V European Conference on Computational Fluid Dynamics
ECCOMAS CFD 2010
J. C. F. Pereira and A. Sequeira (Eds)
Lisbon, Portugal,14-17 June 2010

A FINITE DIFFERENCE TECHNIQUE FOR SOLVING THE SECOND ORDER CONSTITUTIVE EQUATION FOR THREE-DIMENSIONAL FREE SURFACE FLOWS

Murilo F. Tomé* and Igor Revoredo*

*Department of Applied Mathematics and Statistics
University of Sao Paulo,
Av. Trabalhador Saocarlense, 400, Sao Carlos, Sao Paulo, Brazil
e-mail: murilo@icmc.usp.br

Key words: Finite differences, Viscoelastic flows, Marker-and-Cell, Free surfaces, Second Order Fluid, Three-dimensional flows

Abstract. *This work presents a numerical technique for solving the Second Order Fluid constitutive equation for three-dimensional free surface flows. The governing equations are solved by the finite difference method on a 3D-staggered grid. The freee surface is modeled by a Marker-and-Cell type and the full free surface stress conditions are employed. The numerical method developed in this work is validated by comparing the numerical predictions obtained for the flow in a tube with the corresponding analytic solution for Second Order Fluids. By using mesh refinement, the convergence of the numerical technique is verified. The time-dependent extrudate swell is simulated for various values of the Deborah number.*

1 INTRODUCTION

Many processes in polymer industries involve free surface flows of complex fluids. For example, the extrusion and the filling of containers with fluids having complex rheology. These problems have motivated the development of numerical methods so that these problems can be studied via simulation. For instance, the rheological models Upper-Convected Maxwel (UCM), Oldroyd-B and Phan-Thien-Tanner (PTT) have been considered by many investigators and a variety of techniques for simulating viscoelastic flows using these models have been developed. See for example, Carew et al [11], Marchal and Crochet [16], Phillips and Williams [5], Xue et al. [7], Yoo and Na [8], among many other researchers. The problems treated have been the flow through a 4:1 contraction in two and three dimensions (e.g. Alves et al. [4], Phillips and Williams [5], Yoo and Na [8], Mompean and Deville [17], Xue et al. [7])). The extrudate swell problem has also been studied by many investigators (e.g. Brasseur et al. [10], Crochet and Keunings [18, 25], Tomé et al. [15], to cite only a few). Moreover, viscoelastic flows governed by the constitutive equation Criminale-Ericksen-Filbey (CEF), which is known as Second Order Fluid, has

been little studied. Among the works that deal with free surface flows using this constitutive equation we can mention the work of Gast and Ellington [20] who employed the FIDAP code [19] to simulate two-dimensional extrudate swell of a Second Order Fluid. However, Gast and Ellington [20] presented results for fluids with small elasticity where the Deborah number (De) was of order 0.1. More recently, Tomé et al. [9] presented a numerical method for solving two-dimensional free surface flows governed by the Second Order Fluid constitutive equation. More specifically, Tomé et al. [9] employed marker particles [21] and developed a finite difference technique for solving the governing equations. The full free surface stress conditions were treated in details. Results for the flow in a 4:1 contraction and the extrudate swell of fluids with high viscoelasticity were obtained.

The aim of this work is to extend the technique presented by Tomé et al. [9] to three-dimensional flows of Second Order Fluids. We use the Marker-and-Cell method to represent the fluid [1] and present a finite difference technique for solving the governing equations. The numerical method developed is then validated against an analytic solution for steady tube flow of Second Order Fluids. Numerical results include the simulation of the transient extrudate swell of Second Order Fluids for several values of the Deborah number.

2 PROBLEM FORMULATION

The governing equations for incompressible flows of a Second Order Fluid are the mass conservation (1), the momentum equation (2) and the constitutive equation (3), as follows:

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

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

$$\boldsymbol{\tau} = \eta_0 \left[\mathbf{D} + \lambda_2 \overset{\nabla}{\mathbf{D}} + \lambda_4 \left(\mathbf{D} \cdot \mathbf{D}\right)\right]\,, \tag{3}$$

where $\mathbf{u}$ is the velocity field, ρ is the fluid density, p is the pressure field, $\mathbf{g}$ is the gravity field, $\boldsymbol{\tau}$ is the extra-stress tensor. The rate-of-strain tensor $\mathbf{D}$ and the upper convected derivative $\overset{\nabla}{\mathbf{D}}$ are given by

$$\mathbf{D} = \nabla \mathbf{u} + (\nabla \mathbf{u})^T, \quad \overset{\nabla}{\mathbf{D}} = \frac{\partial \mathbf{D}}{\partial t} + \nabla \cdot (\mathbf{u}\mathbf{D}) - \left[(\nabla \mathbf{u})^T \cdot \mathbf{D} + \mathbf{D} \cdot \nabla \mathbf{u}\right]\,, \tag{4}$$

respectively. In the constitutive equation (3), η_0 denotes the viscosity of the fluid and the parameters λ_2 and λ_4 represent material properties. To solve equations (2)-(3) we write the extra-stress tensor $\boldsymbol{\tau}$ as a sum of a Newtonian tensor $\eta_0 \mathbf{D}$ and a Non-Newtonian tensor $\eta_0 \boldsymbol{\Phi}$

$$\boldsymbol{\tau} = \eta_0 \big[\mathbf{D} + \boldsymbol{\Phi}\big] \tag{5}$$

where $\mathbf{\Phi}$ is given by

$$\mathbf{\Phi} = \lambda_2 \overset{\nabla}{\mathbf{D}} + \lambda_4(\mathbf{D}\cdot\mathbf{D}) \; . \tag{6}$$

Introducing (5) into the momentum equation (2) it becomes

$$\frac{\partial \mathbf{u}}{\partial t} + \nabla\cdot(\mathbf{u}\mathbf{u}) = -\nabla p + \left(\frac{\eta_0}{\rho}\right)\left[\nabla^2\mathbf{u} + \nabla\cdot\mathbf{\Phi}\right] + \mathbf{g}, \tag{7}$$

where p denotes the pressure per unit of density.

2.1 Basic Equations

We consider three-dimensional Cartesian free surface flows and use the notation

$$\mathbf{u} = \begin{bmatrix} u \\ v \\ w \end{bmatrix}, \qquad \mathbf{D} = \begin{bmatrix} D^{xx} & D^{xy} & D^{xz} \\ D^{xy} & D^{yy} & D^{yz} \\ D^{xz} & D^{yz} & D^{zz} \end{bmatrix}, \qquad \boldsymbol{\tau} = \begin{bmatrix} \tau^{xx} & \tau^{xy} & \tau^{xz} \\ \tau^{xy} & \tau^{yy} & \tau^{yz} \\ \tau^{xz} & \tau^{yz} & \tau^{zz} \end{bmatrix} \; . \tag{8}$$

Let L and U be length and velocity scales, respectively. We employ the nondimensionalization

$$\bar{\mathbf{x}} = \frac{\mathbf{x}}{L}, \qquad \bar{\mathbf{u}} = \frac{\mathbf{u}}{U}, \qquad \bar{p} = \frac{p}{\rho U^2}, \qquad \bar{\mathbf{g}} = \frac{\mathbf{g}}{g}, \qquad \bar{t} = \frac{U}{L}t.$$

Therefore, equations (1), (7) and (6) can be written as (the bars have been dropped for clarity)

