INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS, VOL. 11, 1015–1031 (1990)

AN EFFICIENT ALGORITHM FOR STRAIN HISTORY TRACKING IN FINITE ELEMENT COMPUTATIONS OF NON-NEWTONIAN FLUIDS WITH INTEGRAL CONSTITUTIVE EQUATIONS

X.-L. LUO AND E. MITSOULIS

Department of Chemical Engineering, University of Ottawa, Ottawa, Ontario, Canada K1N 9B4

SUMMARY

A new finite element technique has been developed for employing integral-type constitutive equations in non-Newtonian flow simulations. The present method uses conventional quadrilateral elements for the interpolation of velocity components, so that it can conveniently handle viscoelastic flows with both open and closed streamlines (recirculating regions). A Picard iteration scheme with either flow rate or elasticity increment is used to treat the non-Newtonian stresses as pseudo-body forces, and an efficient and consistent predictor–corrector scheme is adopted for both the particle-tracking and strain tensor calculations. The new method has been used to simulate entry flows of polymer melts in circular abrupt contractions using the K-BKZ integral constitutive model. Results are in very good agreement with existing numerical data. The important question of mesh refinement and convergence for integral models in complex flow at high flow rate has also been addressed, and satisfactory convergence and mesh-independent results are obtained. In addition, the present method is relatively inexpensive and in the meantime can reach higher elasticity levels without numerical instability, compared with the best available similar calculations in the literature.

KEY WORDS Non-Newtonian fluids Memory integral constitutive equations Polymer melts K-BKZ model Entry flow Vortex growth

1. INTRODUCTION

In the last few years rapid progress has been made in numerical simulation of non-Newtonian flows employing memory integral constitutive equations. Several independent studies[1–4] have demonstrated that much potential lies in memory integral models for the realistic simulation of viscoelastic fluids such as polymer melts. In particular, the K-BKZ integral model has been used quite successfully in several attempts to quantitatively predict the behaviour of real non-Newtonian fluids of commercial importance. Luo and Tanner[1,2] used a streamline finite element method (SFEM) to predict the extrudate swell of a fully characterized, low-density polyethylene (LDPE) melt from long[1] and short[2] circular dies. The K-BKZ integral-type model with multiple relaxation times used in their study provides good fits of both shear and elongational viscosity data. The predicted swelling ratio agreed well with experiments, and the well-known phenomenon of enhanced elastic recovery when extrusion takes place from short dies was very well captured by the simulation.[2] More recently, Luo and Mitsoulis[4] used the same method to study memory phenomena in extrudate swell of a high-density polyethylene (HDPE) melt from straight and tapered annular dies. The results showed that the diameter swell is highest for converging,

0271–2091/90/151015–17$08.50

Received 10 July 1989
Revised 10 January 1990

followed by straight and then diverging dies, in qualitative agreement with experimental findings in the literature, but in sharp contrast with previous simulations[5,6] with the Newtonian and Maxwell constitutive equations, which yielded the wrong trend (opposite to experimental findings).

The SFEM scheme developed by Luo and Tanner[1] consists of building special elements on the updated streamlines obtained from solving a Poisson equation for the streamfunction from the known velocity field. Since all element nodal points fall on a few open streamlines, the particle-tracking and strain tensor calculations are conveniently done on these streamlines by integrating upstream along them. The use of the special streamline elements makes the handling of memory integral models relatively easy, but on the other hand makes it difficult to deal with flows with recirculations, in which the domain cannot be adequately covered by open streamlines.

A more general approach has been taken by Dupont and Crochet.[3,7] For the particle tracking they use parametric equations in terms of a scalar parameter within each element to identify the trajectory of a material point. To calculate the strain history, a scalar function is introduced to relate the components of a material vector at different locations on the pathline. A differential equation for this scalar function has been derived and it can be solved numerically by integrating along the pathline. With this scalar function known, the material vector is fully determined and one is then able to form a complete set of scalar equations for calculating the components of the deformation gradient tensor **F**. The time integral for the stress is transformed into a line integral, and a Gaussian integration rule is employed on each segment. Employing this method and the same K-BKZ integral model as used by Luo and Tanner,[1] Dupont and Crochet[3] simulated the flow of an LDPE melt through abrupt circular contractions. Fitting shear viscosity and normal stress data obtained experimentally by White and Kondo,[8] they were able to successfully simulate vortex growth at relatively low flow rates (hence elasticity levels expressed as a dimensionless stress ratio S_R). The predicted vortex opening angle ϕ as a function of S_R agreed well with experimental measurements. However, their method failed to converge for values of $S_R > 1{\cdot}7$, although experimental measurements go well beyond this range. Another drawback of this method is its lack of efficiency and associated high cost.

Obviously, further work was needed to develop a low-cost numerical technique for fluids of the integral type, which can deal with recirculation and converge at high elasticity levels. This paper will present such a scheme and apply it to simulations of significance in non-Newtonian fluid mechanics. The important question of mesh refinement and convergence for integral models in complex flow at high elasticity level will also be addressed.

2. NUMERICAL METHOD FOR MEMORY INTEGRAL MODELS

The flow is governed by the conservation equations of mass and momentum. For an incompressible fluid under isothermal, creeping flow conditions ($Re = 0$) we have

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

$$\mathbf{0} = -\nabla p + \nabla \cdot \boldsymbol{\tau}, \tag{2}$$

where $\mathbf{v}$ is the velocity vector, $\boldsymbol{\tau}$ is the extra-stress tensor and p is the scalar pressure. The constitutive equation that relates $\boldsymbol{\tau}$ to the deformation history is a K-BKZ equation proposed by Papanastasiou *et al.*[9] and is written as

$$\boldsymbol{\tau}(t) = \int_{-\infty}^{t} \left[\sum \frac{a_k}{\lambda_k} \exp\left(-\frac{t-t'}{\lambda_k} \right) \right] \frac{\alpha}{(\alpha-3) + \beta I_{c^{-1}} + (1-\beta) I_c} \mathbf{C}_t^{-1}(t') \mathrm{d}t', \tag{3}$$

where t is the present time and t' is a time variable, λ_k and a_k are the relaxation times and relaxation modulus coefficients at a reference temperature T_0, α and β are material constants, and I_c and I_{c-1} are the first invariants of the Cauchy–Green tensor $\mathbf{C}_t$ and its inverse $\mathbf{C}_t^{-1}$, the Finger strain tensor.

The Maxwell model of the integral type is of particular importance in the development of numerical techniques for integral models, because it has a simple differential counterpart and because some analytical solutions exist in a few simple cases. The Maxwell integral model is given by

$$\tau(t)=\int_{-\infty}^{t}\left[\frac{\eta_m}{\lambda^2}\exp\left(-\frac{t-t'}{\lambda}\right)\right]\mathbf{C}_t^{-1}(t')\mathrm{d}t', \tag{4}$$

where η_m is the constant viscosity.

Special numerical schemes are required for the implementation of integral-type constitutive equations in finite elements. We will base our present study on the previous work by Luo and Tanner[1] and modify the SFEM scheme to overcome the problem of recirculation in such flows as the abrupt contraction. In particular, the finite element formulation for the conservation equations, the decoupled iterative algorithm for dealing with the non-Newtonian stresses and the basic approach to the calculation of the strain tensors will all be kept unchanged. These details have been fully described elsewhere[1] and hence we will concentrate our attention here on developing a new particle-tracking and strain tensor calculation scheme without the limitation of open streamlines.

