Available online at www.sciencedirect.com
ScienceDirect
Comput. Methods Appl. Mech. Engrg. xxx (2008) xxx–xxx
Computer methods in applied mechanics and engineering
www.elsevier.com/locate/cma

A wave propagation algorithm for viscoelastic fluids with spatially and temporally varying properties ☆

Robert D. Guy [a,*], Aaron L. Fogelson [b]

[a] Department of Mathematics, University of California Davis, One Shields Ave., Davis, CA 95616, United States
[b] Departments of Mathematics and Bioengineering, University of Utah, 155 South 1400 East, Room 233, Salt Lake City, UT 84112, United States

Received 1 March 2007; received in revised form 16 October 2007; accepted 19 November 2007

Abstract

An algorithm is presented for simulating the fluid and stress equations that arise in a continuum model of platelet aggregation [A.L. Fogelson, R.D. Guy, Platelet–wall interactions in continuum models of platelet thrombosis: formulation and numerical solution, Math. Med. Biol. 21 (2004) 293–334]. The model is equivalent to models of viscoelastic fluids with spatially and temporally varying material parameters. A subsystem containing the elastic terms is handled using a wave propagation algorithm. Two different methods for propagating waves are compared in computational tests. One method linearizes the equations about cell edges, and the other is based on the propagation of waves in a heterogeneous elastic medium. When the viscoelastic fluid is in contact with a Newtonian fluid, the method based on linearization about the edges gives more accurate results. No high Weissenberg number instabilities were observed. As long as the time step is chosen to obey a CFL condition, the algorithm is stable and no unphysical oscillations appear in the solution.

PACS: 47.11.−j; 47.11.Df; 47.63.−b; 83.60.Df

Keywords: Finite volume method; Oldroyd-B; Platelet aggregation; Wave propagation; High Weissenberg number

1. Introduction

The continuum model of platelet aggregation presented in [1,2] treats aggregating platelets as a dynamic network of immersed springs. The deformation of the network generates stresses which affect the motion of the fluid. The model contains two spatial scales: the spatial scale of the fluid flow, and the much smaller spatial scale of the platelet–platelet interactions. By assuming that variables only depend on averaged quantities, the dependence on the platelet scale can be eliminated. The fluid–platelet mixture is described by an equation for the stress that depends on the concentration of crosslinking platelets. A similar type of model has been used for polymeric flows, and under certain assumptions these multiscale models reduce to standard constitutive laws for viscoelastic fluids [3]. In Section 2, we show the platelet model is related to more familiar viscoelastic fluids.

The numerical algorithm used in [1,2] to simulate the platelet model involved alternating updates of the fluid velocity and the platelet stress. In some simulations with this algorithm, grid-scale oscillations appeared in some of the components of the stress. These oscillations did not grow significantly in time, so that the simulations remained stable. The oscillations could be eliminated by reducing the time step, however, it was not known *a priori* what time step to use.

In this paper, we present an algorithm based on a different splitting of the system. The advection and elastic terms are handled using a wave propagation algorithm [4], in which the fluid velocity and viscoelastic stresses are coupled.

☆ This work was supported in part by NSF grant #DMS-0139926.
* Corresponding author.
E-mail addresses: guy@math.ucdavis.edu (R.D. Guy), fogelson@math.utah.edu (A.L. Fogelson).

0045-7825/$ - see front matter © 2007 Elsevier B.V. All rights reserved.
doi:10.1016/j.cma.2007.11.022

The other terms (viscosity and relaxation) are discretized implicitly in time. The stability of this algorithm is governed by a CFL constraint. The splitting used in this paper was inspired by methods developed for high Reynolds number [5], in which high-resolution finite volume methods were used to handle the convection terms. Simulation results comparing the algorithm presented in this paper with the algorithm used in [1,2] show that no numerical oscillations appeared as long as the time step was chosen to satisfy the stability constraint. This means that a variable time step can be used, which makes the simulations much more efficient.