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0 \tag{9}$$

$$\begin{aligned} \frac{\partial u}{\partial t} + \frac{\partial (u^2)}{\partial x} + \frac{\partial (uv)}{\partial y} + \frac{\partial (uw)}{\partial z} = -\frac{\partial p}{\partial x} + \frac{1}{Re}\Big[\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} \\ + \frac{\partial \Phi^{xx}}{\partial x} + \frac{\partial \Phi^{xy}}{\partial y} + \frac{\partial \Phi^{xz}}{\partial z}\Big] + \frac{1}{Fr^2} g_x \, , \end{aligned} \tag{10}$$

$$\begin{aligned} \frac{\partial v}{\partial t} + \frac{\partial (uv)}{\partial x} + \frac{\partial (v^2)}{\partial y} + \frac{\partial (vw)}{\partial z} = -\frac{\partial p}{\partial y} + \frac{1}{Re}\Big[\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} + \frac{\partial^2 v}{\partial z^2} \\ + \frac{\partial \Phi^{xy}}{\partial x} + \frac{\partial \Phi^{yy}}{\partial y} + \frac{\partial \Phi^{yz}}{\partial z}\Big] + \frac{1}{Fr^2} g_y \, , \end{aligned} \tag{11}$$

$$\begin{aligned} \frac{\partial w}{\partial t} + \frac{\partial (uw)}{\partial x} + \frac{\partial (vw)}{\partial y} + \frac{\partial (w^2)}{\partial z} = -\frac{\partial p}{\partial z} + \frac{1}{Re}\Big[\frac{\partial^2 w}{\partial x^2} + \frac{\partial^2 w}{\partial y^2} + \frac{\partial^2 w}{\partial z^2} \\ + \frac{\partial \Phi^{xz}}{\partial x} + \frac{\partial \Phi^{yz}}{\partial y} + \frac{\partial \Phi^{zz}}{\partial z}\Big] + \frac{1}{Fr^2} g_z \, . \end{aligned} \tag{12}$$

$$\begin{aligned}\Phi^{xx} =& De\Bigg\{ \frac{\partial \mathrm{D}^{xx}}{\partial t} + \frac{\partial (u\mathrm{D}^{xx})}{\partial x} + \frac{\partial (v\mathrm{D}^{xx})}{\partial y} + \frac{\partial (w\mathrm{D}^{xx})}{\partial z} \\ & - 2\Bigg[\left(\frac{\partial u}{\partial x}\mathrm{D}^{xx}\right) + \left(\frac{\partial u}{\partial y}\mathrm{D}^{xy}\right) + \left(\frac{\partial u}{\partial z}\mathrm{D}^{xz}\right) \Bigg] \Bigg\} \\ & + \kappa\Big[(\mathrm{D}^{xx})^2 + (\mathrm{D}^{xy})^2 + (\mathrm{D}^{xz})^2\Big],\end{aligned} \tag{13}$$

$$\begin{aligned}\Phi^{xy} =& De\Bigg\{ \frac{\partial \mathrm{D}^{xy}}{\partial t} + \frac{\partial (u\mathrm{D}^{xy})}{\partial x} + \frac{\partial (v\mathrm{D}^{xy})}{\partial y} + \frac{\partial (w\mathrm{D}^{xy})}{\partial z} \\ & - \Bigg[\left(\frac{\partial v}{\partial x}\mathrm{D}^{xx}\right) + \left[\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right)\mathrm{D}^{xy}\right] + \left(\frac{\partial v}{\partial z}\mathrm{D}^{xz}\right) + \left(\frac{\partial u}{\partial y}\mathrm{D}^{yy}\right) \\ & + \left(\frac{\partial u}{\partial z}\mathrm{D}^{yz}\right) \Bigg] \Bigg\} + \kappa\Big[(\mathrm{D}^{xx}\mathrm{D}^{xy}) + (\mathrm{D}^{xy}\mathrm{D}^{yy}) + (\mathrm{D}^{xz}\mathrm{D}^{yz})\Big]\end{aligned} \tag{14}$$

$$\begin{aligned}\Phi^{xz} =& De\Bigg\{ \frac{\partial \mathrm{D}^{xz}}{\partial t} + \frac{\partial (u\mathrm{D}^{xz})}{\partial x} + \frac{\partial (v\mathrm{D}^{xz})}{\partial y} + \frac{\partial (w\mathrm{D}^{xz})}{\partial z} \\ & - \Bigg[\left(\frac{\partial w}{\partial x}\mathrm{D}^{xx}\right) + \left(\frac{\partial w}{\partial y}\mathrm{D}^{xy}\right) + \left[\left(\frac{\partial u}{\partial x} + \frac{\partial w}{\partial z}\right)\mathrm{D}^{xz}\right] + \left(\frac{\partial u}{\partial y}\mathrm{D}^{yz}\right) \\ & + \left(\frac{\partial u}{\partial z}\mathrm{D}^{zz}\right) \Bigg] \Bigg\} + \kappa\Big[(\mathrm{D}^{xx}\mathrm{D}^{xz}) + (\mathrm{D}^{xy}\mathrm{D}^{yz}) + (\mathrm{D}^{xz}\mathrm{D}^{zz})\Big]\end{aligned} \tag{15}$$

$$\begin{aligned}\Phi^{yy} =& De\Bigg\{ \frac{\partial \mathrm{D}^{yy}}{\partial t} + \frac{\partial (u\mathrm{D}^{yy})}{\partial x} + \frac{\partial (v\mathrm{D}^{yy})}{\partial y} + \frac{\partial (w\mathrm{D}^{yy})}{\partial z} \\ & - 2\Bigg[\left(\frac{\partial v}{\partial x}\mathrm{D}^{xy}\right) + \left(\frac{\partial v}{\partial y}\mathrm{D}^{yy}\right) + \left(\frac{\partial v}{\partial z}\mathrm{D}^{yz}\right) \Bigg] \Bigg\} \\ & + \kappa\Big[(\mathrm{D}^{xy})^2 + (\mathrm{D}^{yy})^2 + (\mathrm{D}^{yz})^2\Big]\end{aligned} \tag{16}$$

$$
\begin{aligned}
\Phi^{yz} =& De\Bigg\{ \frac{\partial \mathrm{D}^{yz}}{\partial t} + \frac{\partial (u\mathrm{D}^{yz})}{\partial x} + \frac{\partial (v\mathrm{D}^{yz})}{\partial y} + \frac{\partial (w\mathrm{D}^{yz})}{\partial z} \\
& - \Bigg[\left(\frac{\partial w}{\partial x}\mathrm{D}^{xy} \right) + \left(\frac{\partial v}{\partial x}\mathrm{D}^{xz} \right) + \left(\frac{\partial w}{\partial y}\mathrm{D}^{yy} \right) + \left[\left(\frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} \right) \mathrm{D}^{yz} \right] \\
& + \left(\frac{\partial v}{\partial z}\mathrm{D}^{zz} \right) \Bigg] \Bigg\} + \kappa \Big[(\mathrm{D}^{xy}\mathrm{D}^{xz}) + (\mathrm{D}^{yy}\mathrm{D}^{yz}) + (\mathrm{D}^{yz}\mathrm{D}^{zz}) \Big]
\end{aligned} \tag{17}
$$

$$
\begin{aligned}
\Phi^{zz} =& De\Bigg\{ \frac{\partial \mathrm{D}^{zz}}{\partial t} + \frac{\partial (u\mathrm{D}^{zz})}{\partial x} + \frac{\partial (v\mathrm{D}^{zz})}{\partial y} + \frac{\partial (w\mathrm{D}^{zz})}{\partial z} \\
& - 2\Bigg[\left(\frac{\partial w}{\partial x}\mathrm{D}^{xz} \right) + \left(\frac{\partial w}{\partial y}\mathrm{D}^{yz} \right) + \left(\frac{\partial w}{\partial z}\mathrm{D}^{zz} \right) \Bigg] \Bigg\} \\
& + \kappa \Big[(\mathrm{D}^{xz})^2 + (\mathrm{D}^{yz})^2 + (\mathrm{D}^{zz})^2 \Big].
\end{aligned} \tag{18}
$$

where

$$
\mathbf{D} = \begin{bmatrix} 2\frac{\partial u}{\partial x} & \frac{\partial u}{\partial xy} + \frac{\partial v}{\partial x} & \frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} \\ \frac{\partial u}{\partial xy} + \frac{\partial v}{\partial x} & 2\frac{\partial v}{\partial v} & \frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} \\ \frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} & \frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} & 2\frac{\partial w}{\partial z} \end{bmatrix}. \tag{19}
$$

In equations above, $Re = \frac{\rho U L}{\eta_0}, Fr = \frac{U}{\sqrt{Lg}}, De = \lambda_2 \frac{U}{L}$, are the Reynolds, Froude and Deborah numbers, respectively. The nondimensional number κ is defined by $\kappa = \lambda_4 \frac{U}{L}$.

To simulate three-dimensional free surface flow of a Second Order Fluid, we need to solve equations (9)-(19) subject to appropriate boundary conditions.

2.2 Boundary Conditions

On rigid boundaries we adopt the no-slip condition ($u = v = w = 0$) while at fluid entrances (inflows) the normal velocity is prescribed by $u_n = u_{inf}$ and the tangential velocities are zero ($u_{m1} = u_{m2} = 0$). On fluid exits (outflows) the homogeneous Neumann condition $\frac{\partial u}{\partial n} = \frac{\partial v}{\partial n} = \frac{\partial w}{\partial n} = \mathbf{0}$ (here n denotes a normal direction to the boundary and m_1, m_2 tangential directions).

