Available online at www.sciencedirect.com

SCIENCE DIRECT®

Journal of Computational Physics 216 (2006) 92–113

JOURNAL OF COMPUTATIONAL PHYSICS

www.elsevier.com/locate/jcp

Development and evaluation of a micro–macro algorithm for the simulation of polymer flow

Kathleen Feigl *, Franz X. Tanner

Michigan Technological University, Department of Mathematical Sciences, 1400 Townsend Drive, Houghton, MI 49931-1295, USA

Received 22 April 2005; received in revised form 28 October 2005; accepted 18 November 2005
Available online 4 January 2006

Abstract

A micro–macro algorithm for the calculation of polymer flow is developed and numerically evaluated. The system being solved consists of the momentum and mass conservation equations from continuum mechanics coupled with a microscopic-based rheological model for polymer stress. Standard finite element techniques are used to solve the conservation equations for velocity and pressure, while stochastic simulation techniques are used to compute polymer stress from the simulated polymer dynamics in the rheological model. The rheological model considered combines aspects of reptation, network and continuum models. Two types of spatial approximation are considered for the configuration fields defining the dynamics in the model: piecewise constant and piecewise linear. The micro–macro algorithm is evaluated by simulating the abrupt planar die entry flow of a polyisobutylene solution described in the literature. The computed velocity and stress fields are found to be essentially independent of mesh size and ensemble size, while there is some dependence of the results on the order of spatial approximation to the configuration fields close to the die entry. Comparison with experimental data shows that the piecewise linear approximation leads to better predictions of the centerline first normal stress difference. Finally, the computational time associated with the piecewise constant spatial approximation is found to be about 2.5 times lower than that associated with the piecewise linear approximation. This is the result of the more efficient time integration scheme that is possible with the former type of approximation due to the pointwise incompressibility guaranteed by the choice of velocity–pressure finite element.

Keywords: Micro–macro simulations; Viscoelastic flow; Polymeric fluids; Planar die entry domain

1. Introduction

The accurate simulation of polymer flow in complex flow domains relies crucially on the ability of the rheological model to describe accurately the stress–strain relationship in the fluid. Rheological models for polymeric fluids can be formulated at different levels of description, e.g., the atomistic level, the mesoscopic level, and the continuum (or macroscopic) level. Atomistic modeling, which contains the most detailed description of the polymer, is currently not suited for macroscopic flow calculations due to prohibitively large computational

* Corresponding author. Tel.: +1 906 487 2221; fax: +1 906 487 3133.
E-mail address: feigl@mtu.edu (K. Feigl).

0021-9991/$ - see front matter © 2005 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2005.11.026

requirements. On a larger scale, the more coarse-grained mesoscopic models attempt to describe the configurations of the macromolecules and their evolution in flow. The polymer stress is then expressed as a function of these dynamics and involves appropriate probability distribution functions. Examples are models from polymer kinetic theory, such as bead-spring models, network models and reptation models [7]. On the macroscopic level, continuum models appear in closed form, usually as an integral or a set of differential equations relating the polymer stress to the strain history. Although many continuum models are derived from, or based on, mesoscopic models, simplifying assumptions and closure approximations are needed to go from the mesoscale to the macroscale, and this can affect the models' predictions.

Computationally, continuum models are the best suited for implementation into a flow simulation algorithm, and this is the usual approach in computational viscoelastic fluid dynamics. Unfortunately, polymeric fluids are rheologically too complex to be universally described by continuum-level models, particularly in complex flow domains. This has been shown repeatedly in the literature by researchers who compared experimental data with flow simulations using these models. Moreover, these models offer no information about the configuration of polymer molecules, and therefore cannot be used to study the relationship between the flow process, the flow-induced microstructure, and the fluid rheology.

Mesoscopic models allow the inclusion of more physically-based information on the polymer dynamics and are therefore more appropriate for model development than continuum models. However, traditionally they had not been suitable for implementation into a flow solver. In the late 1980s and early 1990s, the first micro–macro methods for polymer flows were introduced [5,8,24,30,31]. Micro–macro methods couple a coarse-grained molecular model (i.e., mesoscale model) with the macroscopic conservation equations from continuum mechanics. The resulting multiscale, or micro–macro, algorithm often relies on stochastic simulation techniques to compute polymer stress from the molecular-based model and standard CFD methods, such as finite elements, to compute velocity, pressure (and perhaps temperature) from the conservation equations. Since their introduction, micro–macro methods have continued to develop steadily, e.g., [1,9,14,15,17–19,21,23,28,34], although their use and applicability lag behind the more established and developed conventional macroscopic methods which rely on continuum models. The current state of micro–macro simulations and additional references can be found in the recent review article of Keunings [22].

The purpose of this paper is to evaluate and modify a micro–macro algorithm which was recently introduced by Feigl and Senaratne [9]. The rheological model considered is multiscale, combining aspects of network and reptation theories for concentrated polymer solutions and melts with continuum models [11,12]. The original algorithm uses a piecewise constant spatial approximation to the configuration fields defining the dynamics of the rheological model. As discussed in the current paper, this allows an efficient second-order time integration scheme due to the pointwise incompressibility resulting from our particular velocity–pressure finite element. In the modified algorithm presented here, a higher order spatial approximation to the configuration fields is used along with a second-order time integration scheme. The performance of the algorithm with both types of approximation is evaluated numerically in an abrupt planar die entry domain for a concentrated polyisobutylene (PIB) solution. Based on this evaluation, it is found that a combination of the two approaches would lead to a more optimal algorithm. The simulation results are also compared to the experimental data of Quinzani et al. [32] who measured the centerline velocity and normal stress difference in slit-die entry experiments with this fluid.

The remaining of the paper is organized as follows. In Section 2, a brief description is given of the rheological model and concentrated PIB solution considered in this paper. The solution algorithm for the model in homogeneous flow is also given along with some predictions of the fluid's material functions. The micro–macro algorithm for solving the resulting coupled, multiscale model for polymeric flow is detailed in Section 3. The performance of the algorithm is evaluated in Section 4 where simulation results are presented for the abrupt planar die entry domain, and comparisons are made to the available experimental data in a slit-die entry domain. A summary and conclusions are given in Section 5.

2. Rheological model

The incompressible flow of a polymeric fluid under isothermal conditions is governed by the mass and momentum balance equations

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

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

where $\boldsymbol{u}$ is the velocity, ρ is the constant density, p is the pressure, and $\boldsymbol{\tau}$ is the extra-stress tensor which may be written as $\boldsymbol{\tau} = \boldsymbol{\tau}_\mathrm{p} + \boldsymbol{\tau}_\mathrm{s}$, where $\boldsymbol{\tau}_\mathrm{p}$ represents the contribution from the polymer and $\boldsymbol{\tau}_\mathrm{s}$ represents the contribution from a Newtonian solvent (if present). The Newtonian contribution $\boldsymbol{\tau}_\mathrm{s}$ is given by $\boldsymbol{\tau}_\mathrm{s} = \eta_\mathrm{s}[\boldsymbol{\kappa} + \boldsymbol{\kappa}^\dagger]$, where $\boldsymbol{\kappa}$ is the transpose of the velocity gradient tensor, η_s is the solvent viscosity, and † represents the transpose. The rheological model for $\boldsymbol{\tau}_\mathrm{p}$ is taken to be a multiscale model introduced by Feigl and Öttinger [11,12], which is briefly described below.

2.1. Model description

The dynamics of the macromolecules are modeled by (i) two stochastic processes which describe the configuration vectors of the macromolecules, (ii) a probability density function which describes the lifetime of each vector, and (iii) evolution equations which describe the motion of the configuration vectors in the flow field during their lifetimes. More specifically, the dimensionless configuration vectors, $\boldsymbol{Q}_1(t)$ and $\boldsymbol{Q}_2(t)$, are Gaussian stochastic processes whose interpretation is based in network theory and reptation theory, respectively, of polymer melts and concentrated polymer solutions. The former represents the end-to-end configuration vector describing the temporary macromolecular entanglements in network theory [7], and the latter represents anisotropic tube cross-section from reptation theory [27]. These vectors are repeatedly created, allowed to evolve in the flow field, destroyed, and then regenerated. During their lifetimes, s_1 and s_2, they evolve according to the equations of motion

$$\frac{\mathrm{d}\boldsymbol{Q}_1(t)}{\mathrm{d}t} = \boldsymbol{\kappa} \cdot \boldsymbol{Q}_1(t), \quad t' < t < t' + s_1, \tag{3}$$

$$\frac{\mathrm{d}\boldsymbol{Q}_2(t)}{\mathrm{d}t} = -\boldsymbol{\kappa}^\dagger \cdot \boldsymbol{Q}_2(t), \quad t' < t < t' + s_2, \tag{4}$$

where t' is the creation time of the vector. Initially (at time $t = t' = 0$) and at subsequent creation times ($t = t'$) each vector has mean $\langle \boldsymbol{Q}_i(t') \rangle = \boldsymbol{0}$ and covariance $\langle \boldsymbol{Q}_i(t')\boldsymbol{Q}_i(t') \rangle = \boldsymbol{\delta}$, $i = 1,2$, where $\boldsymbol{\delta}$ is the 3×3 unit tensor. The lifetime, s, of each configuration vector is determined by a probability density function, $\phi(s)$, which is related to the fluid's linear viscoelastic memory function, $m(t - t')$, via

$$\int_{t-t'}^{\infty} \phi(s)\,\mathrm{d}s = \frac{m(t-t')}{m(0)}.$$

Computationally, a more convenient expression for s, obtained by a coordinate transformation, is given by

$$z(s) = 1 - \frac{m(s)}{m(0)}, \tag{5}$$

where $s = t - t'$ and $z \in [0, 1)$ is a uniformly distributed random number. Typically, the Maxwell linear viscoelastic memory function is used, which has the form

$$m(t - t') = \sum_{k=1}^{N} \frac{\eta_k}{\lambda_k^2} \mathrm{e}^{-(t-t')/\lambda_k}, \tag{6}$$

where $\{\lambda_k, \eta_k\}$ is a set of relaxation times and partial viscosities of the fluid.

The polymer stress tensor $\boldsymbol{\tau}_\mathrm{p}$ is then expressed as the expectation of a function of these dynamics as follows

$$\begin{aligned} \boldsymbol{\tau}_\mathrm{p}(t) &= G(0)\hat{\boldsymbol{\tau}}_\mathrm{p}(t), \\ \hat{\boldsymbol{\tau}}_\mathrm{p}(t) &= \langle f_1(Q_1^2, Q_2^2)\boldsymbol{Q}_1(t)\boldsymbol{Q}_1(t) + f_2(Q_1^2, Q_2^2)\boldsymbol{Q}_2(t)\boldsymbol{Q}_2(t) \rangle, \end{aligned} \tag{7}$$

where $\hat{\boldsymbol{\tau}}_\mathrm{p}$ is the dimensionless form of $\boldsymbol{\tau}_\mathrm{p}$, the factor $G(0)$ is the value of the relaxation modulus, $G(t - t')$, at $t = t'$, and f_1 and f_2 are scalar functions of $Q_i^2 = \mathrm{trace}(\boldsymbol{Q}_i\boldsymbol{Q}_i)$, $i = 1, 2$. The exact form and parameters in the strain functions can be chosen to fit the material functions of a given fluid. The only condition on these functions is that they must satisfy a constraint to ensure that the correct linear viscoelastic behavior is predicted. This constraint is

$$\frac{\eta_\mathrm{p}}{\eta_\mathrm{p}+\eta_\mathrm{s}} = \frac{1}{3}\langle Q_1^{\mathrm{eq}2} f_1^{\mathrm{eq}} - Q_2^{\mathrm{eq}2} f_2^{\mathrm{eq}}\rangle_0 + \frac{2}{15}\langle (Q_1^{\mathrm{eq}2})^2 f_{1,1}^{\mathrm{eq}} - (Q_2^{\mathrm{eq}2})^2 f_{2,2}^{\mathrm{eq}}\rangle_0, \tag{8}$$

where η_p is the polymer contribution to the zero-shear-rate viscosity $\eta_0 = \eta_\mathrm{p} + \eta_\mathrm{s}$, $Q_i^{\mathrm{eq}2} \equiv (Q_i^{\mathrm{eq}})^2$ denotes the value of Q_i^2 at equilibrium, $f_i^{\mathrm{eq}} = f_i(Q_1^{\mathrm{eq}2}, Q_2^{\mathrm{eq}2})$ and $f_{i,j}^{\mathrm{eq}} = \partial f_i(Q_1^{\mathrm{eq}2}, Q_2^{\mathrm{eq}2})/\partial Q_j^2$ for $i,j = 1,2$. The notation $\langle\cdot\rangle_0$ indicates taking a Gaussian average in six-dimensional space with respect to the Gaussian probability density in six-dimensional space with mean $\mathbf{0}$ and square of the width $\boldsymbol{\delta}$. If these integrals are written in spherical coordinates, then they can be reduced to integrals in two-dimensional space, since the angular integrations can be performed analytically. Further details of the model, along with its ability to predict well the material behavior of polymer melts, can be found in Refs. [11,12].