2.1. *Particle tracking*

Since equation (3) takes into account the entire past history of a particle, it is essential to find numerically the particle position as a function of residence time $t-t'$. Previously in the SFEM formulation the task of particle tracking was simplified by the formation of the special streamline elements in which the element boundaries are the updated streamlines. In the presence of closed streamlines (recirculating regions) along with open streamlines it is not practical to build streamline elements in the whole flow domain, and it is more convenient to use conventional elements. In this case the particle tracking has to be done individually for every nodal point. For steady flows the particle acceleration can be expressed by the velocity field as

$$\mathbf{Dv}/\mathrm{D}t=\mathbf{v}\cdot\nabla\mathbf{v}. \tag{5}$$

Making use of the above expression, we propose here the following step-by-step particle-tracking scheme:

$$x'=x-u\mathrm{d}s+\frac{\mathrm{d}s^2}{2}\left(u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}\right)_0+\frac{\mathrm{d}s^2}{6}\left[\left(u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}\right)_1-\left(u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}\right)_0\right]+O(\mathrm{d}s^4), \tag{6}$$

$$y'=y-v\mathrm{d}s+\frac{\mathrm{d}s^2}{2}\left(u\frac{\partial v}{\partial x}+v\frac{\partial v}{\partial y}\right)_0+\frac{\mathrm{d}s^2}{6}\left[\left(u\frac{\partial v}{\partial x}+v\frac{\partial v}{\partial y}\right)_1-\left(u\frac{\partial v}{\partial x}+v\frac{\partial v}{\partial y}\right)_0\right]+O(\mathrm{d}s^4), \tag{7}$$

where (x', y') are the particle co-ordinates at time t' and (x, y) are the co-ordinates at $t'+\mathrm{d}s$ ($\mathrm{d}s>0$); the subscript '0' indicates the location at (x, y), while the subscript '1' indicates the location at (x_1, y_1) which are given by

$$x_1=x-u\mathrm{d}s+\frac{\mathrm{d}s^2}{2}\left(u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}\right)_0, \tag{8}$$

$$y_1 = y - v\,\mathrm{d}s + \frac{\mathrm{d}s^2}{2}\left(u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y}\right)_0. \tag{9}$$

In obtaining equations (6) and (7), the finite difference approximation for the particle derivative of the particle acceleration at (x, y) has been used:

$$\frac{\mathrm{d}^3 x}{\mathrm{d}s^3} = \frac{\mathrm{d}a}{\mathrm{d}s} = \frac{a_1 - a_0}{\mathrm{d}s} + O(\mathrm{d}s), \tag{10}$$

where a_0 and a_1 are the particle accelerations at (x_0, y_0) and (x_1, y_1) respectively. We point out that equations (6) and (7) only involve velocity components and their first derivative, yet they yield a third-order accuracy for each step of particle tracking, provided the velocity gradients are exact. The high order of accuracy in this scheme along with a small time step $\mathrm{d}s$ are essential for accurate particle tracking, since the maximum time lapse required by a multiple-relaxation-time model is usually large and the error in each time step may accumulate. Such is not the case when using the SFEM technique, where the entire particle pathline is predetermined from the Poisson equation solution.

Equations (6)–(9) give the particle location (x', y') as a function of the residence time, which is essential for the calculation of the deformation tensor $\mathbf{F}$ as a function of the residence time.

2.2. Strain tensor calculation

As described in Reference 1, the following relations are used in finding the strain history of a particle located at a nodal point at the present time t:

$$\mathbf{D}\mathbf{F}(s)/\mathrm{D}s = -\mathbf{L}(s)\mathbf{F}_t(s), \tag{11}$$

$$\mathbf{F}_t(s)|_{s=0} = \mathbf{I}, \tag{12}$$

$$\mathbf{C}_t^{-1}(s) = \mathbf{F}_t(s)^{-1}(\mathbf{F}_t(s)^{\mathrm{T}})^{-1}, \tag{13}$$

where $s = t - t'$ is the residence time of the particle, $\mathbf{L}(s)$ is the velocity gradient tensor, $\mathbf{F}(s)$ is the deformation gradient tensor relative to the present configuration, $\mathbf{C}^{-1}(s)$ is the Finger strain tensor and $\mathbf{I}$ is the unit tensor. Previously in the SFEM formulation, equation (11) was integrated upstream along element boundaries which are the streamlines. In the present case, where no special elements are constructed and the particle history tracking is done point by point, an efficient way to numerically integrate equation (11) is the key to the overall efficiency of the scheme. To be consistent with the particle-tracking procedure described in the previous section, an improved Euler method, i.e. the predictor–corrector formula, is chosen for calculating the tensor $\mathbf{F}$:

$$\mathbf{F}_t(t') = \mathbf{I} - (\mathrm{d}s/2)(\mathbf{L} + \mathbf{L}' - \mathrm{d}s\mathbf{L}'\mathbf{L}) + O(\mathrm{d}s^3), \tag{14}$$

where $\mathrm{d}s = t - t'$, $\mathbf{L}$ is the velocity gradient at (x, y) and $\mathbf{L}'$ is evaluated at the new location (x', y'). As explained in Reference 1, the time step $\mathrm{d}s$ is always controlled to be small (typically a small fraction of the characteristic time or the relaxation time in the flow), and for every time step we take the configuration at the current time as the reference configuration. In this way the starting deformation tensor for every time step is always exactly known (the unit tensor), thus avoiding error accumulation due to an incorrect initial value for $\mathbf{F}$. Also note that equation (11) contains only the velocity gradient tensor, and no higher derivatives of the velocity are involved. Furthermore, it is accurate to second order, one order less than the particle tracking, which is considered to be consistent. It is worth mentioning that although the accuracy achieved here is only second-order, while the Runge–Kutta integration employed previously in the SFEM

formulation[1] was fourth-order, we believe the overall performance of the improved Euler method used here will be as good as the Runge–Kutta method for the following reason: the accuracy for calculating **F** with both methods is limited in the same way by the accuracy of the numerical differentiation of the velocity, i.e. the calculation of **L**. In other words, the fourth-order accuracy of the Runge–Kutta integration is only 'apparent' and the second-order improved Euler method can be sufficiently accurate if the truncation error ($\sim O(\mathrm{d}s^3)$) is controlled to be less than the error introduced in the calculation of **L**. Note that errors in computing **L** cannot be reduced with smaller ds.