We consider a viscous fluid flowing into a passive atmosphere so that the components of the stress tensor must be continuous throughout the free surface. Thus, if one neglect surface tension effects, the appropriate conditions on the free surface are (see Batchelor [2], page 153)

$$
\mathbf{n} \cdot (\boldsymbol{\sigma} \cdot \mathbf{n}) = 0 \ , \tag{20}
$$

$$
\mathbf{m1} \cdot (\boldsymbol{\sigma} \cdot \mathbf{n}) = 0 \ , \tag{21}
$$

$$
\mathbf{m2} \cdot (\boldsymbol{\sigma} \cdot \mathbf{n}) = 0 \ , \tag{22}
$$

where $\mathbf{n}$ is the outward unit normal vector to the free surface and $\mathbf{m1}, \mathbf{m2}$ are unit tangential vectors and $\boldsymbol{\sigma}$ is the total stress tensor $\boldsymbol{\sigma} = -p\mathbf{I} + \frac{1}{Re}\big[\mathbf{D} + \boldsymbol{\Phi}\big]$.

3 METHOD OF SOLUTION

To solve equations (9)-(19) together with the boundary conditions we employ the GENSMAC3D methodology of Tomé et al. [14] as follows.

Suppose that $\mathbf{u}(\mathbf{x}, t_n)$ is known and boundary conditions for velocity and pressure are given. Then, $\mathbf{u}(\mathbf{x}, t_{n+1})$ where $t_{n+1} = t_n + \delta t$ can be calculated by the following steps:

1. With $\mathbf{u}(\mathbf{x}, t_n)$ compute the rate-of-strain tensor $\mathbf{D}(\mathbf{x}, t_n)$ by (19) and then compute the components of the non-Newtonian tensor $\mathbf{\Phi}(\mathbf{x}, t_n)$ through equations (13)-(18).

2. Let $\widetilde{p}(\mathbf{x}, t_n)$ be a pressure field that satisfies the correct pressure condition on the free surface. This pressure field is calculated such as the normal stress condition (20) is satisfied.

3. Introduce $\mathbf{\Phi}(\mathbf{x}, t_n)$ and $\mathbf{D}(\mathbf{x}, t_n)$ into the momentum equations (10)-(12) and compute an intermediate velocity field $\widetilde{\mathbf{u}}(\mathbf{x}, t_{n+1})$ from

$$\frac{\partial \widetilde{\mathbf{u}}}{\partial t} = \Big\{ -\nabla \cdot (\mathbf{u}\mathbf{u}) - \nabla \widetilde{p} + \left(\frac{1}{Re}\right) \left[\nabla^2 \mathbf{u} + \nabla \cdot \mathbf{\Phi}\right] + \frac{1}{Fr^2}\mathbf{g} \Big\}^n. \tag{23}$$

This equation is solved by the explicit Euler method. It can be shown that $\widetilde{\mathbf{u}}(\mathbf{x}, t_{n+1})$ possesses the correct vorticity at time t_{n+1} but it does not satisfy the mass conservation equation (9) (see Tomé et al. [6]). Let us define

$$\mathbf{u}(\mathbf{x}, t_{n+1}) = \widetilde{\mathbf{u}}(\mathbf{x}, t_{n+1}) - \nabla \psi(\mathbf{x}, t_{n+1}) \tag{24}$$

where the function $\psi(\mathbf{x}, t_{n+1})$ obeys the Poisson equation

$$\nabla^2 \psi(\mathbf{x}, t_{n+1}) = \nabla \cdot \widetilde{\mathbf{u}}(\mathbf{x}, t_{n+1}). \tag{25}$$

Therefore, $\mathbf{u}(\mathbf{x}, t_{n+1})$ conserves mass and the vorticity remains inaltered.

4. Solve the Poisson equation (25).

5. Calculate the final velocity (24).

6. Compute the pressure by (see Tomé et al. [6])

$$p(\mathbf{x}, t_{n+1} = \widetilde{p}(\mathbf{x}, t_n) + \frac{\psi(\mathbf{x}, t_{n+1})}{\delta t}. \tag{26}$$

7. Update the marker particles positions. This last step find new positions for the markers by solving

$$\frac{\partial \mathbf{x}_P}{\partial t} = \mathbf{u}(\mathbf{x_P}, t_{n+1}), \tag{27}$$

for each particle $\mathbf{x}_P$. The fluid surface is defined by a piecewise linear surface composed of triangles and quadrilaterals that have marker particles on their vertices.

4 FINITE DIFFERENCE APPROXIMATION

The equations described in the procedure presented in the previous section are solved by the finite difference method as follows. A staggered grid is employed: the pressure p, potential function ψ and the non-Newtonian tensor $\mathbf{\Phi}$ are positioned at cell centres while the velocities u, v, w are located at cell faces (see figure 1a). In this work we intend to simulate flows with moving free surfaces so that a procedure to identify the fluid region and the fluid free surface is employed (see Castelo et al. [12] for details). To accommodate this, the cells within the mesh are defined as: EMPTY CELLS (E) - cells that do not contain fluid; FULL CELLS (F) - cells that contain fluid and do not have any face in contact with an E-cell face; SURFACE CELLS (S) - cells that contain fluid and contains at least one face in contact with an E-cell face; INFLOW CELLS (I) - cells that define an inflow boundary; OUTFLOW CELLS (O) - cells that define an outflow boundary; BOUNDARY CELLS (B) - cells that define a rigid boundary. An example of these cells is displayed in figure 1b. The rate-of-strain tensor equation (19), the components of the non-Newtonian

a) **b)**

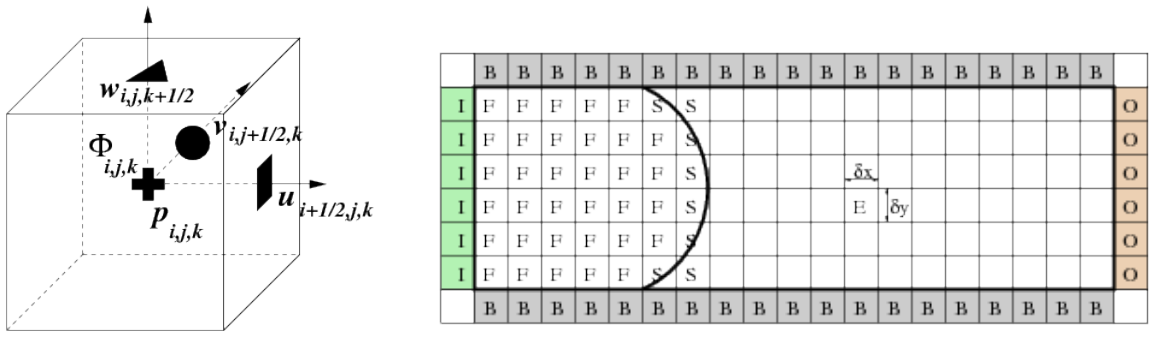


Figure 1: (a) Computational cell used; (b) type of cells employed by GENSMAC3D.

tensor equations (13)-(18) and the Poisson equation (25) are discretized at cell centres. The velocities u, v, w are approximated at the nodes $(i + \frac{1}{2}, j, k), (i, j + \frac{1}{2}, k), (i, j, +\frac{1}{2})$, respectively. The time derivative is discretized by the explicit Euler method while the Laplacian operator is approximated by second-order differences. The pressure gradient and the divergence of the non-Newtonina tensor $\mathbf{\Phi}_{i,j,k}$ are discretized by central differences. The convective terms are approximated by the high order upwind CUBISTA scheme of Alves et al. [3]. For instance, the x-momentum equation (10) is calculated by the following difference equation

$$
\begin{aligned}
\widetilde{u}^{n+1}_{i+\frac{1}{2},j,k} = u^{n}_{i+\frac{1}{2},j,k} + \delta t \Bigg\{ & -\mathcal{C}(uu)\Big|_{i+\frac{1}{2},j,k} - \mathcal{C}(vu)\Big|_{i+\frac{1}{2},j,k} - \mathcal{C}(wu)\Big|_{i+\frac{1}{2},j,k} \\
& - \left(\frac{\widetilde{p}_{i+1,j,k} - \widetilde{p}_{i,j,k}}{\delta x} \right) + \frac{1}{Re} \Bigg[\frac{u_{i+\frac{3}{2},j,k} - 2u_{i+\frac{1}{2},j,k} + u_{i-\frac{1}{2},j,k}}{\delta x^2} \\
& + \frac{u_{i+\frac{1}{2},j+1,k} - 2u_{i+\frac{1}{2},j,k} + u_{i+\frac{1}{2},j-1,k}}{\delta y^2} + \frac{u_{i+\frac{1}{2},j,k+1} - 2u_{i+\frac{1}{2},j,k} + u_{i+\frac{1}{2},j,k-1}}{\delta z^2} \\
& + \frac{\Phi^{xx}_{i+1,j,k} - \Phi^{xx}_{i,j,k}}{\delta x} + \frac{\Phi^{xy}_{i+\frac{1}{2},j+\frac{1}{2},k} - \Phi^{xy}_{i+\frac{1}{2},j-\frac{1}{2},k}}{\delta y} + \frac{\Phi^{xz}_{i+\frac{1}{2},j,k+\frac{1}{2}} - \Phi^{xz}_{i+\frac{1}{2},j,k-\frac{1}{2}}}{\delta z} \Bigg] \\
& + \frac{1}{Fr^2} g_x \Bigg\}^{n}
\end{aligned}
\tag{28}
$$