2.2. Material description and simulation algorithm in homogeneous flow

The fluid simulated was a concentrated polymer solution, specifically, a 5% polyisobutylene (PIB) solution in tetradecane at 25C described by Quinzani et al. [32,33]. This fluid was chosen since it has been characterized in various viscometric flows [33] and has been used in abrupt slit-die entry experiments in which centerline velocity and stress measurements were taken with laser doppler velocimetry (LDV) and flow-induced birefringence (FIB), respectively [32]. It therefore allows some comparison between micro–macro simulation and experiment in terms of both velocity and stress.

A rheological model of the form in Eq. (7) was previously derived for this PIB solution by Feigl and Senaratne [9]. It uses the Maxwell memory function, Eq. (6), with a relaxation spectrum determined by Quinzani et al. [33]. The spectrum is restated in Table 1. The strain functions are

$$f_1(Q_1^2, Q_2^2) = \frac{1}{a_0 + a_1 Q_1^2 + a_2 Q_2^2}, \quad f_2 \equiv 0,$$

where $a_0 = 0.635$, $a_1 = 0.040$ and $a_2 = 0.038$ were chosen to satisfy the small deformation constraint, Eq. (8), and to provide good predictions to the material functions of the PIB solution.

The simulation algorithm used to compute the material functions in homogeneous flow can be briefly described as follows. Let $\{\boldsymbol{Q}_1^k(t)\}_{k=1}^{N_\mathrm{p}}$ and $\{\boldsymbol{Q}_2^k(t)\}_{k=1}^{N_\mathrm{p}}$ represent two independent ensembles for the random variables $\boldsymbol{Q}_1(t)$ and $\boldsymbol{Q}_2(t)$, respectively, where N_p represents the size of the ensembles, and let $t = T$ denote the time at which polymer stress is to be computed. For $k = 1, \ldots, N_\mathrm{p}$, repeat the following three steps until time $t = T$ is reached:

1. Randomly generate a lifetime s_i^k for $\boldsymbol{Q}_i^k$ from Eq. (5). Since the Maxwell memory function in Eq. (6) is used, then for each mode the expression for $z(s)$ can be solved analytically for s. In this case, simulations can be performed separately for each relaxation mode and the results added together [12]. Alternatively, one lifetime can be generated from the entire spectrum by using Newton's method to solve this nonlinear equation for s given a specified value of z. A good initial guess can be found from a piecewise function approximation to $s(z)$. For the fluid under consideration, these two methods produced virtually identical results. In the micro–macro simulations in this study, the latter approach is adopted.

Table 1
Relaxation spectrum for PIB solution at 25C (zero-shear-rate viscosity $\eta_0 = 1.4258$ Pa s; solvent contribution $\eta_\mathrm{s} = 0.002$ Pa s; polymer contribution $\eta_\mathrm{p} = 1.4238$ Pa s)

k	λ_k (s)	η_k (Pa s)
1	0.0059	0.5850
2	0.0389	0.5664
3	0.1396	0.2324
4	0.6855	0.0400

2. Randomly generate $\boldsymbol{Q}_i^k$ according to a Gaussian distribution with mean $\mathbf{0}$ and covariance $\boldsymbol{\delta}$, for $i = 1,2$.
3. Evolve $\boldsymbol{Q}_i^k$ according to the corresponding equation of motion for the duration of the lifetime s_i^k, for $i = 1,2$. For homogeneous flow, a first-order explicit Euler method was used to integrate Eqs. (3) and (4). This method is exact in simple shear flow.

When time $t = T$ is reached, the values of $\boldsymbol{Q}_1^k$ and $\boldsymbol{Q}_2^k$ are substituted into Eq. (7) for each k and averaged over all k to obtain the dimensionless $\hat{\boldsymbol{\tau}}_{\mathrm{p}}$. A variance reduction technique based on equilibrium control variates was employed to reduce the statistical errors at low shear (or deformation) rates. For the transient calculations, an ensemble of $N_{\mathrm{p}} = 10^7$ trajectories was used for each configuration vector. Material functions in steady flows were computed using the principle of ergodicity [29] in which a long single trajectory is tracked over a large simulation time. In each case, vectorization was used to reduce the computational time.

Comparisons of the model's predictions in viscometric flows with experimental data showed that the model describes well the material functions of the fluid [9]. An example of this comparison is shown in Fig. 1(a), where the predicted and measured shear viscosity, η, and first normal stress coefficient, $\Psi_1 = (\tau_{11} - \tau_{22})/\dot{\gamma}^2$, in steady simple shear flow are shown as functions of shear rate $\dot{\gamma}$. A nonzero normal stress ratio, $-\Psi_2/\Psi_1$, was also predicted in steady shear flow, where $\Psi_2 = (\tau_{22} - \tau_{33})/\dot{\gamma}^2$, is the second normal stress coefficient. Predicted values ranged from approximately 0.10 at low shear rates ($\dot{\gamma} = 0.1\ \mathrm{s}^{-1}$) down to 0.025 at higher shear

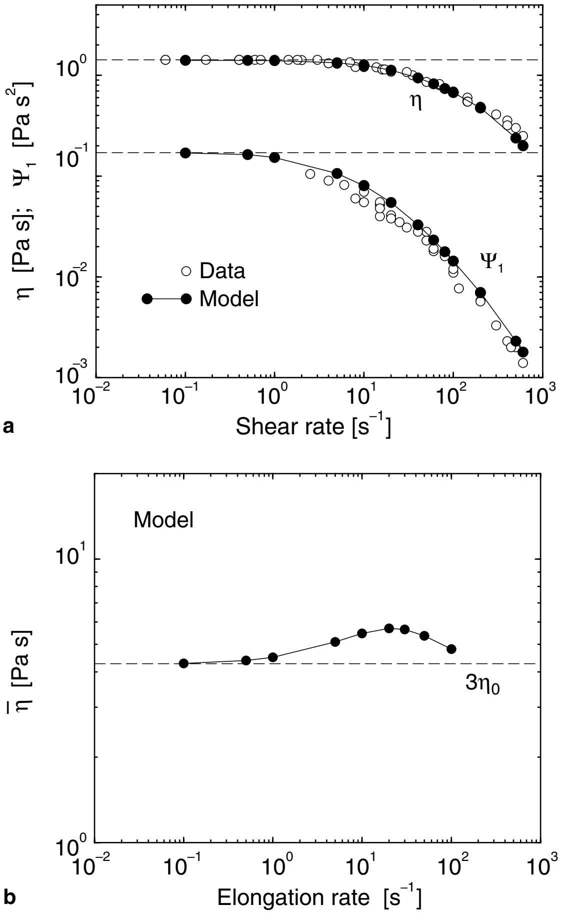

Fig. 1. (a) Comparison of model predictions of steady-state shear viscosity, η, and first normal stress coefficient, Ψ_1, with experimental data of Quinzani et al. [32] for the PIB solution. The dashed horizontal lines indicates the zero-shear-rate viscosity, η_0, and the zero-shear-rate first normal stress coefficient, $\Psi_{1,0}$. (b) Model predictions of steady uniaxial elongational viscosity. The dashed horizontal line indicates the Trouton viscosity, $3\eta_0$.

rates ($\dot{\gamma} = 700\ \text{s}^{-1}$). Due to the difficulty in measuring Ψ_2, there is little experimental data on the normal stress ratio of polymeric fluids. The limited experimental data which does exist indicates that the normal stress ratio typically has a magnitude of about 10%, and values for polymer solutions have been measured in the range of 0.01–0.20 [6,25]. The values predicted by the current model lie within this range.

Fig. 1(b) shows that the model also predicts the correct qualitative behavior of elongational viscosity, $\overline{\eta}$, in steady uniaxial elongational flow, including strain-hardening.

The statistical error bars in these calculations were very small, and are not visible in the figures. Because the errors are so small, the values computed by the above simulation algorithm can be regarded as ''exact'' values of the model predictions in these homogeneous flows. These values can be used for comparison with the micro–macro simulation results presented later. For reference, the simulation algorithm for homogeneous flow described above is referred to as the *standard simulation algorithm*.

3. Micro–macro algorithm

The micro–macro simulation procedure for solving the governing system of equations follows a decoupled, iterative approach. The velocity and pressure fields are computed in one step, in which the divergence of the extra-stress appears as a pseudo-body force from a previously computed stress tensor. In the next step, this velocity field is then used to compute an updated extra-stress tensor $\boldsymbol{\tau} = \boldsymbol{\tau}_\text{p} + \boldsymbol{\tau}_\text{s}$ from the rheological model for the polymer stress contribution, $\boldsymbol{\tau}_\text{p}$, and from the solvent contribution, $\boldsymbol{\tau}_\text{s}$. These two steps are repeated until the relative change in velocity and stress tensor components lie below some preset tolerance. Specifically, the convergence criteria are $\|\boldsymbol{f}_n - \boldsymbol{f}_{n-1}\|/\|\boldsymbol{f}_n\| < \epsilon$, where $\boldsymbol{f}$ represents the nodal values of either velocity or a component of the extra-stress tensor, and the subscripts indicate the iteration. In this study, a tolerance of $\epsilon = 10^{-4}$ was used for velocity and $\epsilon = 10^{-3}$ was used for each component of the extra-stress tensor. For steady flow problems, the initial guesses for the velocity $\boldsymbol{u}$ and pressure p are taken to be the solution to Stokes flow or the solution at a lower flow rate or Deborah number. A description of the individual parts of the simulation procedure is given below.

The macroscopic part of the algorithm consists of computing the velocity and pressure fields by solving the mass and momentum balance equations, Eqs. (1) and (2), using standard finite element techniques. This part of our algorithm is well established and details can be found elsewhere [2–4]. Therefore only a brief description is provided here.

The finite element method is applied to the penalty formulation, in which the incompressibility condition, Eq. (1), is replaced by a penalty equation $\epsilon p + \nabla \cdot \boldsymbol{u} = 0$. The finite element used is the crossed triangle macroelement [2–4,26], consisting of a quadrilateral divided into four triangles by its diagonals. The velocity approximation is piecewise linear in each triangle and continuous over the domain, and the pressure approximation is piecewise constant in each triangle (by the penalty equation). An attractive feature of the discrete velocity approximation of the crossed triangle macroelement is that it is pointwise incompressible (almost everywhere). This feature will be exploited later.