The algorithm was developed for simulating the platelet continuum model, but it can be applied to other models of viscoelastic fluids. The platelet model differs from other viscoelastic flow models in that the material parameters depend on the concentration of activated platelets. Viscoelastic fluids with high Weissenberg numbers are notoriously difficult to simulate [6]. In computational tests, our algorithm remained stable for all Weissenberg numbers, but refinement studies suggest that the results at high Weissenberg numbers may be inaccurate. In the platelet model high Weissenberg numbers are not encountered, so this does not pose a limitation on the algorithm's utility for simulating platelets.

2. Platelet continuum model as a viscoelastic fluid

In a companion paper [1] in this issue, a model of platelet aggregation is presented. It involves interactions among a viscous incompressible fluid, nonactivated and activated platelets, a chemical capable of activating platelets, and 'cohesive links' which can form between activated platelets. These links generate extra stresses on the fluid beyond the usual Newtonian pressure and viscous stress. The model equations are

$$\boldsymbol{u}_t + \boldsymbol{u}\cdot\nabla\boldsymbol{u} = -\nabla p + v\Delta\boldsymbol{u} + \nabla\cdot\boldsymbol{\sigma}^{\mathrm{p}}, \tag{1}$$

$$\nabla\cdot\boldsymbol{u} = 0, \tag{2}$$

$$(\phi_n)_t + \boldsymbol{u}\cdot\nabla\phi_n = D_n\Delta - R(c)\phi_n, \tag{3}$$

$$(\phi_a)_t + \boldsymbol{u}\cdot\nabla\phi_a = R(c)\phi_n, \tag{4}$$

$$c_t + \boldsymbol{u}\cdot\nabla c = D_c\Delta c + AR(c)\phi_n - Kc, \tag{5}$$

$$\boldsymbol{\sigma}^{\mathrm{p}}_t + \boldsymbol{u}\cdot\nabla\boldsymbol{\sigma}^{\mathrm{p}} = \boldsymbol{\sigma}^{\mathrm{p}}\nabla\boldsymbol{u} + \nabla\boldsymbol{u}^{\mathrm{T}}\boldsymbol{\sigma}^{\mathrm{p}} + \alpha_2\phi_a^2\boldsymbol{\delta} - \beta\boldsymbol{\sigma}^{\mathrm{p}}, \tag{6}$$

$$z^{\mathrm{p}}_t + \boldsymbol{u}\cdot\nabla z^{\mathrm{p}} = \alpha_0\phi_a^2 - \beta z^{\mathrm{p}}. \tag{7}$$

Note that for the velocity gradients, we use the convention that $(\nabla\boldsymbol{u})_{ij} = \partial u_j/\partial x_i$.

Eqs. (1) and (2) are the Navier–Stokes equations for the velocity $\boldsymbol{u}(\boldsymbol{x},t)$ and pressure $p(\boldsymbol{x},t)$ of a fluid with constant density and viscosity v. The 'cohesion-stress' tensor $\boldsymbol{\sigma}^{\mathrm{p}}(\boldsymbol{x},t)$ accounts for the stresses generated by interplatelet links. Eqs. (3)–(5) for the concentrations of nonactivated platelets $\phi_n(\boldsymbol{x},t)$, activated platelets $\phi_a(\boldsymbol{x},t)$, and activating chemical $c(\boldsymbol{x},t)$ describe their evolution because of transport and activation reactions. Eq. (6) describes the evolution of the cohesion-stress tensor and accounts for stresses generated at a rate $\alpha_2\phi_a^2$ because of the formation of new links between activated platelets, as well as a reduction in stress at rate $\beta\boldsymbol{\sigma}^{\mathrm{p}}$ because of the breaking of existing links. Eq. (7) describes the evolution of the concentration of links $z^{\mathrm{p}}(\boldsymbol{x},t)$ emanating from activated platelets at $\boldsymbol{x}$ as they move with the fluid, are created at rate $\alpha_0\phi_a^2$ and break at rate βz^{p}. Here we assume that this breaking rate is constant. See [1,2] for more discussion of the assumptions underlying the model and for examples of the model's behavior.