respectively. In equations above, the terms $\mathcal{C}(uu)\Big|_{i+\frac{1}{2},j,k}, \cdots, \mathcal{C}(ww)\Big|_{i,j,k+\frac{1}{2}}$ represent finite difference approximations for the convective terms $\frac{\partial(uu)}{\partial x}, \cdots, \frac{\partial(ww)}{\partial z}$. These approximations are computed by the high order CUBISTA method [3]. Details of the finite difference equations involved in the implementation of the CUBISTA method for three-dimensional flows can be found in [13]. Some derivatives are calculated by terms which are not defined in cell positions. These are computed by averaging the closest neighbours, for instance, $\Phi^{xy}_{i+\frac{1}{2},j+\frac{1}{2},k}$ and $\Phi^{xy}_{i+\frac{1}{2},j-\frac{1}{2},k}$ are computed by

$$
\begin{aligned}
\Phi^{xy}_{i+\frac{1}{2},j+\frac{1}{2},k} &= \frac{\Phi^{xy}_{i,j,k} + \Phi^{xy}_{i+1,j,k} + \Phi^{xy}_{i,j+1,k} + \Phi^{xy}_{i+1,j+1,k}}{4}, \\
\Phi^{xy}_{i+\frac{1}{2},j-\frac{1}{2},k} &= \frac{\Phi^{xy}_{i,j,k} + \Phi^{xy}_{i+1,j,k} + \Phi^{xy}_{i,j-1,k} + \Phi^{xy}_{i+1,j-1,k}}{4}.
\end{aligned}
\tag{29}
$$

However, for cells that lie close to mesh boundaries the cross derivatives from the divergence of the non-Newtoninan tensor $\mathbf{\Phi}$ are calculated by a second order approximations. For instance, with respect to figure 2a, the derivative $\frac{\partial \Phi^{xy}}{\partial y}\Big|_{i+\frac{1}{2},j,k}$ is obtained by differentiating the second order interpolating polynomial of Φ^{xy} passing through the points $(i+\frac{1}{2},j,k), (i+\frac{1}{2},j+1,k), (i+\frac{1}{2},j+2,k)$ at the point $(i+\frac{1}{2},j,k)$. In this case, we obtain the formula

$$
\frac{\partial \Phi^{xy}}{\partial y}\Big|_{i+\frac{1}{2},j,k} = \frac{-3\Phi^{xy}_{i+\frac{1}{2},j,k} + 4\Phi^{xy}_{i+\frac{1}{2},j+1,k} - \Phi^{xy}_{i+\frac{1}{2},j+2,k}}{2\delta y}.
\tag{30}
$$

The expressions for the other cross derivatives of the components of $\mathbf{\Phi}$ are obtained similarly.

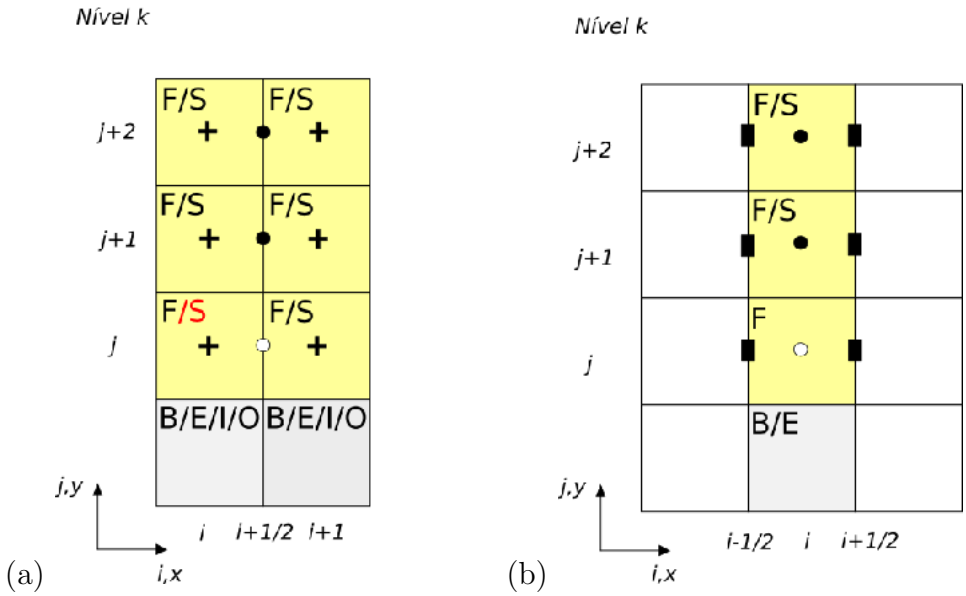


Figure 2: Stencil employed to calculate $\frac{\partial \Phi^{xy}}{\partial y}\Big|_{i+\frac{1}{2},j,k}$ by a second order difference approximation.

4.1 Computation of rate-of-strain tensor D and non-Newtonian tensor Φ

The rate-of-strain tensor is computed at the centre of cells by second order differences as follows. The diagonal components D^{xx}, D^{yy}, D^{zz} are straithforward calculated by

$$D^{xx}_{i,j,k} = 2\frac{\partial u}{\partial x}\Big|_{i,j,k}, \quad D^{yy}_{i,j,k} = 2\frac{\partial v}{\partial y}\Big|_{i,j,k}, \quad D^{zz}_{i,j,k} = 2\frac{\partial w}{\partial z}\Big|_{i,j,k}, \tag{31}$$

where

$$\frac{\partial u}{\partial x}\Big|_{i,j,k} = \frac{u_{i+\frac{1}{2},j,k} - u_{i-\frac{1}{2},j,k}}{\delta x}, \; \frac{\partial v}{\partial y}\Big|_{i,j,k} = \frac{v_{i,j+\frac{1}{2},k} - v_{i,j-\frac{1}{2},k}}{\delta y}, \; \frac{\partial w}{\partial z}\Big|_{i,j,k} = \frac{w_{i,j,k+\frac{1}{2}} - w_{i,j,k-\frac{1}{2}}}{\delta z}.$$

The off-diagonal components of **D** are obtained by

$$D^{xy}_{i,j,k} = \left[\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right]_{i,j,k}, \quad D^{xz}_{i,j,k} = \left[\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x}\right]_{i,j,k}, \quad D^{yz}_{i,j,k} = \left[\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y}\right]_{i,j,k}, \tag{32}$$

where the derivatives are calculated by central differences. For instance,

$$\frac{\partial u}{\partial y}\Big|_{i,j,k} \approx \frac{u_{i,j+\frac{1}{2},k} - u_{i,j-\frac{1}{2},k}}{\delta y}\,, \tag{33}$$

where

$$\begin{aligned} u_{i,j+\frac{1}{2},k} &= \frac{u_{i+\frac{1}{2},j,k} + u_{i-\frac{1}{2},j,k} + u_{i+\frac{1}{2},j+1,k} + u_{i-\frac{1}{2},j+1,k}}{4} \\ u_{i,j-\frac{1}{2},k} &= \frac{u_{i+\frac{1}{2},j,k} + u_{i-\frac{1}{2},j,k} + u_{i+\frac{1}{2},j-1,k} + u_{i-\frac{1}{2},j-1,k}}{4}\,. \end{aligned}$$