Regardless of whether or not a transient calculation is being performed, the time derivative terms, $\rho\partial\boldsymbol{u}/\partial t$, in the momentum equation are retained, so that the spatial discretization produced by finite element method yields a system of ordinary differential equations. This semi-discrete system is then solved using a time-marching scheme, specifically a one-step, predictor–corrector method described by Hughes et al. [20]. A proper choice of integration parameters in this scheme can accelerate the convergence and improve the efficiency of the overall iterative process. For steady-state problems, such as the one considered in this study, the transient behavior is not required, and experience has shown that an efficient choice of (macroscopic) time step is $\delta t = 1$ with one update of velocity, pressure, and stress being performed in each time step. The previously stated convergence criteria ensure that time effects in the macroscopic flow (e.g., acceleration) have been damped out and that the steady state has been reached.

In any given time step, quantities such as the velocity gradient, the pressure field and the stress fields are, by choice of approximations, discontinuous across element boundaries. Therefore, at any time where smooth fields are desired, such as at the end of a flow simulation once the steady solution has been reached, postprocessing is applied to obtain continuous fields over the domain. This is done by projecting each of these quantities onto the space of continuous, piecewise bilinear polynomials, based on the vertices of the quadrilateral macroelements. To monitor the convergence of these quantities from one iterate to the next, smoothing was performed after each (macroscopic) time step. This procedure generally works well, except possibly at nodes along the boundary of the computational domain. Extrapolation techniques are used to improve the smoothed nodal values on the boundary. Details of the postprocessing can be found in Refs. [4,20].

Bernstein et al. [2] established analytical convergence results for the crossed triangle element with postprocessed pressures in both planar and axisymmetric geometry. Moreover, this element and the above-described numerical methods have been used in standard macroscopic finite element simulations of polymer flows in which the polymer stress was described with an integral viscoelastic constitutive equation. This algorithm has been shown to perform very well against other numerical algorithms and to yield good agreement with experimental data in die entry domains [3,10,13,16]. The above numerical methods have also been used by Feigl et al. [14,15] in micro–macro simulations of polymer flow based on particle tracking techniques.

In the microscopic part of the algorithm, the polymer stress in a given flow field represented by $\boldsymbol{\kappa}$ is calculated from the molecular-based rheological model described in the previous section. Since the flow is not homogeneous in general flow domains, the velocity gradient in Eqs. (3) and (4) depends on the position $\boldsymbol{x}$ as well as t. Therefore, the equations of motion in the rheological model are interpreted to be in the Lagrangian form. There are a few basic strategies for solving these equations. First, the equations can be kept in Lagrangian form and the evolution of the configuration vectors can be computed using the *particle tracking approach*. In this approach, ensembles of configuration vectors, representing the polymer molecules, are tracked in the flow along a set of particle paths, and the stress in each element is computed as an appropriate average over all particle paths passing through the element (e.g., [14,15,18,19]).

Alternatively, the individual configuration vectors, or polymer molecules, $\boldsymbol{Q}_i(t)$, can be replaced by continuous configuration fields, $\boldsymbol{Q}_i(\boldsymbol{x},t)$, which are defined at every point, $\boldsymbol{x}$, in the flow domain, and the governing Lagrangian equations of motion given in Eqs. (3) and (4) can be written in Eulerian form as

$$\frac{\partial \boldsymbol{Q}_1(\boldsymbol{x},t)}{\partial t} + \boldsymbol{u}(\boldsymbol{x},t) \cdot \nabla \boldsymbol{Q}_1(\boldsymbol{x},t) = \boldsymbol{\kappa}(\boldsymbol{x},t) \cdot \boldsymbol{Q}_1(\boldsymbol{x},t), \quad t' < t < t' + s_1, \tag{9}$$

$$\frac{\partial \boldsymbol{Q}_2(\boldsymbol{x},t)}{\partial t} + \boldsymbol{u}(\boldsymbol{x},t) \cdot \nabla \boldsymbol{Q}_2(\boldsymbol{x},t) = -\boldsymbol{\kappa}^{\dagger}(\boldsymbol{x},t) \cdot \boldsymbol{Q}_2(\boldsymbol{x},t), \quad t' < t < t' + s_2. \tag{10}$$

This *configuration field approach* was first introduced by van den Brule and coworkers [21,28,34]. It has several advantages over the particle tracking approach, including allowing smaller ensemble sizes (of the order of 10^3), and is the approach adopted in this study.

The general outline of the simulation algorithm for the configuration field method is the same as in the standard simulation algorithm for homogeneous flows, given in Section 2.2. However, instead of evolving an ensemble of configuration vectors, or polymer molecules, $\{\boldsymbol{Q}_i^k(t)\}_{k=1}^{N_{\rm p}}$, an ensemble of configuration fields, $\{\boldsymbol{Q}_i^k(\boldsymbol{x},t)\}_{k=1}^{N_{\rm f}}$, is now evolved according to Eqs. (9) and (10). Similar to the Lagrangian approach, associated with each configuration field in the ensemble is a lifetime during which the whole field evolves in the flow field $\boldsymbol{\kappa}(\boldsymbol{x},t)$ according to Eqs. (9) and (10). Once the lifetime is reached, the entire configuration field is destroyed and regenerated according to the standard Gaussian distribution. When initially generated, the configuration fields are spatially uniform.

For steady-state flow problems, such as the one considered in this study, the time steps in the macroscopic and microscopic parts of the algorithm have different roles. The time step used in the momentum equation is chosen to improve the convergence of the iterative procedure, while the time step used to discretize Eqs. (9) and (10) should be chosen to compute the configuration fields, and hence polymer stress, accurately from the current iterate's guess to the velocity field. That is, in each iterate of the decoupled procedure, the configuration fields and polymer stress are computed from the current guess to the steady-state velocity field, and not from a true time-dependent history of the velocity field. In this case, the velocity gradient in Eqs. (9) and (10) is independent of time. The evolution of the configuration fields continues until steady-state values of polymer stress are reached, which usually takes two to three times the longest relaxation time of the fluid.

The equations of motion, Eqs. (9) and (10), are spatially discretized using the discontinuous Galerkin method. The advantage of this method is that it reduces the coupling between elements and allows the semi-discrete equations for the configuration fields to be solved at the element level. This reduces the computational costs since it avoids the necessity to solve large systems of equations for each of the $N_{\rm f}$ configuration fields in the ensembles.

The weak forms of the Eqs. (9) and (10) are obtained by multiplying each equation by a test function $\boldsymbol{w}$ and integrating over an arbitrary triangular element Ω^{e}. After integrating the convective term by parts twice, the resulting weak equations can be stated as follows on each Ω^{e}:

$$\int_{\Omega^{\mathrm{e}}}\left(\frac{\partial \boldsymbol{Q}_1}{\partial t}+\boldsymbol{u}\cdot\nabla\boldsymbol{Q}_1-\boldsymbol{\kappa}\cdot\boldsymbol{Q}_1\right)\cdot\boldsymbol{w}\,\mathrm{d}\Omega^{\mathrm{e}}+\int_{\partial\Omega^{e-}}\left[(\boldsymbol{n}\cdot\boldsymbol{u})\left(\boldsymbol{Q}_1^{\mathrm{E}}-\boldsymbol{Q}_1\right)\right]\cdot\boldsymbol{w}\,\mathrm{d}\Gamma=0, \tag{11}$$

$$\int_{\Omega^{\mathrm{e}}}\left(\frac{\partial \boldsymbol{Q}_2}{\partial t}+\boldsymbol{u}\cdot\nabla\boldsymbol{Q}_2+\boldsymbol{\kappa}^{\dagger}\cdot\boldsymbol{Q}_2\right)\cdot\boldsymbol{w}\,\mathrm{d}\Omega^{\mathrm{e}}+\int_{\partial\Omega^{e-}}\left[(\boldsymbol{n}\cdot\boldsymbol{u})\left(\boldsymbol{Q}_2^{\mathrm{E}}-\boldsymbol{Q}_2\right)\right]\cdot\boldsymbol{w}\,\mathrm{d}\Gamma=0. \tag{12}$$

In these equations, $\partial\Omega^{\mathrm{e}-}$ denotes the inflow boundary of element Ω^{e} defined by

$$\partial\Omega^{e-}=\{\boldsymbol{x}\in\partial\Omega^{\mathrm{e}}:\boldsymbol{u}(\boldsymbol{x})\cdot\boldsymbol{n}(\boldsymbol{x})<0\},$$

where $\boldsymbol{n}(\boldsymbol{x})$ is the outward pointing unit normal to the boundary $\partial\Omega^{\mathrm{e}}$. Also, $\boldsymbol{Q}_i^{\mathrm{E}}$ denotes the values of $\boldsymbol{Q}_i$ in the upstream neighboring element(s). This quantity is convected across the inflow boundary, $\partial\Omega^{\mathrm{e}-}$, of the element Ω^{e}. If the computational domain has an inlet boundary, then the values of $\boldsymbol{Q}_i^{\mathrm{E}}$ in elements abutting this boundary come from the predecessor flow, that is, the flow before the inlet boundary. At every cross-section that is parallel to the inlet boundary, the predecessor flow is assumed to have the same velocity profile as that along the inlet boundary. For the crossed-triangle element considered here, the predecessor flow is locally homogeneous, so that the standard simulation algorithm for homogeneous flows (Section 2.2) can be used to evolve the configuration fields in the predecessor flow, and hence to compute $\boldsymbol{Q}_i^{\mathrm{E}}$ in triangles with a side along the inlet boundary.

The elements in the finite element mesh are numbered so that, in the absence of recirculation, the values of $\boldsymbol{Q}_i^{\mathrm{E}}$ convected into Ω^{e} from the neighboring upstream elements are known. If a region of recirculation does exist, then at any time, the value of $\boldsymbol{Q}_i^{\mathrm{E}}$, and hence $\boldsymbol{Q}_i$, in an element within this region depends on itself. To eliminate or reduce this dependency, an iterative process based on successive substitution can be used. That is, instead of computing $\boldsymbol{Q}_i$ only once per time step, the calculation can be iterated several times in this region, with each iterate producing updated values of $\boldsymbol{Q}_i$ and $\boldsymbol{Q}_i^{\mathrm{E}}$. If the recirculation region is weak, which is the case in die entry flow problems such as the one considered here, then one or two iterates should be sufficient. In fact, it was shown by Feigl et al. [14] that even taking the flow to be Newtonian in the elements contained in the recirculation region of a die entry domain did not affect the results (for an Oldroyd-B fluid) or its comparison to the results of equivalent macroscopic calculations.

During the lifetimes s_1 and s_2, the configuration fields $\boldsymbol{Q}_1(\boldsymbol{x},t)$ and $\boldsymbol{Q}_2(\boldsymbol{x},t)$ are obtained by solving Eqs. (11) and (12). Two types of spatial approximations for the configuration field variables are considered. These are discussed below.

3.1. Piecewise constant approximation to configuration fields

First, the approximation space for $\boldsymbol{Q}_i$, $i=1,2$ is taken to be the space of piecewise constant polynomials on the triangles of the mesh. The resulting stress field is then piecewise constant in each triangle. This choice is consistent with the constant pressure approximation and linear velocity approximation on triangles in the macro part of the algorithm. This choice is furthermore motivated by the fact that our standard, macro-macro algorithm for solving viscoelastic flow problems using a macroscopic integral constitutive equation also assumes triangle-wise constant stress, and this program has been shown to perform very well [3,10,13,16]. With this approximation, the convection of configuration fields in each triangle comes only through the boundary integral term.