The component form of equation (14) for the axisymmetric case (r, z, θ) can be written as follows:

$$F_{rr}=1-\frac{\mathrm{d}s}{2}\left(\frac{\partial u}{\partial r}+\frac{\partial u'}{\partial r}\right)+\frac{\mathrm{d}s^2}{2}\left(\frac{\partial u'}{\partial r}\frac{\partial u}{\partial r}+\frac{\partial u'}{\partial z}\frac{\partial v}{\partial r}\right)+O(\mathrm{d}s^3), \tag{15}$$

$$F_{rz}=-\frac{\mathrm{d}s}{2}\left(\frac{\partial u}{\partial z}+\frac{\partial u'}{\partial z}\right)+\frac{\mathrm{d}s^2}{2}\left(\frac{\partial u'}{\partial r}\frac{\partial v}{\partial r}+\frac{\partial u'}{\partial z}\frac{\partial v}{\partial z}\right)+O(\mathrm{d}s^3), \tag{16}$$

$$F_{zr}=-\frac{\mathrm{d}s}{2}\left(\frac{\partial v}{\partial r}+\frac{\partial v'}{\partial r}\right)+\frac{\mathrm{d}s^2}{2}\left(\frac{\partial v'}{\partial r}\frac{\partial u}{\partial r}+\frac{\partial v'}{\partial z}\frac{\partial v}{\partial r}\right)+O(\mathrm{d}s^3), \tag{17}$$

$$F_{zz}=1-\frac{\mathrm{d}s}{2}\left(\frac{\partial v}{\partial z}+\frac{\partial v'}{\partial z}\right)+\frac{\mathrm{d}s^2}{2}\left(\frac{\partial v'}{\partial r}\frac{\partial u}{\partial z}+\frac{\partial v'}{\partial z}\frac{\partial v}{\partial z}\right)+O(\mathrm{d}s^3), \tag{18}$$

$$F_{\theta\theta}=r'/r, \tag{19}$$

where the primes indicate the values at location (x', y'). Note that the expression for $F_{\theta\theta}$ has no truncation error. On the centreline, where equation (19) becomes indeterminate, one can easily prove that

$$F_{\theta\theta}=F_{rr}. \tag{20}$$

When there is a solid wall present on which the particle velocity is zero, the following exact relation[10] can be used to directly compute the Cauchy–Green tensor over a finite time difference s:

$$\mathbf{C}=\mathbf{I}-s\mathbf{A}_1+\tfrac{1}{2}s^2\mathbf{A}_2, \tag{21}$$

where $\mathbf{A}_1$ and $\mathbf{A}_2$ are the first and second Rivlin–Ericksen tensors and $s=t-t'$ is the residence time.

Equation (14) gives only the 'local' relative deformation tensor concerning two endpoints on the current increment ds. In order to integrate the constitutive equation we need to calculate the 'global' relative deformation tensor concerning the Laguerre points[1] and the starting point, i.e. the nodal point. As was done in Reference 1, the following chain rule is repeatedly used to connect all the 'local' $\mathbf{F}(\mathrm{d}s)$ to obtain the desired 'global' deformation tensor $\mathbf{F}_t(t')$:

$$\mathbf{F}_{t_1'}(t_3')=\mathbf{F}_{t_2'}(t_3')\mathbf{F}_{t_1'}(t_2'), \quad t_3'<t_2'<t_1'. \tag{22}$$

The Finger strain tensor $\mathbf{C}^{-1}(s)$ can easily be computed from the tensor **F** according to equation (13).

2.3. *Velocity gradient field*

Apparently, both the particle-tracking and strain tensor computations need the calculation of the velocity gradient to be as accurate as possible. Unfortunately, the conventional finite element

interpolation for the velocity field generally yields discontinuous velocity gradients at element boundaries. To avoid this discontinuity, the usual approach is to take a simple average of the values calculated from the several neighbouring elements around the nodal point considered. However, we adopted here a slightly different approach to obtain a better average value of the velocity gradient.

Let $\mathbf{L}=\sum \Psi_i \mathbf{L}_e^i$ be the finite element representation for the velocity gradient field. Here the Ψ_i are linear interpolation functions. The Galerkin discrete form for the equation $\mathbf{L}=\mathrm{d}\mathbf{v}/\mathrm{d}\mathbf{x}$ can now be written as

$$\int_\Omega \Psi_i \left(\Psi_j \mathbf{L}^j - \frac{\mathrm{d}\Phi_j}{\mathrm{d}\mathbf{x}} \mathbf{v}^j \right) \mathrm{d}\Omega = 0, \tag{23}$$

where $\mathbf{v}^j$ is the known velocity value at the jth nodal point and the Φ_j are quadratic interpolation functions. The solution of equation (23) gives a continuous, smooth velocity gradient field, based on which one can more conveniently perform the particle-tracking and strain tensor calculations as described above.

2.4. Iterative scheme

When using an integral-type constitutive equation, successive substitution (Picard iteration) has been the basic scheme for pursuing the calculations since the evaluation of the integral stresses requires knowledge of the velocity field. Another point of major concern in non-Newtonian flow simulation is how to reach high flow rates or high elasticity levels from the Newtonian solution. For simple models like the Maxwell model it is always possible in principle to fix the flow rate while only increasing the relaxation time λ to advance the elasticity level, which is proportional to the value of λ. However, in actual simulations of experimental results the flow rate cannot be fixed and the material parameters are predetermined to fit the material properties. Thus the *flow rate increment scheme* is a natural way of reaching high flow rates or elasticity levels when simulating experimental results. In this case one starts at a low flow rate and further increases it step by step to reach any desired flow rate value. However, previous numerical experiments[4] have shown that this approach sometimes leads to divergence at high flow rate. This is partially due to the relatively large change in upstream velocity profiles when increasing the flow rate.

In our previous work[4] we adopted a more stable iterative scheme in which the flow rate was fixed while the elasticity level was increased step by step. We call this approach the *elasticity increment scheme*. In this approach an artificial parameter ω $(0 \leqslant \omega \leqslant 1)$ is introduced for controlling the elasticity level in the flow:

$$\boldsymbol{\tau}^* = \omega \boldsymbol{\tau} + (1-\omega)\boldsymbol{\tau}_0 \tag{24}$$

where $\boldsymbol{\tau}^*$ is the apparent stress which actually enters in the calculation, $\boldsymbol{\tau}$ is the actual non-Newtonian stress calculated from equation (3) and $\boldsymbol{\tau}_0$ is the Newtonian viscous stress. Compared with the flow rate increment scheme, the elasticity increment causes much smaller changes in the upstream velocity profile. For instance, if the flow rate is increased from one unit to two units, the relative velocity change will be about 100% in the fully developed upstream region, but if the flow rate is fixed at two units and ω is increased from zero to one, the relative change in velocity from Newtonian to viscoelastic profile will be much less than 100%. Nevertheless, as will be shown in the next section, in the present work both the flow rate and the elasticity increment schemes showed good convergence at high flow rates and the results from the two different iterative schemes agreed very well. The advantage of the flow rate increment scheme, when converged, is

that the total computing time required to get solutions at various flow rates is usually less than that of the elasticity increment scheme.

In the case of integral constitutive equations where the viscoelastic stresses are treated as pseudo-body forces, the basic direct substitution method (Picard iteration) can be written as

$$\mathbf{A}\mathbf{v}^{(i+1)} = \mathbf{B}(\mathbf{v}^{(i)}), \tag{25}$$

where $\mathbf{A}$ is the fluid (stiffness) matrix containing the Newtonian viscous contribution, $\mathbf{B}$ is the RHS (load) vector containing the viscoelastic stress contribution and $\mathbf{v}^{(i)}$ is the solution vector at the ith iteration. The above equation can be modified by the addition of a relaxation factor Ω to the previous iterative solution:

$$\mathbf{A}\mathbf{v}^{(*)} = \mathbf{B}(\mathbf{v}^{(i)}), \tag{26}$$

$$\mathbf{v}^{(i+1)} = \Omega\mathbf{v}^{(i)} + (1-\Omega)\mathbf{v}^{(*)}. \tag{27}$$

The relaxation factor Ω varies between zero and one, with $\Omega = 0$ being the standard Picard iteration case. Equations (26) and (27) apply to both the flow rate and the elasticity increment schemes.