The other cross derivatives are obtained similarly. However, for cells that are situated near rigid boundaries or near the free surface, a second order formula similar to that employed for computing the divergence of the non-Newtonian tensor $\mathbf{\Phi}$ (see equation (30)) is used. For instance, related to figure 2b, the value of $\frac{\partial u}{\partial y}\Big|_{i,j,k}$ is calculated by

$$\frac{\partial u}{\partial y}\Big|_{i,j,k} \approx \frac{-3u_{i,j,k} + 4u_{i,j+1,k} - u_{i,j+2,k}}{2\delta y}, \tag{34}$$

where

$$u_{i,j,k} = \frac{u_{i+\frac{1}{2},j,k} + u_{i-\frac{1}{2},j,k}}{2}\ ,\ u_{i,j+1,k} = \frac{u_{i+\frac{1}{2},j+1,k} + u_{i-\frac{1}{2},j+1,k}}{2},\ u_{i,j+2,k} = \frac{u_{i+\frac{1}{2},j+2,k} + u_{i-\frac{1}{2},j+2,k}}{2}.$$

After computing the rate-of-strain tensor $\mathbf{D}$ for all interior cells in the mesh, the components of the non-Newtonian tensor $\mathbf{\Phi}$, on the centre of the cells, are calculated from equations (13)–(18). In this work we intend to simulate surface flows that reach a steady state so that in equations (13)–(18) the time derivative was neglected. For instance, the component $\Phi^{xx}_{i,j,k}$ was calculated by

$$\begin{aligned}\Phi^{xx}_{i,j,k} =& De\Bigg\{ \frac{\partial(u\mathrm{D}^{xx})}{\partial x}\Big|_{i,j,k} + \frac{\partial(v\mathrm{D}^{xx})}{\partial y}\Big|_{i,j,k} + \frac{\partial(w\mathrm{D}^{xx})}{\partial z}\Big|_{i,j,k} \\ &- 2\left[\left(\frac{\partial u}{\partial x}\mathrm{D}^{xx}\right)\Big|_{i,j,k} + \left(\frac{\partial u}{\partial y}\mathrm{D}^{xy}\right)\Big|_{i,j,k} + \left(\frac{\partial u}{\partial z}\mathrm{D}^{xz}\right)\Big|_{i,j,k}\right]\Bigg\} \\ &+ \kappa\Big[(\mathrm{D}^{xx})^2\big|_{i,j,k} + (\mathrm{D}^{xy})^2\big|_{i,j,k} + (\mathrm{D}^{xz})^2\big|_{i,j,k}\Big],\end{aligned} \tag{35}$$

The other components of $\mathbf{\Phi}$ are obtained similarly (see Revoredo [24] for details).

4.2 Approximate boundary conditions on the free surface

To apply the boundary conditions on the free surface according to equations (20)–(22), we suppose that the mesh spacing is small so that, locally, the free surface can be approximated by three types of linear surface as follows:

0^0 Planar surfaces: these surfaces are defined to have the outward unit normal vector parallel to one of the coordinate axis ($\mathbf{n} = (\pm 1, 0, 0)$ or $\mathbf{n} = (0, \pm 1, 0)$ or $\mathbf{n} = (0, 0, \pm 1)$). **$0^0$-Planar surfaces** are identified by SURFACE cells that have only one face in contact with EMPTY cells (figure 3(a) illustrates one example of **0^0-planar surface**).

45^0 Planar surfaces: these surfaces are defined to have the outward unit normal vector making a 45^0 angle with two coordinate axes so that in these surfaces the normal vector takes the form: $\big(\pm\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}, 0\big)$, $\big(\frac{\sqrt{2}}{2}, \pm\frac{\sqrt{2}}{2}, 0\big)$, $\big(\pm\frac{\sqrt{2}}{2}, 0, \frac{\sqrt{2}}{2}\big)$, $\big(\frac{\sqrt{2}}{2}, 0, \pm\frac{\sqrt{2}}{2}\big)$,

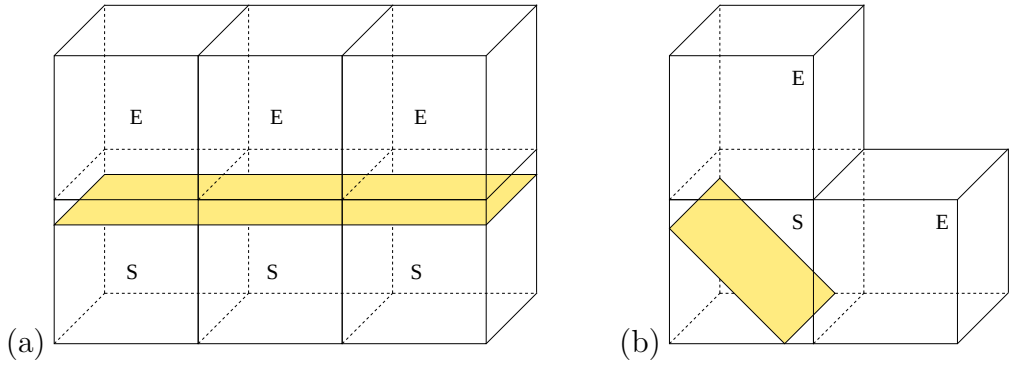


Figure 3: Examples of **0^0-Planar surface** (a) and **45^0-Planar surface** (b).

$\left(0, \pm\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}\right)$, $\left(0, \frac{\sqrt{2}}{2}, \pm\frac{\sqrt{2}}{2}\right)$. **$45^0$-Planar surfaces** are identified by SURFACE cells that possess only two adjacent faces contiguous with EMPTY cell faces (figure 3(b) displays one example of **45^0-Planar surfaces**).

60^0 -Planar surfaces: these surfaces are defined to have the outward unit normal vector making an angle of 60^0 with the coordinate axes (see figure 4). On these surfaces the outward normal vector takes the form $\left(\pm\frac{\sqrt{3}}{3}, \pm\frac{\sqrt{3}}{3}, \pm\frac{\sqrt{3}}{3}\right)$. They are regonized by SURFACE cells that have only three adjacent faces in contact with EMPTY cell faces.

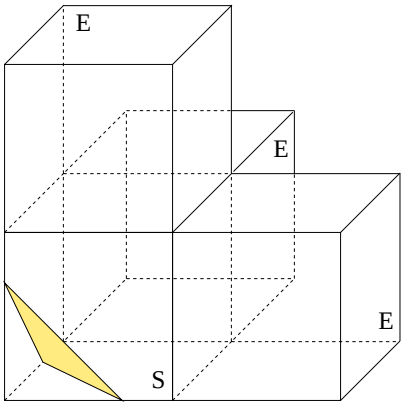


Figure 4: Example of a **60^0-Planar surface**.

It can be seen that in total there are 6 types of **0^0-Planar surfaces**, 12 kinds of **45^0-Planar surfaces** and 8 different categories of **60^0-Planar surfaces**. In this work we approximate equations (20)–(22) according to each case of planar surfaces. These approximations are the same employed by an Oldroyd-B fluid and the finite difference equations involved are presented by Tomé et al. [22]. Details on how the finite difference equations

are solved for each case of planar surface can be found in [24]. For lack of space they are not presented here.

5 Time step procedure

A procedure to optimize the time step size used in every calculation cycle is employed. The time-step is calculated according to the following restrictions:

(a) A fluid particle can not cross more than one cell during a calculational cycle. This leads to the restriction

$$\delta t < \frac{h}{|\mathbf{u}|} \text{ (componentwise)}, \text{ where } h = \min\{\delta x, \delta y, \delta z\}. \tag{36}$$

(b) Due to the explicit calculation of the intermediate velocity $\widetilde{\mathbf{u}}$, the following stability restriction is imposed

$$\begin{cases} \delta t = 0.25\, Re\, h^2, \text{ if } Re < 1\,, \\ \delta t = 0.25 h^2, \text{ otherwise.} \end{cases}$$

To satisfy (36) it is sufficient that

$$\delta t < \frac{h}{|U_{max}|}\,, \tag{37}$$

where U_{max} represents the maximum of the velocity $\mathbf{u}$ everywhere.

These restrictions are the same employed for Newtonian fluids so the implementation used is the same presented by Tomé et al. GENSMAC3D [14].

6 Validation results

To simulate three-dimensional flows governed by the Second Order Constitutive equation the finite difference equations presented in Section 4 were implemented into the Freeflow3D code [12]. Validation and convergence results are given next.