This continuum model of platelet aggregation can be described as a generalization of a multiscale model for polymer melts called *transient network theory* [7,8], which is similar to the theory of rubber elasticity. In some cases, the constitutive law for the fluid is identical to standard macroscopic descriptions of viscoelastic fluids [3]. The platelet model differs from these network theories in that the formation rate of links depends on the concentration of activated platelets, which is a function of space and time.

We introduce a change of variables in the stress to make this model resemble standard models of viscoelastic fluids. Define the stress tensor $\boldsymbol{\tau}$ by

$$\boldsymbol{\tau} = \boldsymbol{\sigma}^{\mathrm{p}} - G\boldsymbol{\delta}, \tag{8}$$

where G is proportional to the number of links:

$$G = \frac{\alpha_2}{\alpha_0}z^{\mathrm{p}}. \tag{9}$$

As we show below, this quantity G represents the elastic modulus of the material. By shifting the stress by an isotropic tensor, we are effectively changing the pressure. Since the fluid is incompressible, this does not change the resulting velocity field. This new stress satisfies the equation

$$\boldsymbol{\tau}_t + \boldsymbol{u}\cdot\nabla\boldsymbol{\tau} - \boldsymbol{\tau}\nabla\boldsymbol{u} - \nabla\boldsymbol{u}^{\mathrm{T}}\boldsymbol{\tau} - G(\nabla\boldsymbol{u} + \nabla\boldsymbol{u}^{\mathrm{T}}) = -\beta\boldsymbol{\tau}. \tag{10}$$

Dividing through by β and rearranging gives

$$\lambda(\boldsymbol{\tau}_t + \boldsymbol{u}\cdot\nabla\boldsymbol{\tau} - \boldsymbol{\tau}\nabla\boldsymbol{u} - \nabla\boldsymbol{u}^{\mathrm{T}}\boldsymbol{\tau}) + \boldsymbol{\tau} = 2\mu\boldsymbol{D}, \tag{11}$$

where the relaxation time, λ, and viscosity, μ, are defined by

$$\lambda = \beta^{-1}, \tag{12}$$

$$\mu = G\beta^{-1} \tag{13}$$

and where the deformation rate tensor $\boldsymbol{D}$ is

$$\boldsymbol{D} = \frac{1}{2}(\nabla\boldsymbol{u} + \nabla\boldsymbol{u}^{\mathrm{T}}). \tag{14}$$

Eq. (11) is the upper convected Maxwell (UCM) equation, except that the relaxation time and the viscosity are not material parameters, but functions of other variables in the model. Thus the continuum model of platelet aggregation can be thought of as a viscoelastic fluid with spatially and temporally varying material properties. Note that in the platelet model, the breaking rate, β, is also not constant, so that the constitutive law is more like the PTT model [9] which was generalized for this case in [10].

The UCM equation is typically written as in (11), but the form (10) is equally valid. The quantity z^{p} is proportional to the concentration of crosslinks between platelets.

Please cite this article in press as: R.D. Guy, A.L. Fogelson, A wave propagation algorithm for viscoelastic fluids with spatially ..., Comput. Methods Appl. Mech. Engrg. (2008), doi:10.1016/j.cma.2007.11.022

By relating this to the UCM equation, we see that we can reinterpret this quantity as a scaled elastic modulus of the fluid, G. For our purposes, it is more convenient to use the form (10) for which the material parameters are the elastic modulus, G, and the breaking rate, β.