3. ABRUPT CONTRACTION FLOW OF LDPE MELTS

We now proceed to solve some complex problems to test the ability and efficiency of the program. One of the most challenging problems in non-Newtonian flow simulations is no doubt the polymer entry flow through abrupt contractions. This problem has a simple flow geometry but at the same time exhibits significant viscoelastic effects. A better understanding of this flow will ultimately lead to improvement of predictions for such polymer flow processes as extrusion and injection moulding, where many contractions and expansions may be involved. Until now many attempts to solve this problem have concentrated on using the Maxwell-type differential constitutive equations and failed to give even qualitative predictions for polymer melts of industrial importance. Nevertheless, the upper-convected Maxwell model has both differential and integral forms and it is a good test to compare numerical results of integral schemes with those of its differential counterpart in complex flows where no analytical solution can be found. We first solved the 4:1 circular abrupt contraction problem using the integral Maxwell model (equation (4)) and the newly developed scheme, and the results were compared with those obtained previously by Luo and Tanner[11] using the differential form of the Maxwell model. We found that the two solutions were numerically identical in terms of streamline/vortex patterns and the non-Newtonian stress field. For details of the numerical solution see Reference 11.

It is believed that the recent work by Dupont and Crochet[3] mentioned earlier in this paper is the first successful attempt to simulate LDPE melts in abrupt contraction flow, although their calculation failed to converge beyond a moderate flow rate or elasticity level. Different finite element meshes were used in Reference 3 and the results showed satisfactory mesh independence. Regarding the results of Dupont and Crochet[3] as reliable and independent of mesh influence, we can further test our new integral method by solving the same 4:1 circular contraction flow problem with the same K-BKZ model and material parameter data. Table I shows values of the relaxation times λ_k and the relaxation modulus coefficients a_k along with the parameters α and β as used by Dupont and Crochet[3] in their work. Figure 1 plots the model predictions for shear and elongational viscosities and normal stresses. Dupont and Crochet[3] compared the resulting curves of shear viscosity η_S and first normal stress difference N_1 with those obtained experimentally by White and Kondo[8] for characterizing their LDPE melts. It was found that the shear properties based on parameters in Table I and equation (3) compared satisfactorily with the experimental

Table I. Material parameter values used in equation (3) for fitting data of LDPE melts at 160 °C ($\alpha = 14{\cdot}38$, $\beta = 0{\cdot}018$)

k	λ_k(s)	a_k(Pa)
1	$7{\cdot}01 \times 10^{-5}$	$1{\cdot}29 \times 10^{5}$
2	$7{\cdot}01 \times 10^{-4}$	$9{\cdot}48 \times 10^{4}$
3	$7{\cdot}01 \times 10^{-3}$	$5{\cdot}86 \times 10^{4}$
4	$7{\cdot}01 \times 10^{-2}$	$2{\cdot}67 \times 10^{4}$
5	$7{\cdot}01 \times 10^{-1}$	$9{\cdot}80 \times 10^{3}$
6	$7{\cdot}01 \times 10^{0}$	$1{\cdot}89 \times 10^{3}$
7	$7{\cdot}01 \times 10^{1}$	$1{\cdot}80 \times 10^{2}$
8	$7{\cdot}01 \times 10^{2}$	$1{\cdot}00 \times 10^{0}$

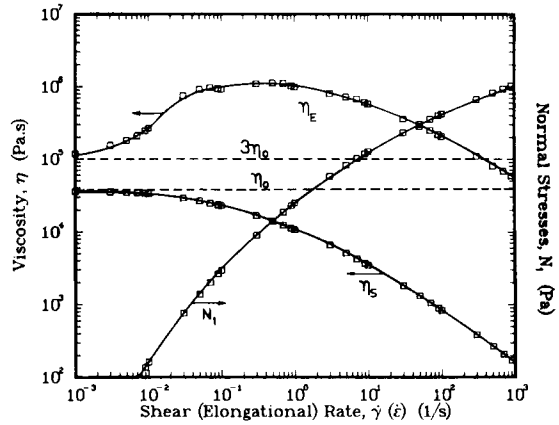

Figure 1. Model predictions of shear viscosity η_S, first normal stress difference N_1 and elongational viscosity η_E for LDPE at 160 °C using equation (3)

data, especially at low shear rates. For the elongational viscosity curve a value of $\beta = 0{\cdot}018$ has been assumed as done in earlier works[1,3,9] to describe a strain-thickening behaviour of LDPE melts.

Referring to Figure 2, let L_0 and L_{res} be the lengths of the downstream and upstream tubes respectively and let R_0 be the radius of the downstream tube. For our calculations we have chosen the ratios L_0/R_0 and L_{res}/R_0 to be 30 and 16 respectively. For the boundary conditions we assume there is no slip along the walls and impose a fully developed velocity profile at the entry and exit sections. The relative strain tensor upstream of the entry section is calculated on the basis of the fully developed velocity profile. The computations were performed originally with mesh M2 shown in Figure 3(a) in its full length. M2 contains 300 elements, 1007 nodes and 2368 degrees of freedom for the velocity components and the pressure. Figure 3(b) shows a partial view of the finite element mesh near the die entry. Meshes coarser and finer than the one in Figure 3 have also been tried and results showed satisfactory mesh independence as discussed later in Section 4. Note that the mesh in Figure 3 is finer than either of the two meshes used by Dupont and Crochet.[3]

An appropriate flow parameter needs to be chosen to describe the solution in this complex case. One of them is the apparent shear rate Γ which is defined as

$$\Gamma = 4\bar{V}_0/R_0, \tag{28}$$

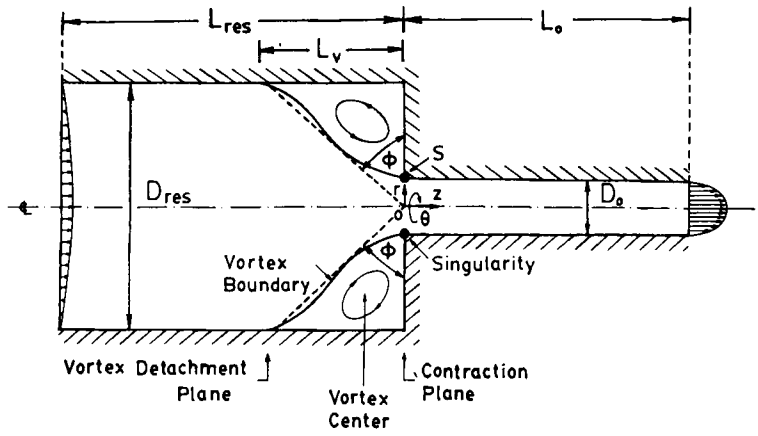

Figure 2. Schematic diagram of the circular abrupt contraction geometry and definition of the vortex opening angle ϕ