6.1 Fully developed flow of a second order fluid

As for Second Order Fluids the extra-stress tensor is obtained explicitly as a function of the velocity field then, if we consider fully developed flow in a tube (see figure 5) we can find an analytic solution for the non-Newtonian tensor $\mathbf{\Phi}$ according to equation (6). Indeed, if we suppose fully developed flow given by

$$\frac{\partial p}{\partial z} = P \text{ (constante)}, \ \mathbf{u} = (0, 0, w(x, y)), \text{ with } w(x, y) = \mathcal{A}\big(x^2 + y^2\big) + \mathcal{B},\ \mathcal{A}, \mathcal{B} \in R \tag{38}$$

then it can be verified that the components of $\mathbf{\Phi}$ are given by (see Revoredo [24])

$$\begin{aligned} &\Phi^{xx} = 4\kappa\, \mathcal{A}^2\, x^2\,, \qquad \Phi^{xy} = 4\kappa\, \mathcal{A}^2\, x\, y\,, \qquad \Phi^{xz} = 0\,, \\ &\Phi^{yy} = 4\kappa\, \mathcal{A}^2\, y^2\,, \qquad \Phi^{yz} = 0\,, \qquad \Phi^{zz} = 4\,(\kappa - 2De)\, \mathcal{A}^2 \big[x^2 + y^2\big]\,. \end{aligned} \tag{39}$$

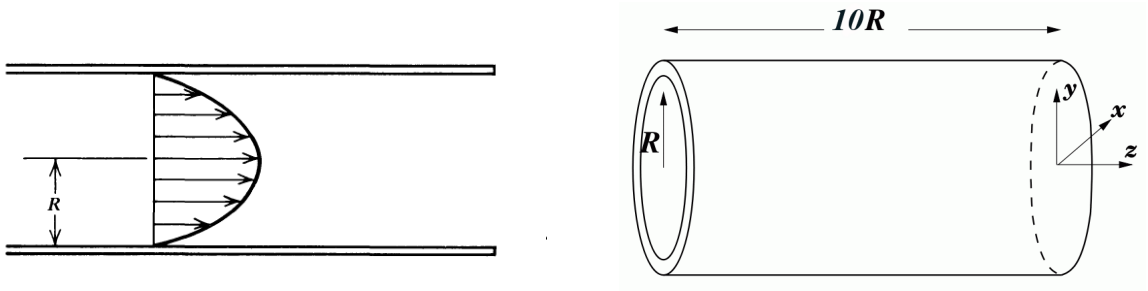


Figure 5: Flow in a tube: representation of flow domain

We point out that, under steady state, the analytic solutions given by equation (39) are valid in any cross-section of the tube.

To validate the numerical method presented in this work we simulated the flow in a tube (see figure 5) until steady state was stablished and we compared the numerical solutions with the respective analytic solutions given in equation (39). The input data employed were: $R = 1$, $U = 1$, $Re = 1$, $De = 0.4$, $\kappa = 0.4$, $g = 0$ (gravity was neglected). To verify the convergence of the numerical method, tube flow was simulated in three meshes:

- Mesh M0 - $\delta x = \delta y = \delta z = 0.16666667$ ($12 \times 12 \times 60$ computational cells);
- Mesh M1 - $\delta x = \delta y = \delta z = 0.125$ ($16 \times 16 \times 80$ computational cells);
- Mesh M2 - $\delta x = \delta y = \delta z = 0.1$ ($20 \times 20 \times 100$ computational cells).

On the walls of the tube the no-slip condition was imposed while at the tube entrance the fluid was injected until the tube was completely filed. At the tube entrance, a parabolic flow given by $u = v = 0$, $w(x, y) = \frac{2U}{R^2}\left[R^2 - (x^2 + y^2)\right]$ was employed while at the tube exit the homogeneous Neumann condition was imposed. At the begining there was a free surface where the boundary conditions were those presented in Section 4.2. The simulations started with the tube empty ($t = 0$) and ended at time $t = 50$. Figure 6 displays the variation of the velocity w at a cross section at the middle of the tube ($z = 5R$) as well as the velocity w along the plane xz passing on the centre of the tube ($y = 0$). We can see in figure 6 that the isolines are parallel indicating that steady state has established. To verify the correctness of the code and to show the convergence of the numerical method for solving the Second Order Fluid equations, we compared the numerical solutions with the respective analytic solutions given by equation (39) at the cross section $z = 5R$. Indeed, figure 7 displays the numerical solutions together with the analytic solutions on the three meshes. We can see that the numerical solutions approximate well the analytic solutions on the three meshes. To quantify this fact, we calculated the relative errors between the analytic and the numerical solutions by

$$E(Sol_{\text{Num}}) = \sqrt{\frac{\Sigma_{i,j}(Sol_{\text{Ex}} - Sol_{\text{Num}})^2}{\Sigma_{i,j}(Sol_{\text{Ex}})^2}}. \tag{40}$$

Table 1 shows the errors obtained on the three meshes while figure 8 illustrates the decay of the errors as a function of the mesh spacing. We can see in Table 1 that the errors decrease when the mesh refined. This result demonstrates the convergence of the numerical method presented in this work.

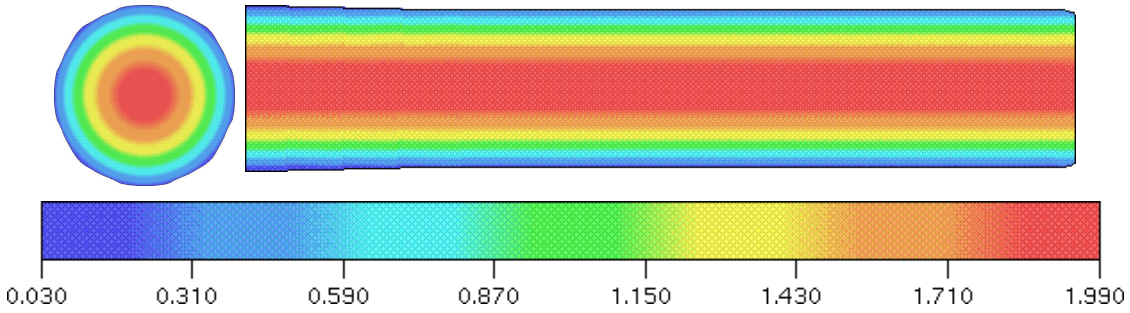


Figure 6: Isolines of $w(x, y)$ at a cross section $z = 5R$ and along the plane xz passing at the centre of the tube ($y = 0$). Result shown at time $t = 50$ and Mesh M2.

Table 1: Errors obtained on the three meshs.

Mesh	**M0**	**M1**	**M2**
$E(w(x,y))$	4.956×10^{-3}	2.696×10^{-3}	1.548×10^{-3}
$E(\Phi^{xx}(x,y))$	2.309×10^{-2}	1.099×10^{-2}	7.166×10^{-3}
$E(\Phi^{xy}(x,y))$	2.275×10^{-2}	1.082×10^{-2}	6.336×10^{-3}
$E(\Phi^{yy}(x,y))$	2.321×10^{-2}	1.078×10^{-2}	7.704×10^{-3}
$E(\Phi^{zz}(x,y))$	2.320×10^{-2}	1.037×10^{-2}	6.678×10^{-3}

7 Numerical simulation of the time-dependent extrudate swell

We applied the numerical method developed in this work to simulate the extrudate swell of a Second Order Fluid. In this problem, a viscous jet flows inside a tube of diameter D and length L until the jet reaches the tube exit from where it is extruded into the atmosphere where, under certain conditions, its diameter increases and attains a maximum value D_{max}. It is known that the higher is the elasticity of the fluid (represented by the Deborah number De) the greater is the extrudate rate defined by $S_r = D_{max}/D$. This phenomenon appears in many industrial applications so that many researchers have been studying this problem both experimentally and numerically (see for example the works [15],[22]-[27]).