For the remainder of this paper, we focus on numerical methods for the system of equations:

$$\boldsymbol{u}_t + \boldsymbol{u}\cdot\nabla\boldsymbol{u} = -\nabla p + \nu\Delta\boldsymbol{u} + \nabla\cdot\boldsymbol{\tau}, \tag{15}$$

$$\nabla\cdot\boldsymbol{u} = 0, \tag{16}$$

$$\boldsymbol{\tau}_t + \boldsymbol{u}\cdot\nabla\boldsymbol{\tau} - \boldsymbol{\tau}\nabla\boldsymbol{u} - \nabla\boldsymbol{u}^{\mathrm{T}}\boldsymbol{\tau} = 2G\boldsymbol{D} - \beta\boldsymbol{\tau}, \tag{17}$$

$$G_t + \boldsymbol{u}\cdot\nabla G = 0. \tag{18}$$

In the elastic modulus equation, we have assumed that the formation and breaking of links is in equilibrium. We do not explicitly nondimensionalize these equations because the number of parameters is not reduced. When nondimensionalized, the fluid viscosity, ν, becomes the inverse of the Reynolds number, the elastic modulus becomes the square of the elastic Mach number, and the breaking rate becomes the Weissenberg number. The Reynolds number is the ratio of inertial stress to viscous stress. The elastic Mach number is the ratio of elastic wave speed to convective flow speed, and the Weissenberg number is the ratio of the flow time scale to the relaxation time.

3. Numerical method

The method we present in this paper is inspired by methods developed for high Reynolds number flow based on high-resolution finite volume methods for conservation laws [5]. For viscoelastic fluids, the challenge in constructing robust numerical schemes is often not because of a high Reynolds number, but because of a high Weissenberg number or for strongly elastic flows. We use finite volume methods for a careful treatment of a hyperbolic subsystem consisting of the advection and elastic terms, rather than just the convection terms as in [5]. The system that arises is very similar to the that presented for wave propagation in a heterogeneous elastic medium [11].

3.1. Spatial discretization

In finite volume methods, the domain is discretized into volumes or grid cells, and functions are represented by their average value in each cell. We consider a uniform grid in two dimensions, pictured in Fig. 1. The discrete quantity q_{ij}^n represents

$$q_{ij}^n \approx \frac{1}{\Delta x\,\Delta y}\int_{y_{i-1/2}}^{y_{i+1/2}}\int_{x_{i-1/2}}^{x_{i+1/2}} q(x,y,t^n)\,\mathrm{d}x\,\mathrm{d}y. \tag{19}$$

Besides cell averages, the method makes use of quantities defined at cell edges. For example an approximation to q at the right edge of the cell indexed by (i,j) is labeled $q_{i+1/2,j}$. Although we only consider two spatial dimensions in this paper, all of the methods extend naturally to three dimensions.

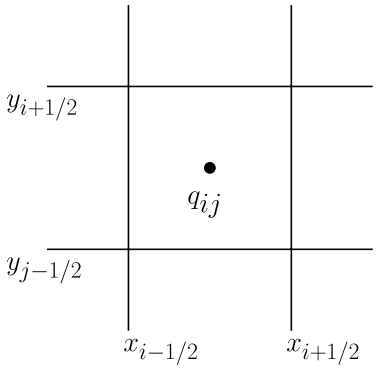

Fig. 1. Regular finite volume grid in two dimensions. The value of q_{ij} represents the average value of the function of the cell $[x_{i-1/2}, x_{i+1/2}] \times [y_{i-1/2}, y_{i+1/2}]$.

3.2. Navier–Stokes

Before presenting the algorithm for viscoelastic fluids, we first review solution of the Navier–Stokes equations. The time-dependent incompressible Navier–Stokes equations are

$$\boldsymbol{u}_t + \boldsymbol{u}\cdot\nabla\boldsymbol{u} = -\nabla p + \nu\Delta\boldsymbol{u} + \boldsymbol{f}, \tag{20}$$

$$\nabla\cdot\boldsymbol{u} = 0. \tag{21}$$

A common method for advancing the solution in time is a fractional step approach that breaks one time step into three substeps. A hyperbolic system is solved for the convective terms, a parabolic system accounts for the fluid viscosity, and the incompressibility constraint is taken into account by performing a projection.