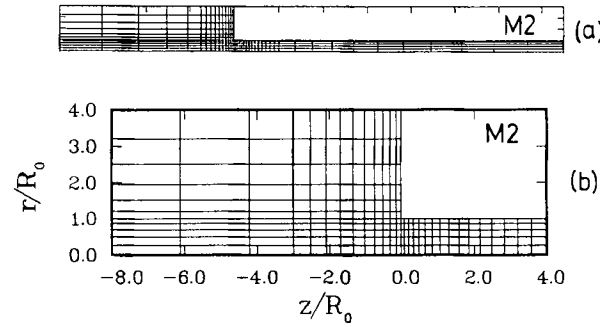

Figure 3. (a) Full finite element mesh containing 300 elements (M2) originally used in the 4:1 circular abrupt contraction computation. (b) Partial view of the finite element mesh near the contraction entrance (the mesh extends upstream to $-16\,R_0$ and downstream to $+30\,R_0$)

where $\bar{V}_0$ is the average axial velocity in the downstream tube. With a fixed flow rate Q or apparent shear rate Γ one can calculate the corresponding value of the fully developed shear rate $\dot{\gamma}_w$ on the downstream wall and hence the dimensionless recoverable shear (or stress ratio) S_R, which is the ratio of the first normal stress difference N_1 to twice the shear stress τ_w, i.e.

$$S_R = N_1/2\tau_w. \tag{29}$$

We first used the *flow rate increment scheme* to obtain solutions from low to high flow rates. The calculation started as usual from a low-flow-rate solution and we increased the apparent shear rate value Γ to reach higher elasticity levels. As reported by Dupont and Crochet,[3] their calculations converged up to a value of $\Gamma = 4\ s^{-1}$, beyond which they failed to obtain a solution. However, our calculations have reached $\Gamma = 70\ s^{-1}$ ($\dot{\gamma}_w = 104\ s^{-1}$, $S_R = 2{\cdot}5$) with good convergence, and we stopped increasing Γ because of lack of interest. It took about 17 min CPU time on a VAX-11/780 to complete one iteration with the mesh shown in Figure 3.

It is interesting to compare our results with those of Dupont and Crochet.[3] Confirming their results, we have also found that the flow solution is characterized by an early growth of the corner vortex. Figure 4 shows the streamline/vortex patterns at various Γ-values. The first two patterns

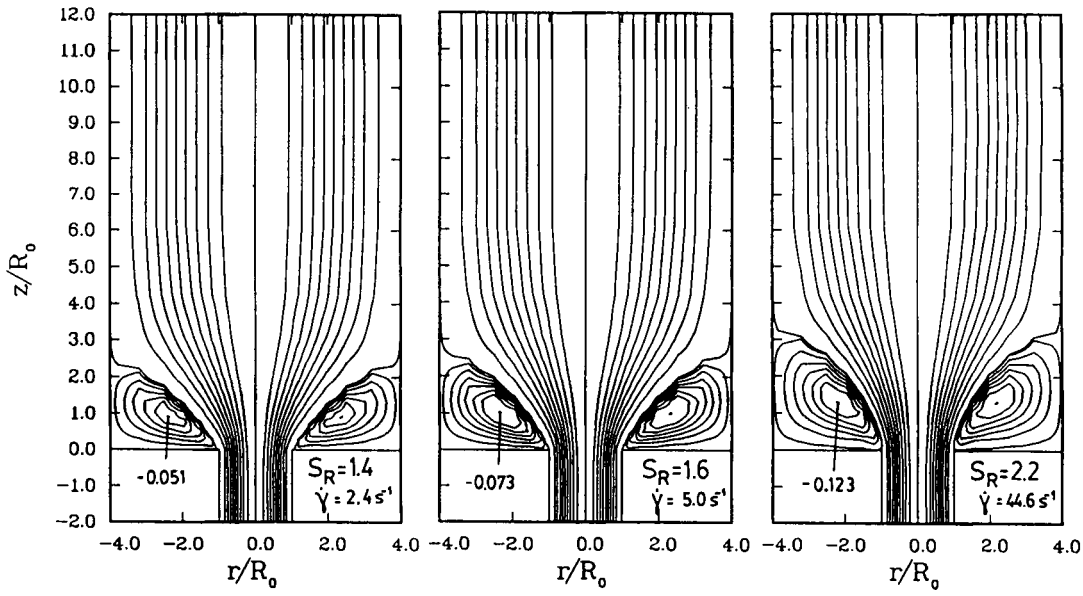

Figure 4. Streamline/vortex patterns at Γ-values of 2, 4 and 30 s^{-1} for the flow of an LDPE melt in a 4:1 circular contraction (data given in Table I). The patterns for $\Gamma = 2$ and 4 s^{-1} correspond to results by Dupont and Crochet[3]

for $\Gamma = 2$ and 4 s^{-1} correspond to the results by Dupont and Crochet.[3] The virtual identity of the flow patterns between the two works is remarkable. However, our work extends to higher Γ-values and we also show in Figure 4 the pattern for $\Gamma = 30\ s^{-1}$. We see that the size of the entry vortex continues to increase with shear rate, but the increase slows down at higher shear rates where the increase of S_R with Γ also slows down.

In order to quantify the vortex intensity, we can compute the maximum difference between the streamfunction value in the vortex region and the value on the wall. The relative vortex intensity is then calculated as the ratio of this maximum difference to the flow rate in the main stream according to

$$-\psi^*_{v,max} = \frac{\psi_{v,max} - \psi_w}{\psi_{cl} - \psi_w}, \tag{30}$$

where ψ_w and ψ_{cl} are the streamfunction values at the wall and centreline respectively. In Figure 5 we have plotted the relative vortex intensity $-\psi^*_{v,max}$ as a function of S_R, in comparison with results by Dupont and Crochet.[3] The agreement between the two is good.

The vortex size can be quantified by the opening angle ϕ defined schematically in Figure 2. We plot the opening angle ϕ as a function of S_R in Figure 6. The agreement between our prediction and that of Reference 3 is very good. Another important quantity is the entrance correction n_{en} corresponding to excess pressure losses ΔP_{en} over and above the fully developed flow value owing to contraction. The definition of n_{en} is given by

$$n_{en} = \frac{\Delta P - \Delta P_{res} - \Delta P_0}{2\tau_w}, \tag{31}$$

where ΔP is the total pressure loss in the system and ΔP_{res} and ΔP_0 are the pressure losses based on the fully developed flow in the upstream and downstream tubes respectively. Figure 7 shows

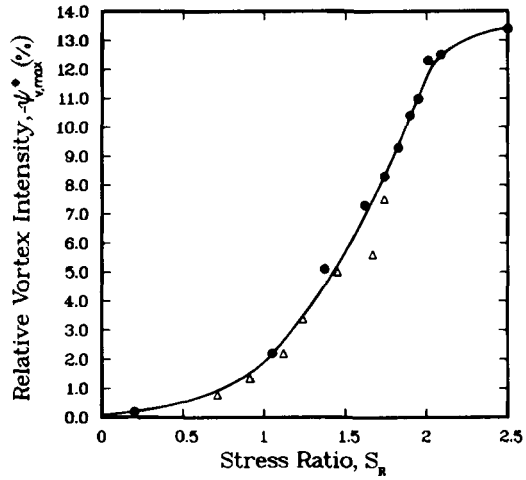

Figure 5. Relative vortex intensity $-\psi^*_{v,max}$ as a function of S_R in a 4:1 circular contraction (LDPE): ——, this work; △, Dupont and Crochet[3]