To show that the numerical method described in this work is capable of simulating viscoelastic flows governed by the Second Order Fluid constitutive equation we simulated the time-dependent extrudate swell of a SOF jet. We considered an axisymmetric jet

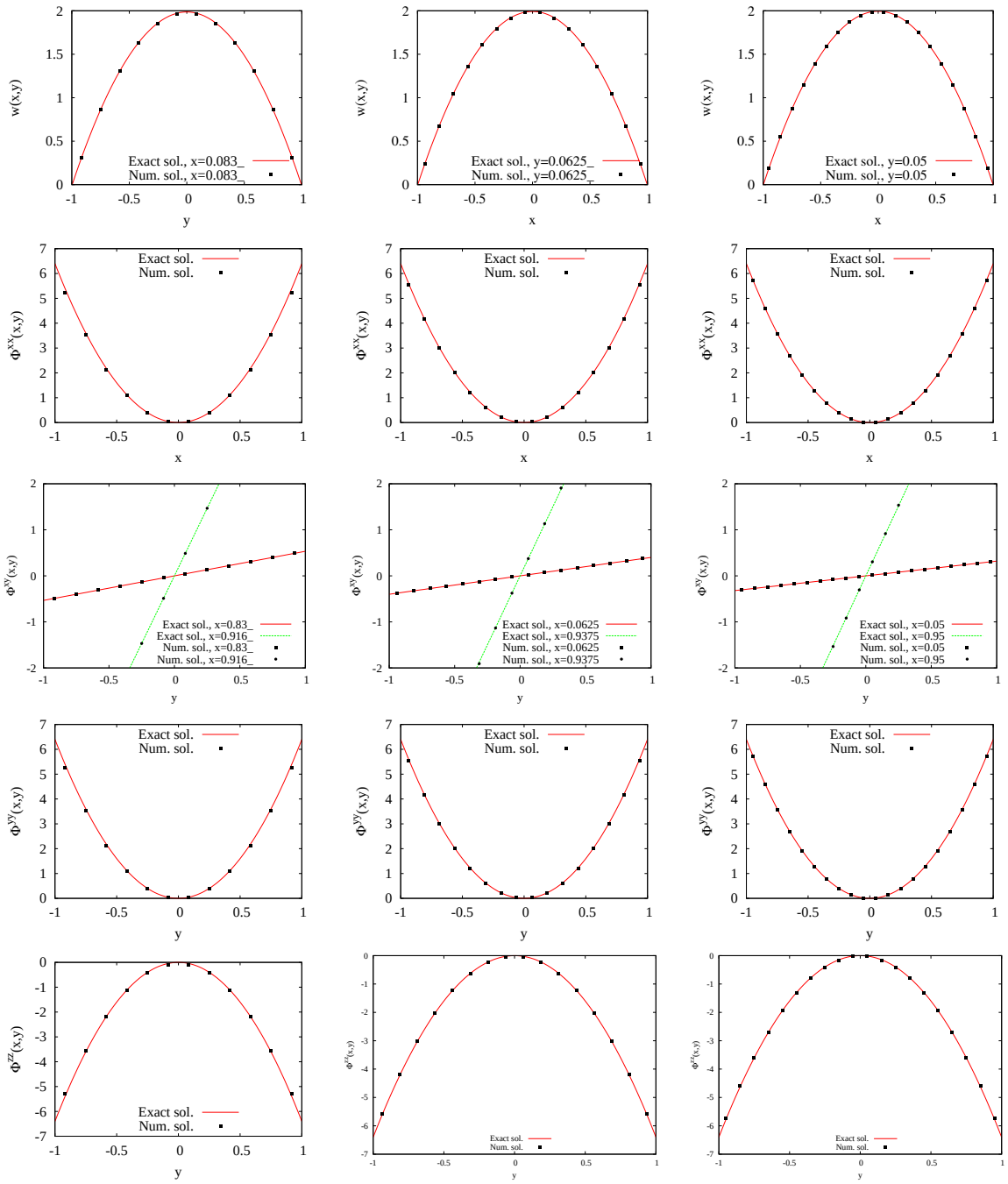


Figure 7: Comparison between the numerical and analytic solutions of $w(x, y)$, Φ^{xx}, Φ^{xy}, Φ^{yy}, Φ^{zz} on the three meshes. Left column: Mesh M0; Middle column: Mesh M1; Right column: Mesh M2.

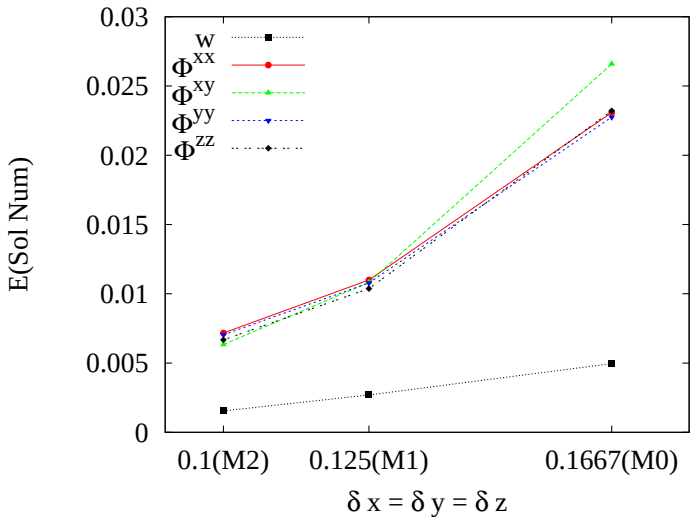
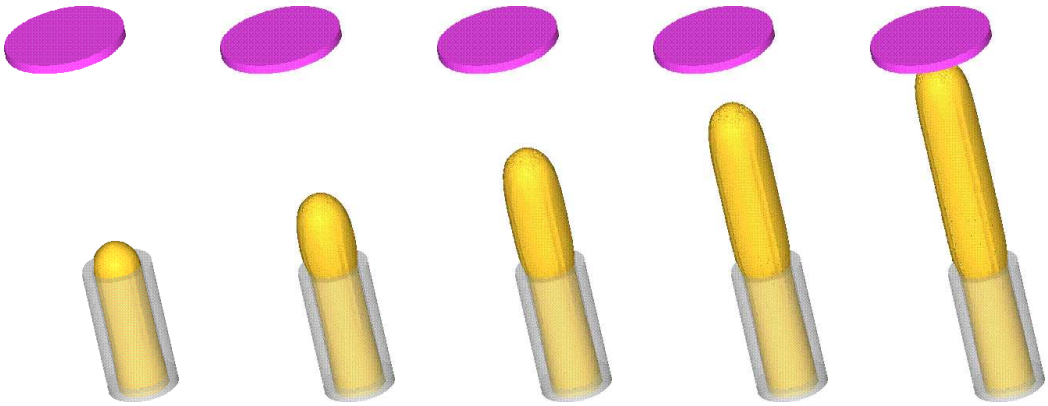


Figure 8: Descreasing of the errors with mesh refinement.

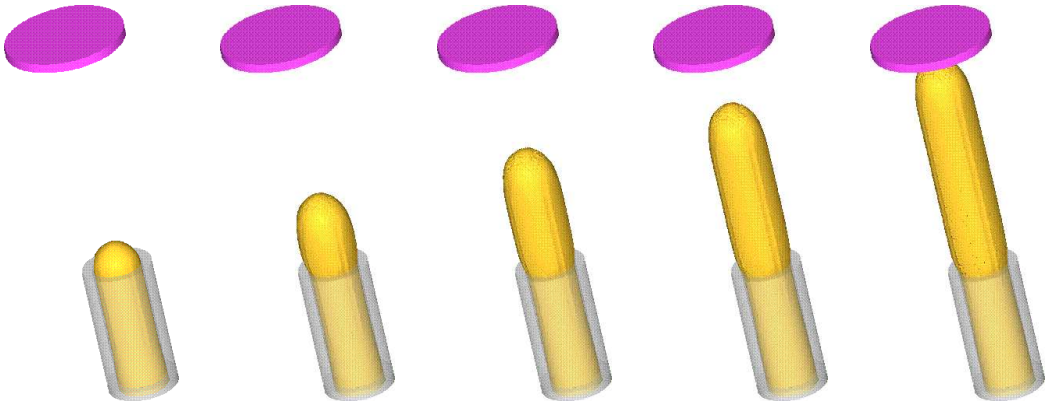
flowing inside a tube with $R = 1$ and lenght $L = 5$. At the tube entrance, a Poiseuille flow with velocity $w(x, y)$ given by equation (38) with $U = 1\text{ms}^{-1}$, $\mathcal{A} = \frac{-2U}{R^2}$, $\mathcal{B} = 2U$. The simulations started with the tube empty and the fluid was injected at the tube entrance until it filled the tube and reached the tube exit from where it was extruded into the atmosphere where it was expected that the jet would swell. An outflow boundary was positioned at a distance $2L$ far from the tube exit. A mesh with $32 \times 32 \times 120$ cells ($\delta x = \delta y = \delta z = 0.125$) was employed. We fixed the value of the Reynolds number and performed simulations for three values of De and κ as shown in Table 2. These simulations were performed from $t = 0$ until $t = 15$. Figure 9 displays a three-dimensional view of the fluid flow configuration at selected times. We can observe in figure 9 that as the Deborah number increases the swell becomes larger. Indeed, we considered the results obtained at $t = 15$ and computed the swell rate for each De as shown in Table 2. The extrudate swell rates obtained were $S_r = 1.0734, 1.0800, 1.2260$ for $De = 0.1, 0.2, 0.15$, respectively. These results are in agreement with the Deborah numbers employed as we expect that the elasticity of the fluid increases if the Deborah number is increased.