Denoting the approximations to $\boldsymbol{Q}_1(\boldsymbol{x},t)$ and $\boldsymbol{Q}_2(\boldsymbol{x},t)$ in triangle Ω^{e} by the 3×1 column vectors $\boldsymbol{Q}_1^{\mathrm{e}}=\boldsymbol{Q}_1^{\mathrm{e}}(t)$ and $\boldsymbol{Q}_2^{\mathrm{e}}=\boldsymbol{Q}_2^{\mathrm{e}}(t)$, respectively, yields the spatially discrete equations

$$\frac{\mathrm{d}\boldsymbol{Q}_1^{\mathrm{e}}(t)}{\mathrm{d}t}=(c\boldsymbol{\delta}+\boldsymbol{\kappa})\cdot\boldsymbol{Q}_1^{\mathrm{e}}(t)+\boldsymbol{b}_1(t), \tag{13}$$

$$\frac{\mathrm{d}\boldsymbol{Q}_2^{\mathrm{e}}(t)}{\mathrm{d}t}=(c\boldsymbol{\delta}-\boldsymbol{\kappa}^{\dagger})\cdot\boldsymbol{Q}_2^{\mathrm{e}}(t)+\boldsymbol{b}_2(t), \tag{14}$$

where

$$\boldsymbol{b}_i(t) = -\frac{1}{\mu(\Omega^{\mathrm{e}})}\int_{\partial\Omega^{e-}} \boldsymbol{Q}_i^{\mathrm{eE}}(t)(\boldsymbol{n}\cdot\boldsymbol{u})\,\mathrm{d}\Gamma, \quad i = 1,2,$$
$$c = \frac{1}{\mu(\Omega^{\mathrm{e}})}\int_{\partial\Omega^{e-}} (\boldsymbol{n}\cdot\boldsymbol{u})\,\mathrm{d}\Gamma.$$

In these equations, the velocity $\boldsymbol{u}$ is the discrete velocity field which is known from the previous iterate of our decoupled solution algorithm, and $\mu(\Omega^{\mathrm{e}})$ represents the area of triangle Ω^{e}. Furthermore, $\boldsymbol{Q}_i^{\mathrm{eE}}$ denotes the approximate values of $\boldsymbol{Q}_i$ in the upstream neighboring triangle(s) or in the predecessor flow.

The above semi-discrete system of equations, together with initial conditions at t', forms a first-order, linear initial value problem. The type of velocity–pressure finite element used in the macro part of the algorithm leads to an attractive solution procedure for this system in steady flow, such as the die entry flow considered here. Specifically, in steady flow, $\boldsymbol{\kappa}$ and c are constant in each triangle, so that the coefficient matrix of this system is constant. The solutions, $\boldsymbol{Q}_1^{\mathrm{e}}(t,t')$ and $\boldsymbol{Q}_2^{\mathrm{e}}(t,t')$, to Eqs. (13) and (14) may therefore be written as follows in triangle Ω^{e}

$$\boldsymbol{Q}_1^{\mathrm{e}}(t,t') = \mathrm{e}^{c(t-t')}\mathrm{e}^{\boldsymbol{\kappa}(t-t')}\cdot\boldsymbol{Q}_1^{\mathrm{e}}(t') + \int_{t'}^{t} \mathrm{e}^{c(t-s)}\mathrm{e}^{\boldsymbol{\kappa}(t-s)}\cdot\boldsymbol{b}_1(s)\,\mathrm{d}s, \tag{15}$$

$$\boldsymbol{Q}_2^{\mathrm{e}}(t,t') = \mathrm{e}^{c(t-t')}\mathrm{e}^{-\boldsymbol{\kappa}^{\dagger}(t-t')}\cdot\boldsymbol{Q}_2^{\mathrm{e}}(t') + \int_{t'}^{t} \mathrm{e}^{c(t-s)}\mathrm{e}^{-\boldsymbol{\kappa}^{\dagger}(t-s)}\cdot\boldsymbol{b}_2(s)\,\mathrm{d}s. \tag{16}$$

The matrix exponentials involving $\boldsymbol{\kappa}$ and $\boldsymbol{\kappa}^{\dagger}$ depend on the eigenvalues, λ, of these matrices. Since our finite element guarantees pointwise incompressibility of the discrete velocity field, it follows that $\mathrm{trace}(\boldsymbol{\kappa}) = 0$ and the characteristic equation for both matrices reduces to $\lambda^2 + \det(\boldsymbol{\kappa}) = 0$. Therefore, the eigenvalues $\lambda = \pm\sqrt{-\det(\boldsymbol{\kappa})}$ are either distinct and real (if $\det(\boldsymbol{\kappa}) < 0$), pure imaginary (if $\det(\boldsymbol{\kappa}) > 0$), or zero (if $\det(\boldsymbol{\kappa}) = 0$).

Consequently, the matrix exponentials can be computed exactly and very efficiently via

$$\mathrm{e}^{\boldsymbol{\kappa}\Delta t} = \begin{cases} \boldsymbol{\delta} + \boldsymbol{\kappa}\Delta t, & \text{if } \lambda = 0,\\ \cosh(\lambda\Delta t)\boldsymbol{\delta} + \frac{\sinh(\lambda\Delta t)}{\lambda}\boldsymbol{\kappa}, & \text{if } \lambda \neq 0, \end{cases}$$
$$\mathrm{e}^{-\boldsymbol{\kappa}^{\dagger}\Delta t} = \begin{cases} \boldsymbol{\delta} - \boldsymbol{\kappa}^{\dagger}\Delta t, & \text{if } \lambda = 0,\\ \cosh(\lambda\Delta t)\boldsymbol{\delta} - \frac{\sinh(\lambda\Delta t)}{\lambda}\boldsymbol{\kappa}^{\dagger}, & \text{if } \lambda \neq 0. \end{cases}$$

Finally, the integrals in Eqs. (15) and (16) are evaluated by the trapezoid rule, making the time integration scheme second order. Furthermore, the scheme is explicit since the values of $\boldsymbol{Q}_i^{\mathrm{eE}}(t+\Delta t)$ used to compute $\boldsymbol{b}_i(t+\Delta t)$ are known from previous processing of the upstream triangles. Time steps, Δt, are typically between 10^{-7} and 10^{-3}.

3.2. Piecewise linear approximation to configuration fields

An alternative approximation to the configuration fields uses the space of linear polynomials in each triangle. Unlike the piecewise constant approximation, this approximation allows the convection of the configuration fields within the triangles. Let $Q_{1i}(\boldsymbol{x},t)$ and $Q_{2i}(\boldsymbol{x},t)$, $i = 1, 2, 3$, be the ith components of the configuration fields $\boldsymbol{Q}_1(\boldsymbol{x},t)$ and $\boldsymbol{Q}_2(\boldsymbol{x},t)$, respectively, and let $\boldsymbol{Q}_{1i}^{\mathrm{e}}$ and $\boldsymbol{Q}_{2i}^{\mathrm{e}}$ be 3×1 column vectors containing the coefficients in the triangle-wise linear expansion of Q_{1i} and Q_{2i}, respectively. Substituting linear approximations of Q_{1i} and Q_{2i} into Eqs. (11) and (12) yields a system of three equations for each coefficient vector $\boldsymbol{Q}_{1i}^{\mathrm{e}}$ and $\boldsymbol{Q}_{2i}^{\mathrm{e}}$ of the form

$$\boldsymbol{M}\frac{\mathrm{d}\boldsymbol{Q}_{1i}^{\mathrm{e}}}{\mathrm{d}t} + (\boldsymbol{C}+\boldsymbol{B})\boldsymbol{Q}_{1i}^{\mathrm{e}} - \boldsymbol{K}_{ij}\boldsymbol{Q}_{1j}^{\mathrm{e}} = \boldsymbol{F}_{1i}, \tag{17}$$

$$\boldsymbol{M}\frac{\mathrm{d}\boldsymbol{Q}_{2i}^{\mathrm{e}}}{\mathrm{d}t} + (\boldsymbol{C}+\boldsymbol{B})\boldsymbol{Q}_{2i}^{\mathrm{e}} + \boldsymbol{K}_{ji}\boldsymbol{Q}_{2j}^{\mathrm{e}} = \boldsymbol{F}_{2i}, \tag{18}$$

where the 3×3 matrices $\boldsymbol{M}$, $\boldsymbol{C}$, $\boldsymbol{B}$ and $\boldsymbol{K}_{ij}$, and the 3×1 vector $\boldsymbol{F}_{ji}$ are given by

$$\boldsymbol{M} = \int_{\Omega^{\mathrm{e}}} \left(\boldsymbol{\psi}\boldsymbol{\psi}^{\dagger}\right) \mathrm{d}\Omega^{\mathrm{e}},$$
$$\boldsymbol{C} = \int_{\Omega^{\mathrm{e}}} \left(\boldsymbol{\psi}(\boldsymbol{\psi}^{\dagger}\boldsymbol{u}_j)\frac{\partial \boldsymbol{\psi}^{\dagger}}{\partial x_j}\right) \mathrm{d}\Omega^{\mathrm{e}},$$
$$\boldsymbol{K}_{ij} = \int_{\Omega^{\mathrm{e}}} \kappa_{ij}\left(\boldsymbol{\psi}\boldsymbol{\psi}^{\dagger}\right) \mathrm{d}\Omega^{\mathrm{e}},$$
$$\boldsymbol{B} = -\int_{\partial\Omega^{\mathrm{e}-}} (\boldsymbol{n}\cdot\boldsymbol{u})\left(\boldsymbol{\psi}\boldsymbol{\psi}^{\dagger}\right) \mathrm{d}\Gamma,$$
$$\boldsymbol{F}_{ji} = -\int_{\partial\Omega^{\mathrm{e}-}} (\boldsymbol{n}\cdot\boldsymbol{u})\boldsymbol{\psi} Q_{ji}^{\mathrm{eE}} \,\mathrm{d}\Gamma, \quad j = 1, 2.$$

The 3×1 vector $\boldsymbol{\psi} = \boldsymbol{\psi}(\boldsymbol{x})$ contains the linear basis functions of the approximation space, and Q_{1i}^{eE} and Q_{2i}^{eE}, $i = 1,2,3$, denote the approximations to Q_{1i} and Q_{2i}, respectively, in the upstream neighboring triangle(s) corresponding to the inflow boundary $\partial\Omega^{\mathrm{e}-}$ of the current triangle.

In this case, the attractive time integration scheme of the piecewise constant approximation is lost. Therefore, an explicit, second-order Runge–Kutta method is used to solve the initial value problem given in Eqs. (17) and (18). Again, typical time steps are between 10^{-7} and 10^{-3}.

4. Micro–macro simulation results

The performance of the micro–macro algorithm is evaluated by simulating the flow of the PIB solution in an abrupt planar die entry domain. The algorithm is analyzed based on the dependence of the solution on mesh size, ensemble size, type of approximation for the configuration fields, and computational cost. Furthermore, the simulation results are compared with known qualitative behavior of polymers, and the stress values in the downstream fully developed flow region are compared with the simulation results from the standard simulation algorithm described in Section 2.2. As previously mentioned, the values from the standard simulation algorithm can be considered to be exact model predictions in steady shear flow at a given shear rate. Finally, the simulation results are compared with available experimental data taken in related slit-die entry experiments for the PIB solution performed by Quinzani et al. [32].