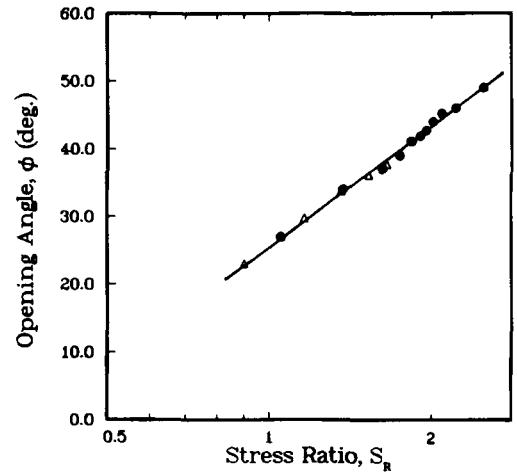

Figure 6. Opening angle ϕ as a function of S_R in a 4:1 circular contraction (LDPE): ——, this work; △, Dupont and Crochet[3]

our prediction for n_{en} as a function of S_R for the 4:1 contraction flow together with data from Reference 3. We find that the entrance correction exhibits a large increase as S_R increases, in agreement with data from Reference 3 and the general trend found in experimental studies.[12,13]

4. MESH REFINEMENT AND CONVERGENCE STUDIES

The most serious problem in the numerical simulation of non-Newtonian flow used to be the *high-Weissenberg-number* problem, i.e. numerical computations became unstable at relatively high values of the Weissenberg number, which is a measure of the elasticity strength in the flow and can be shown to be equivalent to S_R used in this work. However, a few recent works,[1,3] including the present one, have convincingly shown that a computation of a complex viscoelastic

flow does not necessarily fail as a result of reaching high elasticity levels if a realistic constitutive model as well as a stable algorithm are used. This is not to say the problem has been completely solved, since faster, more accurate and more general algorithms for non-Newtonian flow simulation will always be in demand. In particular, when there is no instability in the computation, a few problems concerning numerical aspects will still be present: mesh dependence of results, convergence speed and relative errors. This section is devoted to these important questions. We intend to evaluate realistically the performance of the present algorithm through a mesh refinement and convergence study.

So far the results for the 4:1 contraction flow were obtained by using the 300-element mesh shown in Figure 3. To find out the extent of mesh dependence we used two more meshes to carry out the same 4:1 contraction calculations: a coarser one with 200 elements (M1) and a finer one with 400 elements (M3). Table II gives useful information about the three meshes used and the corresponding CPU time per iteration. The results from different meshes will be compared at $\dot{\gamma}_w = 104\ s^{-1}$ ($\Gamma = 70\ s^{-1}$, $S_R = 2{\cdot}5$), which corresponds to the highest flow rate for which we have

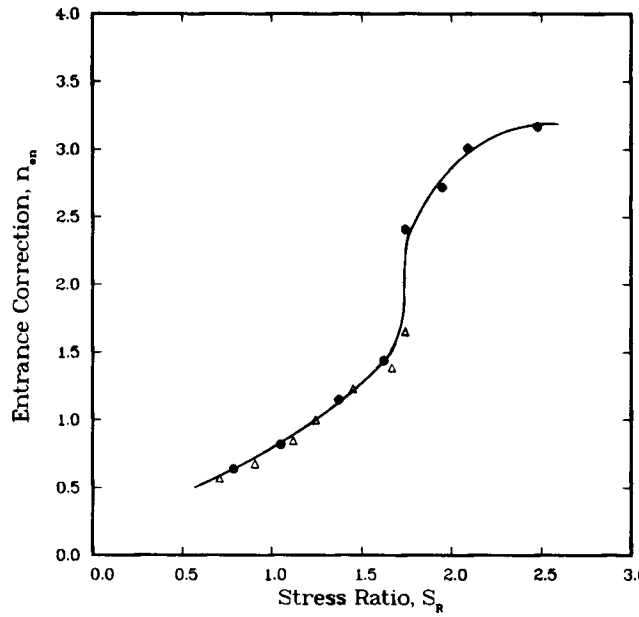

Figure 7. Entrance correction n_{en} as a function of S_R in a 4:1 circular contraction (LDPE): ——, this work; △, Dupont and Crochet[3]

Table II. Mesh characteristics of the three different finite element meshes used in the computations

Mesh	Number of elements	Number of nodes	Number of degrees of freedom	Size of corner element ($r \times z$)	CPU time* (seconds/iteration)
M1	200	689	1623	0·18 × 0·22	320
M2	300	1007	2368	0·04 × 0·09	1000
M3	400	1317	3093	0·01 × 0·07	1650

* On a VAX-11/780 computer.

computed with the 300-element mesh (M2). Furthermore, we have used here the *elasticity increment scheme* for the two new meshes to directly get the results at $\dot{\gamma}_w = 104\ s^{-1}$, while we had used before the *flow rate increment scheme* to get results at different flow (shear) rates for the 300-element mesh. This is an additional test of our numerical method, since different iterative schemes should give the same results for the same problem.

Figure 8 shows partial views of the 200- and 400-element meshes. In refining the mesh we have concentrated more elements near the re-entry corner and close to the walls where the solution is most sensitive. As convergence criterion we have taken here the relative difference in the solution vector of velocity, which gives rise to the Euclidean norm-of-the-error ε defined as

$$\varepsilon = \frac{\| \Delta \mathbf{v}^{(i)} \|}{\| \mathbf{v}^{(i)} \|}, \tag{32}$$

where i is the current iteration, $\Delta \mathbf{v}^{(i)} = \mathbf{v}^{(*)} - \mathbf{v}^{(i)}$ (referring also to equation (26)) and $\| \cdot \|$ is the magnitude of a vector. In terms of number of iterations N and the corresponding norm-of-the-error ε, not much difference was found for the three meshes. Figure 9 plots the relative error as a function of the number of iterations. Note here that the number of iterations was counted after the full elasticity level was reached, i.e. after the elasticity increment ω of equation (24) reached one for meshes M1 and M3, and after the full flow rate was reached for mesh M2. This figure clearly shows that the convergence speed and accuracy are largely mesh-independent. The relatively steep decrease in the value of ε, occurring once in each of the three curves in Figure 9, is due to the change of the relaxation factor Ω in equation (27) from zero to 0·5, after which the iterations were continued with $\Omega = 0{\cdot}5$.

The iteration was very stable even after as many as 70 iterations, and the relative error decreased monotonically with the number of iterations to the order of 10^{-4}, which we believe is very satisfactory for a complicated integral model and a complex viscoelastic flow at high flow rate. On the other hand, the convergence was relatively slow compared with calculations for inelastic cases.[14] Unfortunately, no comparison can be made with any other work which uses the same complicated viscoelastic integral model for the same problem and gets convergence at high flow rates, since the results shown here seem to be the only ones available in the literature for the time being.