Table 2: Numerical simulation of extrudate swell of Second Order Fluids: extrudate rates obtained.

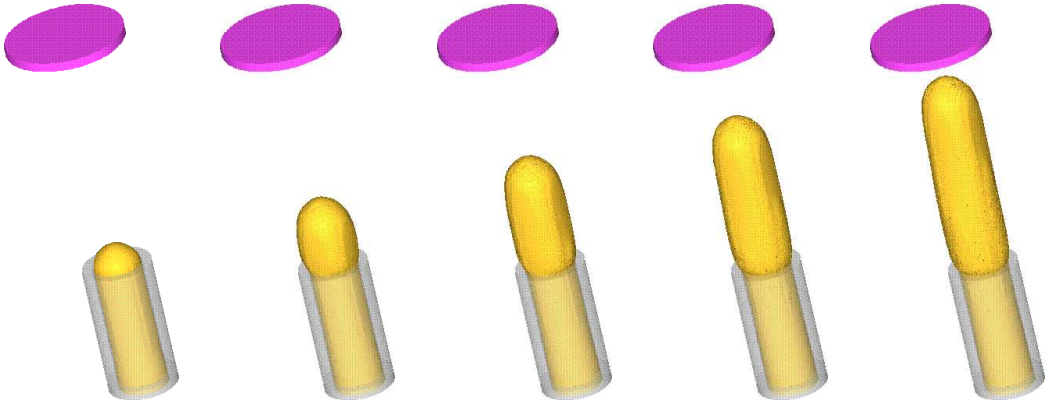
Re	De	κ	S_r
1.0	0.15	0.075	1.0734
1.0	0.20	0.10	1.0800
1.0	0.30	0.300	1.2260



$De = 0.15,\ \kappa = 0.075$



$De = 0.20,\ \kappa = 0.10$



$De = 0.30,\ \kappa = 0.30$

Figure 9: Numrical simulation of extrudate swell for various values of De and κ on mesh M1. Fluid flow configuration at times $t = 5, 7.5, 10, 12.5, 15$.

8 CONCLUDING REMARKS

This work presented the developments for obtaining a numerical method for simulating three-dimensional free surface flows governed by the Second Order Constitutive equation. The governing equations were presented in details as well as the nondimensionalization employed. The boundary conditions for the velocity and the free surface stress conditions were fully discussed. The solution of the governing equations by the finite difference method under the scope of the GENSMAC3D method was presented in details. The numerical method presented in this work was an extension of the two-dimensional technique of Tomé et al. [9] to three-dimensional flows. An analytic solution for tube flow of a Second Order Fluid was given. By performing mesh refinement on tube flow, the numerical solutions were compared with the respective analytic solutions. The results showed good agreement between the numerical and analytic solutions. To demonstrate that the numerical method developed in this work could deal with three-dimensional viscoelastic flows governed by the Second Order Constitutive equation we simulated the extrudate swell problem for various values of the material constants λ_2 and λ_4. Numerical results were obtained for Deborah number $De \leq 0.3$.

Although the numerical method proved capable of simulating 3D-free surface flows of Second Order Fluids, it can be improved in several ways: the momentum equations can be solved by an implicit technique; the pressure on the free surface can be calculated implicitly by a second order method.

9 ACKNOWLEDGEMENTS

The authors would like to acknowledge the financial support given by the Brazilian funding agencies: FAPESP - Fundação de Amparo a pesquisa do Estado de São Paulo (grant 04/16064-9) and CNPq - Conselho Nacional de Desenvolvimento Científico e Tecnológico (grants Nos. 304422/2007-0, 470764/2007-4).

REFERENCES

[1] A. Amsden and H. Welch, The SMAC method, *J. Comput. Physics*, **6** (1970), 322-325.

[2] G.K. Batchelor. *An Introduction to Fluid Dynamics*. Cambridge University Press, 2000.

[3] M. A. Alves, P. J. Oliveira, and F. T. Pinho, A convergent and universally bounded interpolation schemes for the treatment of advection, *Intern. J. Numer. Meth. Fluids*, **41** (2003), pp. 47–75 .

[4] M. A. Alves, P. J. Oliveira, and F. T. Pinho, Benchmark solutions for the flow of Oldroyd-b and PTT fluids in planar contractions, *J. Non-Newtonian Fluid Mech.*, **110** (2003), pp. 45–75.

[5] T.N. Phillips, and A.J. Williams, Viscoelastic flow through a planar contraction using a semi-Lagrangian finite volume method, *J. Non-Newtonian Fluid Mech.*, **87** (1999), pp. 215-246.

[6] M.F. Tomé, B. Duffy and S. McKee, A finite difference technique for simulating Non-Newtonian free surface flows, *J. Non-Newtonian Fluid Mech.*, **62** (1996), pp. 9-34.

[7] S.-C. Xue, N. Phan-Thien and R.I. Tanner, Three dimensional numerical simulations of viscoelastic flows through planar contractions, *J. Non-Newtonian Fluid Mech.*, **74** (1998), pp. 195-245.

[8] J.Y. Yoo and Y. Na, A numerical study of the planar contraction flow of a viscoelastic fluid using the SIMPLER algorithm, *J. Non-Newtonian Fluid Mech.*, **30** (1991), pp.89-106.

[9] 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, *Intern. J. Numer. Meth. Fluids*, **53** (2007), pp. 599-627.

[10] E. Brasseur, M.M. Fyrillas, G.C. Georgiou, M.J. Crochet, The time-dependent extrudate-swell problem of an Oldroyd-B fluid with slip along the wall, *J. Rheology.* **42** (1998), pp. 549-566.

[11] E.O.A. Carew, P. Townsend, M.F. Webster, A Taylor-Petrov-Galerkin algorithm for viscoelastic flow, *J. Non-Newtonian Fluid Mech.* **50** (1993), pp. 253-287.

[12] A. Castelo, M. F. Tomé, C. N. L. Cézar, S. McKee, and J. A. Cuminato. Freeflow: an integrated simulation system for three-dimensional free surface flows. *Computing and Visualization in Science*, **2** (2000), pp. 199–210.

[13] A. B. Costacurta. Estratégias "upwind" e modelagem k-epsilon para simulação numérica de escoamentos com superfícies livres em altos números de Reynolds. Dissertação de Mestrado, ICMC/USP, 2005.

[14] M. F. Tomé, A. Castelo, J. A. Cuminato, N. Mangiavacchi, and S. McKee. GENSMAC3D: A numerical method for solving unsteady three-dimensional free surface flows. *Intern. J. Numer. Meth. Fluids*, **37** (2001), pp. 747–796.

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

[16] J.M. Marchal, M.J. Crochet, A new mixed finite element for calculating viscoelastic flow, *J. Non-Newtonian Fluid Mech.*, **26** (1987), pp. 77-114.

[17] G. Mompean, M. Deville, Unsteady finite volume of Oldroyd-B fluid through a three-dimensional planar contraction, *J. Non-Newtonian Fluid Mech.*, **72** (1997), pp. 253-279.

[18] M.J. Crochet and R. Keunings, Die swell of a Maxwell fluid - numerical prediction, *J. Non-Newtonian Fluid Mech.*, **7** (1980), pp. 199-212.

[19] Fidap Manuals, Fluid Dynamics International, Evanstron, IL, 1995.

[20] L. Gast and W. Ellingson, Die swell measurements of second order fluids: numerical experiments, *Intern. J. Numer. Meth. Fluids*, **29** (1999), pp. 1-18.

[21] F. Harlow and J.E. Welch, Numerical Calculation of time-dependent viscous incompressible flow of fluid with a free surface, *Phys. Fluids*, **8** (1965), pp. 2182-2189.

[22] 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-Newtonian Fluid Mech.*, **154** (2008), pp. 179–206.

[23] R. I. Tanner. A theory of die-swell. *J. Polymer Science*, **8** (1970), pp. 2067–2078.

[24] I. Revoredo, Solução numérica da equação constitutiva Fluido de Segunda Ordem para escoamentos tridimensionais com superfíes livres, M.Sc dissertation, (2010).

[25] M. J. Crochet e R. Keunings. Finite element analysis of die-swell of a highly elastic fluid. *J. Non-Newtonian Fluid Mech.*, 10: 339–356, 1982.

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

[27] R. I. Tanner. A theory of die-swell revisited. *J. Non-Newtonian Fluid Mech.*, **129** (2005), pp. 85–87.