. Colella, G.H. Miller, A stable and convergent scheme for viscoelastic flow in contraction channels, J. Comput. Phys. 205 (2005) 315–342.

[63] R.G.M. van Os, T.N. Phillips, Efficient and stable spectral element methods for predicting the flow of an XPP fluid past a cylinder, J. Non-Newton. Fluid Mech. 129 (2005) 143–162.

[64] W.M.H. Verbeeten, G.W.M. Peters, F.P.T. Baaijens, Differential constitutive equations for polymer melts: the eXtended Pom-Pom model, J. Rheol. 45 (2001) 823–844.

[65] W.M.H. Verbeeten, G.W.M. Peters, F.P.T. Baaijens, Viscoelastic analysis of complex polymer melt flows using the eXtended Pom-Pom model, J. Non-Newton. Fluid Mech. 108 (2002) 301–326.

[66] W.M.H. Verbeeten, G.W.M. Peters, F.P.T. Baaijens, Numerical simulations of the planar contraction flow for a polyethylene melt using the XPP model, J. Non-Newton. Fluid Mech. 117 (2004) 73–84.

[67] S.-C. Xue, N. Phan-Thien, R.I. Tanner, Numerical study of secondary flows of viscoelastic fluid in straight pipes by an implicit finite volume method, J. Non-Newton. Fluid Mech. 59 (1995) 191–213.

[68] B. Yang, A. Prosperetti, A second-order boundary-fitted projection method for free-surface flow computations, J. Comput. Phys. 231 (2006) 574–590.

[69] P. Wapperom, M.F. Webster, A second-order hybrid finite-element/volume method for viscoelastic flows, J. Non-Newton. Fluid Mech. 79 (1998) 405–431.

[70] K. Walters, M.F. Webster, The distinctive CFD challenges of computational rheology, Int. J. Numer. Meth. Fluids 43 (2003) 577–596.

[71] K. Walters, M.F. Webster, H.R. Tamaddon-Jahromi, The numerical simulation of some contraction flows of highly elastic liquids and their impact on the relevance of the Couette correction in extensional rheology, Chem. Eng. Sci. 64 (2009) 4632–4639.