Following these experiments, the dimensions of the computational die entry domain were as follows: the half-heights of the downstream and upstream channels were $H_{\mathrm{d}} = 0.32$ cm and $H_{\mathrm{u}} = 1.27\ \mathrm{cm} \approx 4H_{\mathrm{d}}$, respectively, and the lengths of the downstream and upstream channels were $L_{\mathrm{d}} = 13\ \mathrm{cm} \approx 40H_{\mathrm{d}}$ and $L_{\mathrm{u}} = 8\ \mathrm{cm} = 25H_{\mathrm{d}}$, respectively. (The width of $W = 25$ cm in the slit-die experiments was neglected in our two-dimensional simulations.) The lengths of the upstream and the downstream channel proved to be long enough to allow the flow to become fully developed for the flow rates considered.

The boundary conditions were as follows. The no-slip boundary condition was enforced along the channel walls and the symmetry boundary condition was imposed along the centerline. Along the outlet boundary, the appropriate (non-Newtonian) fully-developed velocity profile was specified. The velocity profile for the outlet boundary was obtained, for a given centerline velocity, by computing the fully-developed flow of the PIB solution in a channel of uniform width which has the same height as the downstream channel of the die entry domain. Along the inlet boundary, either a (Newtonian) parabolic velocity profile or the appropriate (non-Newtonian) fully-developed velocity profile can be specified. Our numerical experiments showed that it was sufficient to specify a (Newtonian) parabolic velocity profile, since the length of the upstream channel was long

Table 2
Flow rates and corresponding Deborah numbers used in the abrupt die entry simulations

Q (cm^2/s)	De
4.5	1.3
5.6	1.6
8.9	2.6
13.5	3.9

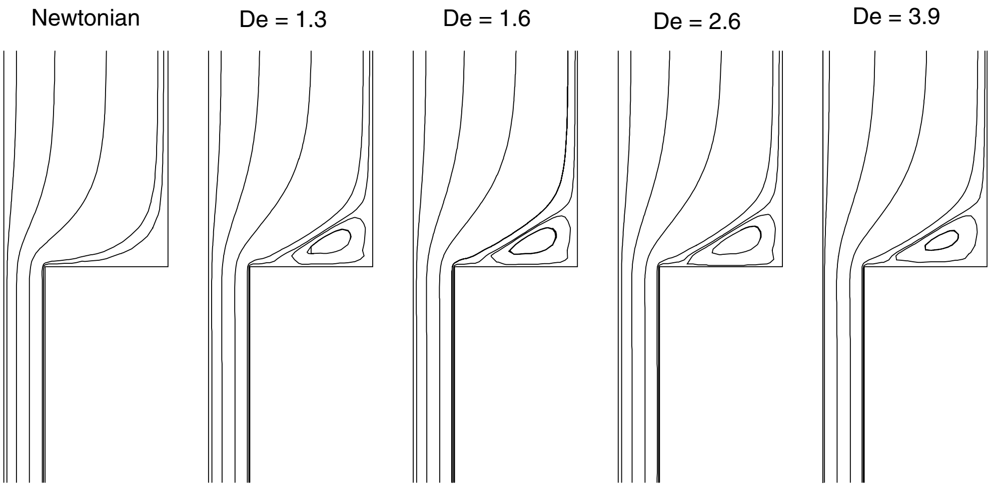

Fig. 2. Predicted streamline patterns in the abrupt die entry domain.

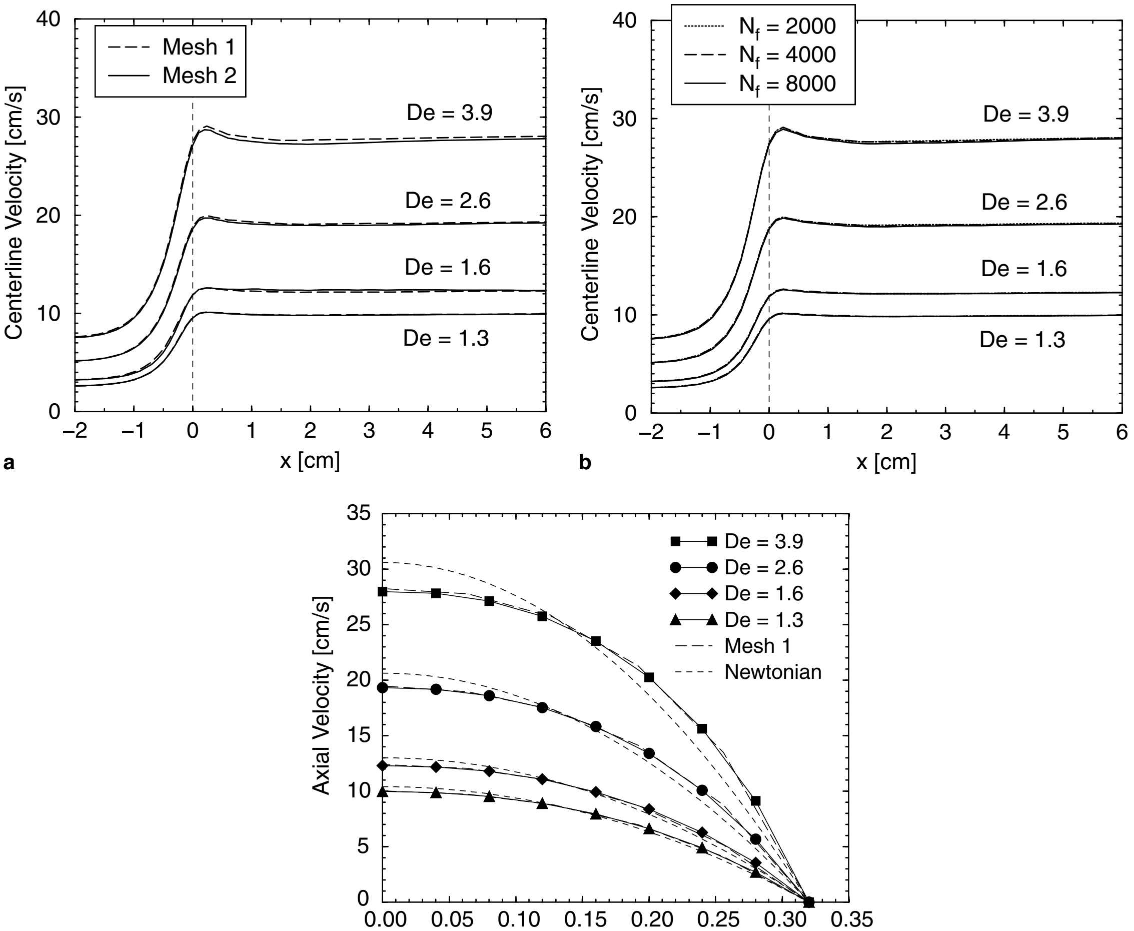

Fig. 3. Predicted velocity profiles in the abrupt die entry domain. (a) Centerline velocity profiles on both meshes for $N_f = 2000$. (b) Centerline velocity profiles for different ensemble sizes. (c) Fully developed downstream axial velocity profiles at $x = 10$cm on both meshes for $N_f = 2000$. Symbols represent results on Mesh 2.

enough for the appropriate fully-developed non-Newtonian velocity and stress profiles to be reached well before the die entry region.

The simulations were performed on two finite element meshes. The first mesh contained 651 rectangular macro-elements and 1357 velocity nodes, and the second mesh consisted of 1200 macro-elements and 2514 velocity nodes. (Recall that each rectangle consists of four triangular elements.) Each mesh is nonuniform with smaller elements closer to the die entry. The typical size of the rectangles near the die entry region (non-dimensionalized by H_d) was 0.375×0.2 in Mesh 1 and 0.3125×0.125 in Mesh 2, leading to an almost 50% reduction in the area of the elements.

Results are presented for four volumetric flow rates which correspond to cases considered in the experiments. Table 2 shows the flow rates, Q, used in the simulations as well as the corresponding Deborah numbers, De. The Deborah number was defined as $De = \overline{\lambda}\dot{\gamma}$, where $\overline{\lambda} = (\sum \eta_k \lambda_k)/(\sum \eta_k)$ is a weighted average of relaxation times and $\dot{\gamma} = \overline{V}/H_d$ is an average shear rate in the downstream channel, based on the average velocity $\overline{V} = Q/(2H_d)$.

4.1. Analysis of die entry results

The simulation results are first evaluated numerically in terms of the qualities stated at the beginning of this section. Unless otherwise noted, attention is first restricted to the results obtained using the piecewise constant

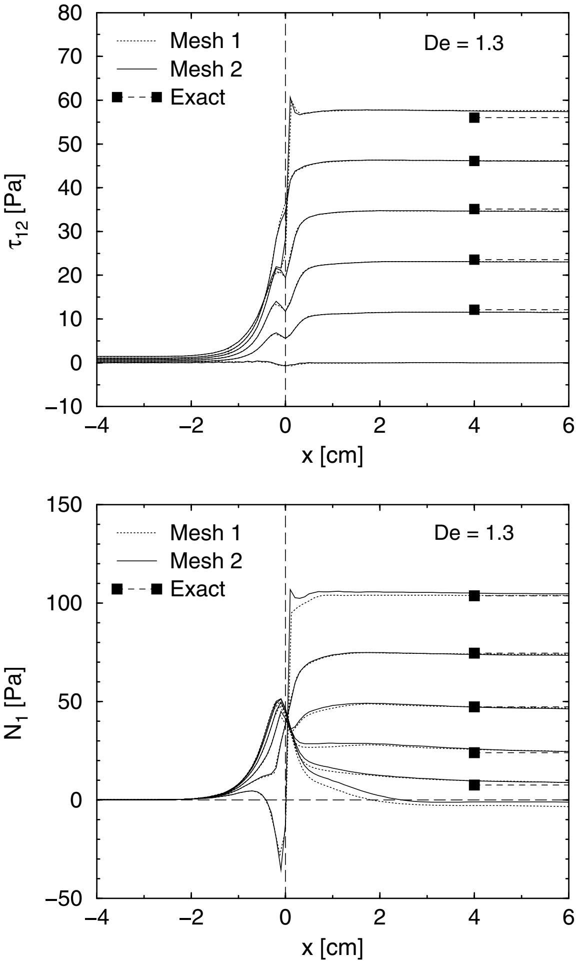

Fig. 4. Comparison of stress values from the micro–macro simulation algorithm along several axial cross-sections (labeled by Y) for $De = 1.3$. From bottom to top, the curves in each graph correspond to $Y = 0.0$ (centerline), 0.2, 0.4, 0.6, 0.8, 1.0 (wall of downstream channel). The results from both meshes are shown and $N_f = 2000$. The exact model predictions in fully developed downstream flow are shown as horizontal line segments labeled with symbols.

approximation to the configuration fields. After that, the effect of the piecewise linear approximation is considered.

Fig. 2 shows the streamline patterns from the micro–macro simulations for the flow rates considered. Also shown is the streamline pattern for Newtonian flow. A significant vortex, which did not grow with increased flow rate, is observed for the PIB solution. The presence of the vortex at the reentrant corner is typical for viscoelastic flows in abrupt contraction domains (cf. [13]). At all flow rates, the size of the vortex, $\chi = L_v/(2H_u)$, remained at approximately 0.24, where L_v is the vortex detachment length along the upstream wall. Lack of vortex growth is often observed in planar abrupt contractions (cf. [13]). The size of the vortex in the experiment was not reported. The streamline patterns were independent of mesh, ensemble size and type of approximation to the configuration fields.