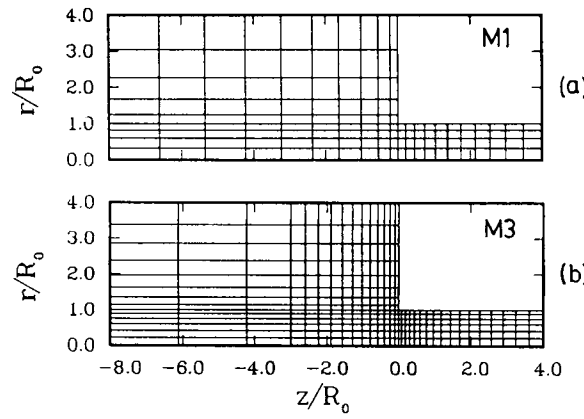

Figure 8. Partial views of different finite element meshes used for convergence studies: (a) 200 elements; (b) 400 elements (see also Table II for details)

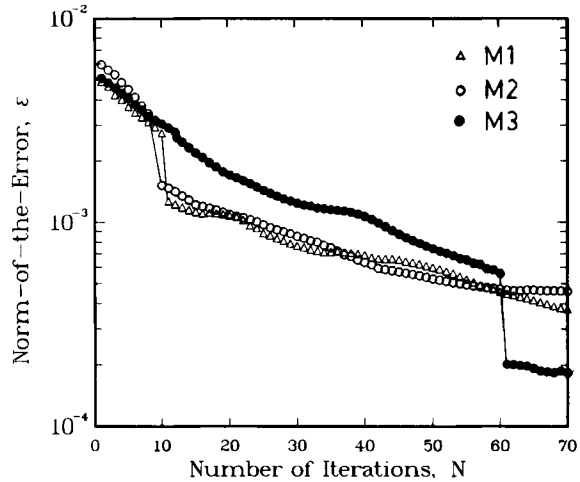

Figure 9. Norm-of-the-error as a function of number of iterations for the three meshes

In an attempt to speed up the convergence, we have considered some other alternatives. It is well known that a Newton–Raphson iterative algorithm converges faster than Picard iteration, but it seems extremely difficult and time-consuming, if not impossible, to perform Newton–Raphson iterations when integral-type constitutive equations are used. In the case of integral models the residual vector **R** can be written as

$$\mathbf{R}(\mathbf{v}^{(i+1)}) = \mathbf{A}\mathbf{v}^{(i+1)} - \mathbf{B}(\mathbf{v}^{(i)}) = \mathbf{0}, \tag{33}$$

where **A** is the fluid (stiffness) matrix, **B** is the RHS (load) vector and $\mathbf{v}^{(i)}$ is the solution vector at the ith iteration. In the Newton–Raphson method one solves for

$$\mathbf{J}(\mathbf{v}^{(i)})\Delta\mathbf{v}^{(i)} = -\mathbf{R}(\mathbf{v}^{(i)}), \tag{34}$$

where $\mathbf{J}(\mathbf{v}^{(i)})$ is the Jacobian matrix $\partial\mathbf{R}(\mathbf{v}^{(i)})/\partial\mathbf{v}^{(i)}$. The evaluation of the Jacobian would involve the calculation of either the analytical derivative of $\mathbf{B}(\mathbf{v}^{(i)})$ or the numerical finite difference approximation of $\partial\mathbf{B}/\partial\mathbf{v}$, which is not feasible because the stress vector is a complicated functional of the entire upstream velocity field.

The dominant-eigenvalue method[15,16] provides an alternative for speeding up the convergence while keeping the Picard method as the basic iterative scheme. Karagiannis *et al.*[14] performed a convergence study for power-law fluids and found that in very non-linear cases (very small power-law index) the dominant-eigenvalue method could be faster than the Newton–Raphson method. The basic idea is to perform a number of Picard iterations on the equation set, building up an iteration history, and then to accelerate the iterative process by predicting the apparent solution from the previous iterations. The number of dominant eigenvalues needed is problem-dependent, but following Karagiannis *et al.*[14] we have only tried here the one-dominant-eigenvalue approach. According to Crowe and Nishio,[16] for the simple one-dominant-eigenvalue case the apparent solution vector is predicted as

$$\mathbf{v}^{(*)} = \mathbf{v}^{(i)} + \frac{\mathbf{v}^{(i+1)} - \mathbf{v}^{(i)}}{1+\mu_1}, \tag{35}$$

$$\mu_1 = -\frac{\Delta\mathbf{v}^{(i)} \cdot \Delta\mathbf{v}^{(i-1)}}{\Delta\mathbf{v}^{(i-1)} \cdot \Delta\mathbf{v}^{(i-1)}}. \tag{36}$$

Promoting too frequently leads to numerical instability. In our numerical experiments we promoted once or twice for every three or five Picard iterations. Unfortunately, we found that the performance of the iterations was not consistent, i.e. it sometimes speeded up the convergence, sometimes not, compared with the simple Picard iteration. Therefore the overall effect of the dominant-eigenvalue method here was not significant. Many more numerical experiments are needed to determine whether the simple one-dominant-eigenvalue approach works in the case of integral viscoelastic models. It is an open question whether the dominant eigenvalue reflects the essential non-linear viscoelastic nature of the equation set as it does in the case of inelastic models.

We have compared the performances of the three meshes in terms of convergence speed measured by number of iterations and in terms of accuracy measured by relative difference in velocity. We now proceed to compare the actual solutions among the three different meshes. The maximum differences among results of different meshes occur near the singularity in the domain. Figure 10 plots predictions by all three meshes of the non-Newtonian stress components τ_{zz} and

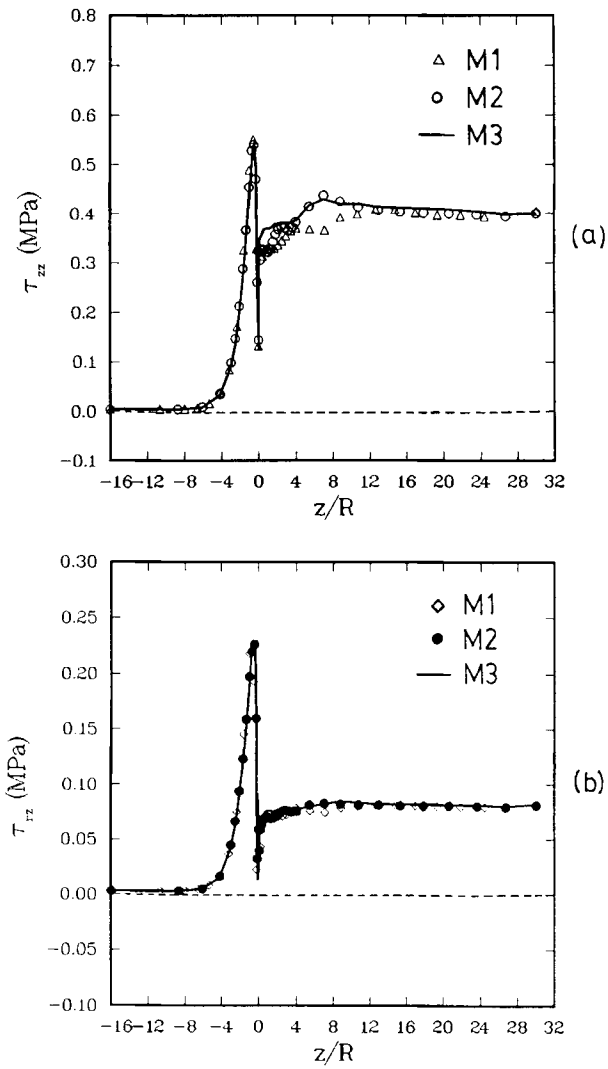

Figure 10. Predictions by all three meshes of the non-Newtonian stress components (a) τ_{zz} and (b) τ_{rz} along the $r = 1{\cdot}0$ line