The mesh and ensemble size independence of the velocity field is also evidenced in Fig. 3, which compares the centerline velocity profiles for both meshes (Fig. 3(a)) and for three different ensemble sizes (Fig. 3(b)). The die entry is at $x = 0$, with $x < 0$ and $x > 0$ corresponding to the upstream and downstream channels, respectively. This figure also shows that there is a velocity overshoot that occurs within the die which increases with flow rate. This behavior is again typical for viscoelastic flows in abrupt die entry domains. The axial velocity profiles at various cross-sections also showed mesh and ensemble size independence. This is illustrated in Fig. 3(c) which shows the fully developed velocity profiles downstream at $x = 10$ cm for both meshes. The

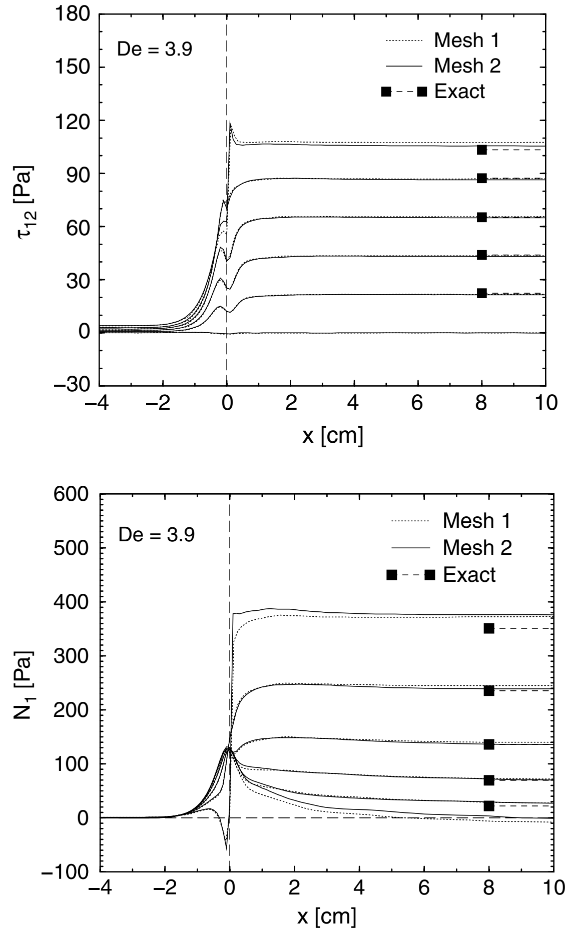

Fig. 5. Comparison of stress values from the micro–macro simulation algorithm along several axial cross-sections (labeled by Y) for $De = 3.9$. From bottom to top, the curves in each graph correspond to $Y = 0.0$, 0.2, 0.4, 0.6, 0.8, 1.0. The results from both meshes are shown and $N_f = 2000$. The exact model predictions in fully developed downstream flow are shown as dashed horizontal line segments labeled with symbols.

shear-thinning behavior of the fluid is seen by comparison with the Newtonian velocity profiles at the same flow rates.

The stress values, τ_{12} and $N_1 = \tau_{11} - \tau_{22}$, computed from the micro–macro simulations are plotted along lines that are parallel to the centerline and run through the whole domain, from inlet boundary to outlet boundary. Each line is labeled by its distance, Y, from the centerline, normalized by the half-height of the downstream channel. The line $Y = 0$ represents the centerline and $Y = 1$ represents the line that coincides with the wall of the downstream channel.

Figs. 4 and 5 show the values of τ_{12} and the first normal stress difference $N_1 = \tau_{11} - \tau_{22}$ for several Y values for the lowest and highest flow rate considered. The results shown were computed with $N_f = 2000$ configuration fields and the curves from both meshes are shown. The figures show that the stress values are mesh independent, except close to the die entry in some cases.

In each graph of Figs. 4 and 5, the dashed horizontal lines labeled with the symbols represent the exact model predictions computed in homogeneous shear flow from the standard simulation algorithm. For each Y value, the shear rate used in the standard algorithm was taken to be the (essentially) constant value of shear rate computed by the micro–macro algorithm in the downstream channel. The figures show that there is very good agreement between the exact model predictions for τ_{12} and N_1 and the micro–macro results. Possible exceptions are along the wall of the downstream channel or perhaps along the centerline at the high flow rate. This discrepancy is not due to the ensemble size, since increasing N_f had only a small, if any, effect on the stress

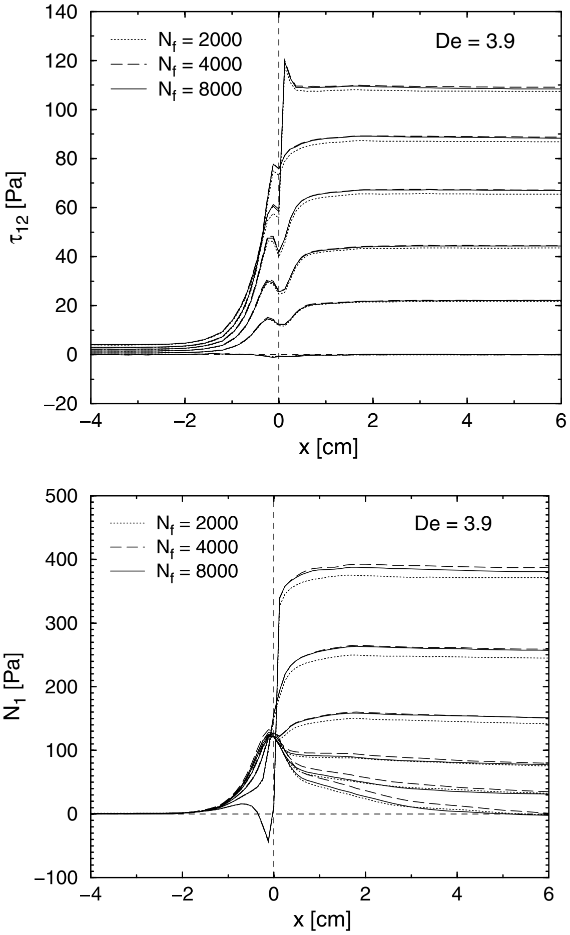

Fig. 6. Effect of ensemble size, N_f, on the computed stress values along several axial cross-sections (labeled by Y) for $De = 3.9$. From bottom to top, the curves in each graph correspond to $Y = 0.0$, 0.2, 0.4, 0.6, 0.8, 1.0.

values and their comparisons to the exact model predictions. This is shown in Fig. 6. As discussed shortly, this discrepancy is also not due to the piecewise constant approximation to the configuration fields.

A large contributing factor to the discrepancy near the computational boundary could be due to the smoothing and extrapolation postprocessing of the discontinuous velocity gradient and stress fields as discussed in Section 3. This hypothesis is based on the fact that good agreement is obtained except possibly along the boundary, and on our past experience with the smoothing and extrapolation postprocessing scheme when solving die entry flow problems using macroscopic viscoelastic constitutive equations. Note that if this hypothesis is true, then mesh refinement close to the wall should improve the agreement. Figs. 4 and 5 show that mesh refinement does decrease the error in N_1 at the centerline in the downstream shear flow, bringing the values closer to zero. However, mesh refinement did not significantly improve the agreement along the wall in the downstream channel. It could be that additional mesh refinement close to the channel wall is needed in order to see significant improvements. Further investigation is needed to resolve this issue definitively.

The effect of a piecewise linear approximation to the configuration fields is now considered. First, the velocity field was seen to be mostly independent of whether a triangle-wise constant or linear approximation to the configuration fields was used. This is shown in Fig. 7 which compares the results of these two approximations for $N_f = 2000$ along the centerline and at a cross-section at $x = 10$cm in the downstream channel. Fig. 7(a) shows that at the lower or moderate flow rates, the centerline velocities coincide, while at the higher flow rates,

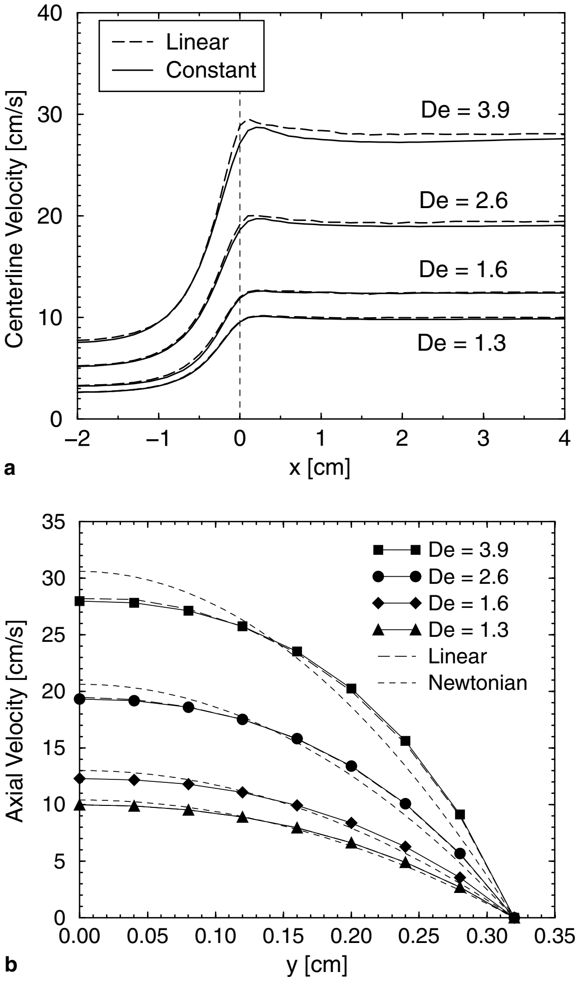

Fig. 7. Comparison of velocity profiles using the piecewise constant and piecewise linear approximation to the configuration fields. (a) Centerline velocity profiles using $N_f = 2000$. (b) Fully developed downstream axial velocity profiles at $x = 10$cm using $N_f = 2000$.

the centerline velocity computed using the piecewise linear approximation to the configuration fields is somewhat larger close to the die entry. Both graphs show that away from the die entry, the respective velocities coincide at all flow rates.

The dependence of the stress field on the type of configuration field approximation is illustrated in Figs. 8 and 9 for $De = 1.3$ and $De = 2.6$, respectively. As before, each pair of curves corresponds to values of τ_{12} and N_1 at (dimensionless) constant distances, Y, from the centerline. The figures show that τ_{12} is virtually independent of the order of the approximation for configuration fields. The values of N_1 also agree generally well. The exceptions are near the die entry, particularly closer to the centerline.

The performance of the micro–macro algorithm was also evaluated in terms of its computational costs. It was found that the computational time per triangle per iterate was approximately proportional to the ensemble size and the number of elements. Moreover, using the piecewise linear approximations to the configuration fields required approximately 2.5 times more computational time than using the piecewise constant approximations for a given mesh and ensemble size. This factor of 2.5 is due primarily to the fact that the time integration scheme used for the piecewise linear approximation requires twice as many time steps, or function evaluations, as that required by the piecewise constant approximation, although both schemes are second order. This illustrates one benefit of working with the piecewise constant approximation; more precisely, because of the type of velocity and pressure approximation, the piecewise constant approximation to the configuration fields leads to an efficient time integration scheme. The remaining increase in computational time

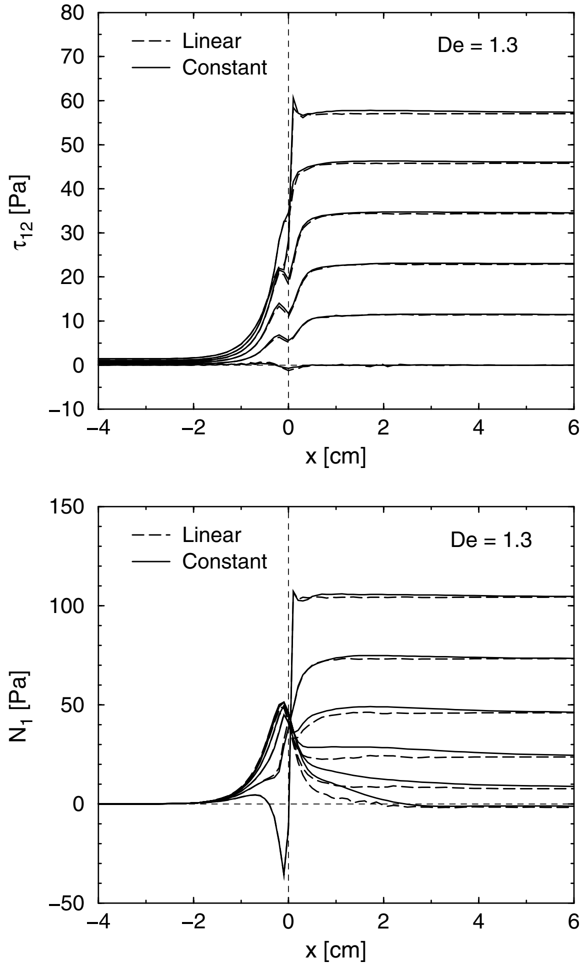

Fig. 8. Comparison of stress computed with piecewise constant and piecewise linear approximation to configuration fields at $De = 1.3$. From bottom to top, the curves in each graph correspond to $Y = 0.0, 0.2, 0.4, 0.6, 0.8, 1.0$.

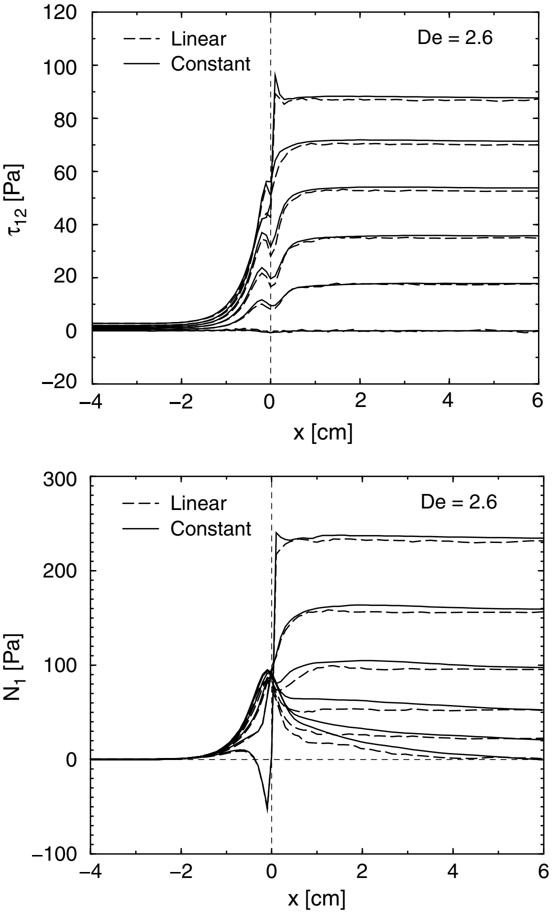

Fig. 9. Comparison of stress computed with piecewise constant and piecewise linear approximation to configuration fields at $De = 2.6$. From bottom to top, the curves in each graph correspond to $Y = 0.0$, 0.2, 0.4, 0.6, 0.8, 1.0.

when going from a piecewise constant approximation to a piecewise linear approximation to the configuration fields is due to the threefold increase in the number of configuration field variables (degrees of freedom) and the temporally discretized equations involving them. The storage requirements for the piecewise linear approximation are also larger due to the threefold increase in the number of configuration field variables. The relevance of any differences that exist between the simulation results from the two types of approximations to the configuration fields should be judged in view of the computational costs associated with each.

4.2. Comparison with experimental data

The micro–macro simulation results are now compared with the experimental data reported by Quinzani et al. [32]. The experimental data consisted of the velocity, elongation rate and first normal stress difference along the centerline of the slit-die entry domain. The velocity was measured using laser doppler velocimetry while the stress values were measured using flow-induced birefringence. The elongation rates were derived from the measured velocity field. The experimental data displayed here were taken from Quinzani et al. [32].

We point out that the Deborah numbers, *De*, used in the three-dimensional slit-die experiments were slightly larger than the Deborah numbers used in the two-dimensional simulations. This difference resulted from how we chose the outlet boundary velocity profiles in the simulations. Specifically, we computed an outlet velocity profile as the fully-developed flow of the model fluid in a channel of constant height with a specified centerline velocity which we took from the experiment.

A comparison of the predicted centerline velocity and elongation rates with the experimental data is given in Fig. 10. In each graph, the open symbols represent experimental data. In Fig. 10(a), the computed centerline velocity is shown from both types of approximation to the configuration fields. This graph shows relatively good agreement between simulation and experiment in the die entry region. One notable discrepancy is that the experimental data does not indicate the significant velocity overshoot that the simulation results show at the highest flow rate and that is typical of polymeric fluids.

Fig. 10(b) shows the comparison of centerline elongation rates. Simulation results, indicated by the closed symbols, are shown only for the case of piecewise constant approximation to the configuration fields. Both the simulation and the experiment show that the maximum elongation rate is reached immediately before the die entry which is a characteristic behavior in die entry flow [13]. Generally good agreement between the computed elongation rates and the experiment is observed. The discrepancy in values is due to the difference in the velocity fields close to the die entry. The simulation predicts a more rapid increase in the velocity field than the experiment indicates. The centerline elongation rates resulting from the piecewise linear approximation to the configuration fields are graphically identical at the lower flow rates and reach a higher maximum at the highest two flow rates. This is clear from the more rapid increase in centerline velocity at these flow rates immediately before the die entry, as shown in Fig. 10(a).

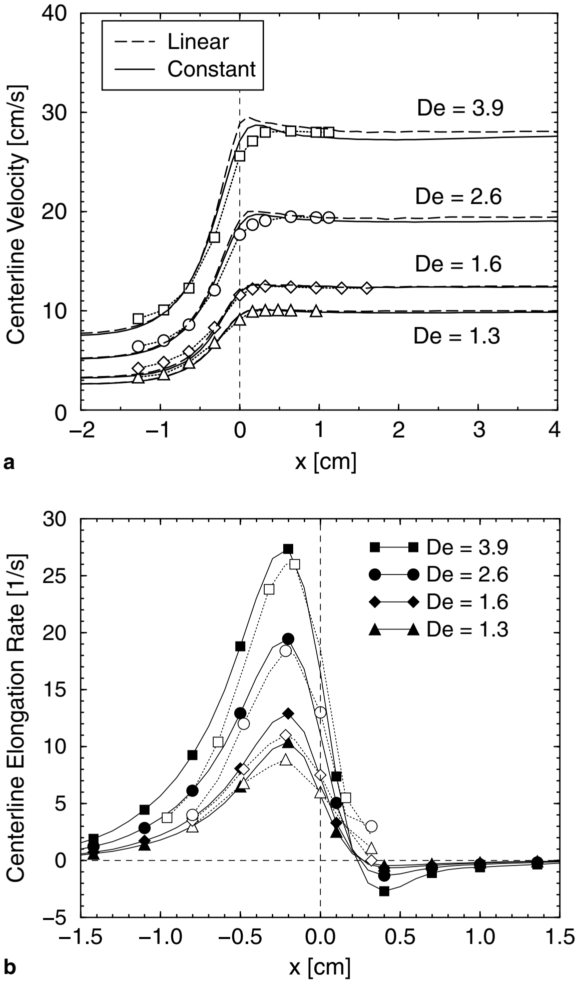

Fig. 10. Comparison of micro–macro simulation results with experimental data of Quinzani et al. [32]. (a) Centerline velocity profiles. (b) Centerline elongation rates (for piecewise constant approximation to the configuration fields). Curves with closed symbols or no symbols correspond to simulation results. Curves with open symbols correspond to experimental data. For graphical clarity, a symbol is not shown at every simulation and experimental data point in the lower graph.

The comparison between the computed and measured centerline first normal stress difference, N_1, is shown in Fig. 11. Simulation results are shown from the piecewise constant approximation to the configuration fields (a) and the piecewise linear approximation (b). The closed symbols represent the simulation results. Both graphs show that the simulation overpredicts the values of N_1 close to the die entry and, in both cases, the discrepancy between simulation and experiment increases with increased flow rate. The graphs also show a reduction in the computed values of N_1 close to the die entry when the higher order approximation is used, and this reduction leads to better agreement with the experimental data.

In general, the piecewise linear approximation provides better agreement with experiments. Although Fig. 10 indicates that the piecewise constant approximation gives slightly better agreement with the experimental centerline velocity in a limited case (i.e., close to the die entry and at the highest flow rate), this difference in agreement is relatively small compared to the better agreement that the piecewise linear approximation gives to the experimental centerline normal stress difference at all flow rates considered (Fig. 11).

There are several possible reasons for the discrepancy between the simulated and measured centerline velocity and first normal stress difference. These include (i) inadequacies in our two-dimensional simulation of the three-dimensional experiments, (ii) the inability of the rheological model to describe the PIB fluid sufficiently well, (iii) numerical error, and (iv) experimental error. We have already mentioned one possible source of numerical error: our postprocessing scheme may produce errors in N_1 along the computational boundary,

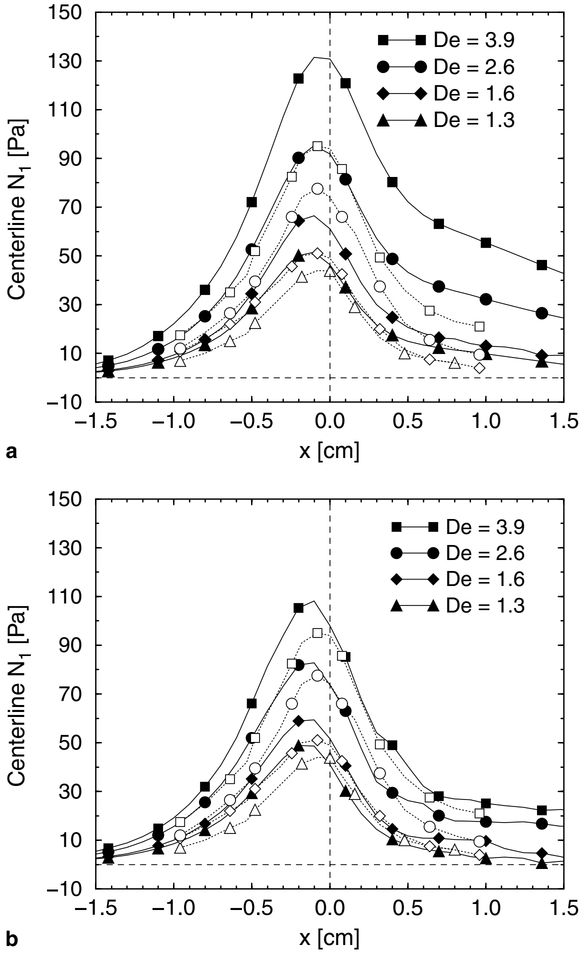

Fig. 11. Comparison of computed and measured centerline first normal stress difference. Simulation results are from piecewise constant approximation (a) and piecewise linear approximation (b) to the configuration fields. Experimental data (open symbols) are from graphs in Quinzani et al. [32]. For graphical clarity, a symbol is not shown at every simulation and experimental data point.

such as the centerline. Another source of numerical error is the approximation error related to the discretization of the configuration fields since, as Fig. 11 shows, the values of centerline N_1 near the die entry depend on the type of spatial approximation to the configuration fields. Perhaps higher order approximations for the configuration fields (and possibly velocity) would further improve agreement with experiments in this problematic region of the domain. Finally, the rheological model is probably a contributing factor to the discrepancy between simulation and experiment. The dynamics of the rheological model presented in Section 2 are most likely an oversimplification for the fluid and require improvements before the model can predict more accurately all aspects of flow in complex domains such as the die entry domain.