τ_{rz} along the $r=1{\cdot}0$ line, which passes through the re-entry corner and coincides with the downstream wall. Similar plots for Maxwell stress components have been shown in earlier works by Marchal and Crochet[17] and by Luo and Tanner.[11] Figure 10 shows that the predictions by the different meshes indeed agree very well, especially those computed with the 300- and 400-element meshes. It is interesting to note that most differences occur *after* the singularity, which is not very hard to explain if we remember we are dealing with fluids with memory, i.e. what happened upstream of the singularity will affect the flow downstream, and the effect will fade away as fluid particles flow further downstream. Comparing the results of Figure 10 with Maxwell stress plots in the same geometry[11,17], a significant difference arises: the K-BKZ model predicts a stress overshoot for both τ_{zz} and τ_{rz} after the singularity, while the Maxwell model predicted none. Finally, Figure 11 shows the vortex patterns obtained from the three meshes. They are all almost identical with each other. Quantitatively, Table III shows important global quantities such as vortex opening angle, relative vortex intensity and total pressure drop obtained from the three meshes. Again the agreement is very good. This is particularly encouraging since it shows that a mesh with 200 elements such as M1 is adequate for obtaining reasonably accurate solutions without spending too much CPU time.

All the comparisons made above have shown good accuracy and very satisfactory mesh independence of the calculations, and therefore prove the reliability of our new numerical method using integral-type constitutive equations for solving complex viscoelastic flow problems encountered in real industrial applications.

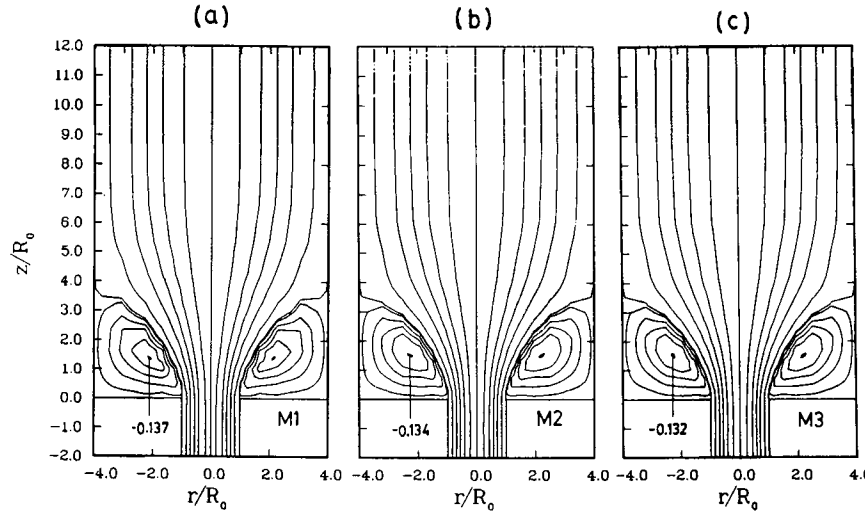

Figure 11. Vortex patterns obtained from the three meshes at $\dot{\gamma}_w=104\,s^{-1}$ ($\Gamma=70\,s^{-1}$, $S_R=2{\cdot}5$): (a) 200 elements; (b) 300 elements; (c) 400 elements

Table III. Global quantities obtained from the three meshes at $\dot{\gamma}_w=104\,s^{-1}$ ($\Gamma=70\,s^{-1}$, $S_R=2{\cdot}5$)

Mesh	Number of elements	Opening angle ϕ (deg)	Relative vortex intensity $-\psi^*_{v,max}$ (%)	Total pressure drop ΔP (MPa)
M1	200	49	13·7	5·61
M2	300	49	13·4	5·62
M3	400	48	13·2	5·64

5. CONCLUDING REMARKS

In this paper a new finite element technique has been developed for using integral-type constitutive equations in viscoelastic flow simulations that involve recirculating regions. The new method has been tested in complex abrupt contraction flows of viscoelastic fluids where reliable numerical results are available in the literature. The performance of the new method is further evaluated by a mesh refinement and convergence study. It is found that the present method is relatively inexpensive and in the meantime can reach higher elasticity levels without numerical instability, compared with the best available similar calculations in the literature. The accuracy of the numerical results is satisfactory both in terms of relative errors between iterations and in terms of mesh independence.

The present results show that it is now possible to study complex flows with or without recirculating regions using realistic integral-type constitutive equations. The range of simulations is not restricted to low shear rates but pretty well covers the practical range of operations, and the cost is moderate as well. Our current objective is to extend this new scheme to include free surfaces and interfaces which occur in multilayer co-extrusion operations in the polymer-processing industry. A more fundamental research task is to develop a faster iterative scheme for integral-type viscoelastic constitutive equations, taking into consideration the unique mathematical characteristics of the non-linear viscoelastic system.

ACKNOWLEDGEMENT

Financial assistance from the Natural Sciences and Engineering Research Council of Canada (NSERC) is gratefully acknowledged.

REFERENCES

1. X.-L. Luo and R. I. Tanner, *J. Non-Newt. Fluid Mech.*, **22,** 61 (1986).
2. X.-L. Luo and R. I. Tanner, *Int. J. Numer. Methods Eng.*, **15,** 9 (1988).
3. S. Dupont and M. J. Crochet, *J. Non-Newt. Fluid Mech.*, **29,** 81 (1988).
4. X.-L. Luo and E. Mitsoulis, *J. Rheol.*, **33,** 1307 (1989).
5. M. J. Crochet and R. Keunings, *Proc. 2nd World Congr. on Chemical Engineering, Vol. 6* C.S.Ch.E., Montreal, 1981, p. 285.
6. E. Mitsoulis and F. L. Heng, *Rheol. Acta*, **26,** 414 (1987).
7. S. Dupont and M. J. Crochet, *J. Non-Newt. Fluid Mech.*, **17,** 157 (1985).
8. J. L. White and A. Kondo, *J. Non-Newt. Fluid Mech.*, **3,** 41 (1978).
9. A. C. Papanastasiou, L. E. Scriven and C. W. Macosko, *J. Rheol.*, **27,** 387 (1983).
10. B. Caswell, *Arch. Rat. Mech. Anal.*, **26,** 385 (1967).
11. X.-L. Luo and R. I. Tanner, *J. Non-Newt. Fluid Mech.*, **31,** 143 (1989).
12. T. F. Ballenger and J. L. White, *J. Appl. Polym. Sci.*, **15,** 1949 (1971).
13. J. L. White, *Appl. Polym. Symp.*, **20,** 155 (1973).
14. A. Karagiannis, H. Mavridis, A. N. Hrymak and J. Vlachopoulos, *Int. J. Numer. Methods Fluids*, **8,** 123 (1988).
15. O. Orbach and C. M. Crowe, *Can. J. Chem. Eng.*, **49,** 509 (1971).
16. C. M. Crowe and M. Nishio, *AIChE J.*, **21,** 528 (1975).
17. J. M. Marchal and M. J. Crochet, *J. Non-Newt. Fluid Mech.*, **26,** 77 (1987).