5. Summary and conclusions

A multiscale, or micro–macro, algorithm for simulating the flow of a polymeric fluid was described and tested numerically. The multiscale model consisted of the mass and momentum balance equations from continuum mechanics and a rheological model for polymer stress which combines aspects of network, reptation and continuum models. The algorithm iterates two main steps, corresponding to the macro and micro parts of the algorithm. In the macro part of the algorithm, the velocity and pressure fields are updated by solving the mass and momentum balance equations using standard finite element techniques, where the polymer stress is treated as a known pseudo-body force. In the micro part of the algorithm, the updated velocity field is used to compute an update to the polymer stress from the rheological model. This was done using stochastic simulation techniques based on the configuration field formulation of the evolution equations for the two configuration variables in the model. The discontinuous Galerkin method was used to discretize these evolution equations spatially, using either triangle-wise constant or linear approximations. The temporal discretization depended on the type of configuration field approximation, but in both cases, was second order.

The algorithm was evaluated numerically by simulating the flow of a concentrated PIB solution in a planar abrupt die entry domain. At each flow rate considered, simulations were performed using two meshes (standard and refined), three ensemble field sizes, and the two types of configuration field approximation. The algorithm's performance was evaluated in terms of the dependence of the results on these three factors, and in terms of its qualitative and quantitative predictions and the computational cost.

It was found that the velocity and stress fields were virtually independent of mesh size and configuration field ensemble size. There was some dependence of the velocity and stress on the type of configuration field approximation. Away from the die entry, the velocity and stress from the two types of approximation always agreed. However, close to the die entry, where the velocity and its gradient vary most rapidly, discrepancies existed in some cases. Specifically, at the higher flow rates, the linear approximation predicted a faster increase in centerline velocity and a slightly larger overshoot. Moreover, the linear approximation generally predicted smaller normal stress differences in the die entry region, particularly closer to the centerline. The shear stress, on the other hand, was seen to be virtually independent of the type of configuration field approximation.

In all cases, the algorithm predicted the correct qualitative behavior of polymeric fluids in this type of flow domain, e.g., significant vortex in the corner of the upstream channel, centerline velocity overshoot near die entry, shear-thinning velocity profiles and large normal stress differences in the fully-developed downstream channel flow. Moreover, the polymer stress values in the fully-developed downstream channel flow agreed quantitatively with the exact model predictions computed from the standard simulation algorithm at the corresponding shear rate.

Comparisons were also made with available experimental data on the PIB solution taken in a slit-die entry domain. While the centerline velocity agreed relatively well, the simulations overpredicted the centerline normal stress difference close to the die entry. This discrepancy was considerably smaller when the linear configuration field approximation was used, as opposed to the constant approximation. This indicates that numerical approximation error in the configuration fields may be a large cause for the discrepancy between simulation and experiment, and that the discrepancy may be further reduced with higher order approximations or additional mesh refinement. Given that there is currently no rheological model that is completely satisfactory in terms of comparison with experimental data in complex flows, another significant contributing cause for the discrepancy could be the rheological model. Indeed, one purpose and motivation of performing micro–macro simulations is to test microscopic-based models in complex flows and thereby assist in model development.

Finally, in terms of computational costs, there was a significant advantage in working with the piecewise constant configuration field approximation for the given model. The computational time per iterate was nearly 2.5 times higher when the linear configuration field approximation was used compared to the constant approximation. This is mostly due to the fact that the pointwise incompressibility guaranteed by our particular velocity–pressure finite element leads to an attractive and efficient second-order time integration scheme for the spatially discrete evolution equations in the piecewise constant case. In short, only half as many function evaluations, or time steps, are needed for the constant approximation compared to the linear approximation, using a second-order time integration for both methods. Moreover, the linear approximation requires the storage and processing of three times as many degrees of freedom for the configuration variables.

The question of which configuration field approximation is better depends on the relative priorities of the evaluation criteria. Based on the performance of the two approaches, a more optimal algorithm may be found by combining them, that is, by using the piecewise constant approximation to the configuration fields thoughout most of the domain and the piecewise linear approximation in regions of high velocity gradient (e.g., near the die entry). In this way, the higher order spatial approximation of the latter approach can be achieved in the high gradient region while the efficiency of the the former approach can be retained as much as possible.

Acknowledgment

This work was supported by the National Science Foundation under Grant NSF DMS–0096847.

References

[1] T.W. Bell, G.H. Nyland, M.D. Graham, J.J. de Pablo, Combined Brownian dynamics and spectral method simulations of the recovery of polymeric fluids after shear, Macromolecules 30 (6) (1997) 1806–1812.

[2] B. Bernstein, K. Feigl, E.T. Olsen, A first order exactly incompressible finite element for axisymmetric fluid flow, SIAM J. Num. Anal. 33 (5) (1996) 1736–1758.

[3] B. Bernstein, K. Feigl, E.T. Olsen, Steady flows of viscoelastic fluids in axisymmetric abrupt contraction geometry: a comparison of numerical results, J. Rheol. 38 (1) (1994) 53–71.

[4] B. Bernstein, D.S. Malkus, E.T. Olsen, A finite element for incompressible plane flows of fluids with memory, Int. J. Num. Meth. Fluids 5 (1985) 43–70.

[5] P. Biller, F. Petruccione, The flow of dilute polymer solutions in confined geometries: a consistent numerical approach, J. Non-Newtonian Fluid Mech. 25 (1987) 347–364.

[6] R.B. Bird, R.C. Armstrong, O. Hassager, in: Dynamics of Polymeric Liquids, second ed.Fluid Mechanics, vol. 1, Wiley-Interscience, New York, 1987.

[7] R.B. Bird, C.F. Curtiss, R.C. Armstrong, O. Hassager, in: Dynamics of Polymeric Liquids, second ed.Kinetic Theory, vol. 2, Wiley-Interscience, New York, 1987.

[8] X. Fan, Molecular models and flow calculations. II. Simulation of steady planar flow, Acta Mech. Sin. 5 (1989) 216–226.

[9] K. Feigl, D.C. Senaratne, Calculation of the die entry flow of a concentrated polymer solution using micro–macro simulations, J. Fluid Eng., 128 (1) (2006).

[10] K. Feigl, F.X. Tanner, B.J. Edwards, J.R. Collier, A numerical study of the measurement of elongational viscosity of polymeric fluids in a semihyperbolically converging die, J. Non-Newtonian Fluid Mech. 115 (2–3) (2003) 191–215.

[11] K. Feigl, H.C. Öttinger, The equivalence of the class of Rivlin–Sawyers equations and a class of stochastic models for polymer stress, J. Math. Phys. 42 (2) (2001) 796–817.

[12] K. Feigl, H.C. Öttinger, A new class of stochastic simulation models for polymer stress calculation, J. Chem. Phys. 109 (2) (1998) 815–826.

[13] K. Feigl, H.C. Öttinger, A numerical study of the flow of a low-density polyethylene melt in a planar contraction and comparison to experiments, J. Rheol. 40 (1) (1996) 21–35.

[14] K. Feigl, M. Laso, H.C. Öttinger, The CONNFFESSIT approach for solving a two-dimensional viscoelastic fluid problem, Macromolecules 28 (9) (1995) 3261–3274.

[15] K. Feigl, M. Laso, H.C. Öttinger, A two dimensional test problem for the CONNFFESSIT approach to solving viscoelastic fluid problems, in: C. Gallegos (Ed.), Progress and Trends in Rheology IV, Proceedings of the Fourth European Rheology Conference (Steinkopff, Darmstadt), 1994, pp. 347–349.

[16] K. Feigl, H.C. Öttinger, The flow of a LDPE melt through an axisymmetric contraction – A numerical study and comparison to experimental results, J. Rheol. 38 (4) (1994) 847–874.

[17] P. Halin, G. Lielens, R. Keunings, V. Legat, The Lagrangian particle method for macroscopic and micro–macro viscoelastic flow computations, J. Non-Newtonian Fluid Mech. 79 (1998) 387–403.

[18] C.C. Hua, J.D. Schieber, Viscoelastic flow through fibrous media using the CONNFFESSIT approach, J. Rheol. 42 (1998) 477–491.

[19] C.C. Hua, J.D. Schieber, Application of kinetic theory models in spatiotemporal flows for polymer solutions, liquid crystals and polymer melts using the CONNFFESSIT approach, Chem. Eng. Sci. 51 (9) (1996) 1473–1485.

[20] T.J.R. Hughes, W.K. Liu, A. Brooks, Finite element analysis of incompressible viscous flows by the penalty function formulation, J. Comp. Phys. 30 (1979) 1–60.

[21] M.A. Hulsen, A.P.G. van Heel, B.H.A.A. van den Brule, Simulation of viscoelastic flows using Brownian configuration fields, J. Non-Newtonian Fluid Mech. 70 (1997) 79–101.

[22] R. Keunings, Micro–macro methods for the multiscale simulation of viscoelastic flow using molecular models of kinetic theory, in: D.M. Binding, K. Walters (Eds.), Rheology Reviews 2004, British Society of Rheology, 2004, pp. 67–98.

[23] M. Laso, M. Picasso, H.C. Öttinger, 2-D time-dependent viscoelastic flow using CONNFFESSIT, AIChE J. 43 (1997) 877–892.

[24] M. Laso, H.C. Öttinger, Calculation of viscoelastic flow using molecular models: the CONNFFESSIT approach, J. Non-Newtonian Fluid Mech. 47 (1) (1993) 1–20.

[25] F.A. Morrison, Understanding Rheology, Oxford University Press, New York, 2001.

[26] J. Nagtigaal, D.M. Parks, J.R. Rice, On numerically accurate finite element solutions in the fully plastic range, Comp. Methods Appl. Mech. Eng. 4 (1974) 153–178.

[27] H.C. Öttinger, Thermodynamically admissible reptation models with anisotropic tube cross sections and convective constraint release, J. Non-Newtonian Fluid Mech. 89 (2000) 165–185.

[28] H.C. Öttinger, B.H.A.A. van den Brule, M.A. Hulsen, Brownian configuration fields and variance reduced CONNFFESSIT, J. Non-Newtonian Fluid Mech. 70 (1997) 255–261.

[29] H.C. Öttinger, Stochastic Processes in Polymeric Fluids, Springer, Berlin, 1996.

[30] H.C. Öttinger, M. Laso, Smart polymers in finite element calculations, in: P. Moldenaers, R. Keunings (Eds.), Theoretical and Applied Rheology. Proceedings of the XIth International Congress on Rheology, vol. 1, Elsevier, Amsterdam, 1992, pp. 286–288.

[31] F. Petruccione, P. Biller, A consistent numerical analysis of the tube flow of dilute polymer solutions, J. Rheol. 32 (1988) 1–21.

[32] L.M. Quinzani, R.C. Armstrong, R.A. Brown, Use of coupled birefringence and LDV studies of flow through a planar contraction to test constitutive equations for concentrated polymer solutions, J. Rheol. 39 (6) (1995) 1201–1228.

[33] L.M. Quinzani, G.H. McKinley, R.A. Brown, R.C. Armstrong, Modeling the rheology of polyisobutylene solutions, J. Rheol. 34 (1990) 705–748.

[34] A.P.G. van Heel, M.A. Hulsen, B.H.A.A. van den Brule, Simulation of Doi-Edwards model in complex flow, J. Rheol. 43 (5) (1999) 1239–1260.