For high Reynolds numbers the solution may contain sharp gradients, and it is imperative that the convection terms be discretized carefully. In [5], a high-resolution Godunov method is used to approximate the convection terms which gives a robust methods for all Reynolds numbers. The method is based on methods for conservation laws [12]. We briefly summarize the idea here.

The cell-centered velocities are extrapolated to the cell edges. Limited differences are used in approximating the spatial derivatives that arise in the extrapolation to prevent oscillations. This procedure generates two sets of edge values, one extrapolated from each side. The velocity at the edge is selected based on the solution to Riemann problems for Burger's equation

$$u_t + uu_x = 0. \tag{22}$$

Once the edge values have been determined, the convection terms are approximated by flux differencing in conservation form.

3.3. Viscoelastic flows

Simulating strongly elastic fluids presents similar challenges as simulating high Reynolds number flows in that care must be taken to discretize the hyperbolic terms. For

Please cite this article in press as: R.D. Guy, A.L. Fogelson, A wave propagation algorithm for viscoelastic fluids with spatially ..., Comput. Methods Appl. Mech. Engrg. (2008), doi:10.1016/j.cma.2007.11.022

viscoelastic flows the hyperbolic terms include both the convection and the elastic terms. The method described in the previous section was adapted to viscoelastic fluids using Lax–Wendroff in [13], and later using high-resolution Godunov methods in [14]. These methods used adaptive viscoelastic stress splitting similar to [15] to make the method more robust. For a comprehensive discussion on numerical methods for viscoelastic fluids, see [6]. The algorithm presented in this paper differs from exiting methods in that we consider fluids which are elastic in only a portion of the domain and the remainder of the domain is filled with Newtonian fluid.

We present an algorithm that is based on wave propagation methods [4] which generalize easily to nonconservative systems. As with Navier–Stokes, the method is a fractional step method that breaks the system of momentum equation, continuity equation, and viscoelastic constitutive equation into three parts: a hyperbolic system for the convective and elastic terms, a parabolic system for the fluid viscosity term, and an elliptic problem in which the incompressibility constraint is enforced by performing a projection.

Beginning with the velocity, stress, and pressure at the start of a time step, we advance a system that involves the convection and elastic terms:

$$\boldsymbol{u}_t + \boldsymbol{u}\cdot\nabla\boldsymbol{u} = \nabla\cdot\boldsymbol{\tau}, \tag{23}$$

$$\boldsymbol{\tau}_t + \boldsymbol{u}\cdot\nabla\boldsymbol{\tau} - \boldsymbol{\tau}\nabla\boldsymbol{u} - \nabla\boldsymbol{u}^{\mathrm{T}}\boldsymbol{\tau} = G(\nabla\boldsymbol{u} + \nabla\boldsymbol{u}^{\mathrm{T}}). \tag{24}$$

An explicit wave propagation method is used, and this is described in detail in Section 3.4. (For a nonconstant elastic modulus, the elastic modulus Eq. (18) should also be included in this system, but this does not change the eigendecomposition presented below.) This step gives an intermediate velocity field $\boldsymbol{u}^*$ and intermediate stress $\boldsymbol{\tau}^*$. Next the fluid viscosity, the stress relaxation, and external forcing terms are accounted for by evolving the system

$$\boldsymbol{u}^*_t = \nabla q + \nu\Delta\boldsymbol{u}^* + \boldsymbol{f}, \tag{25}$$

$$\boldsymbol{\tau}^*_t = -\beta\boldsymbol{\tau}^*. \tag{26}$$

The value q is an approximation to the fluid pressure and is held constant through the time step. This system is discretized implicitly in time. These two steps give an approximation to the velocity and stress at the end of the time step, but the velocity that is obtained, $\boldsymbol{u}^{**}$ is not divergence-free. To complete the time step, the intermediate velocity $\boldsymbol{u}^{**}$ is decomposed into a divergence-free field and a gradient field

$$\boldsymbol{u}^{**} = \boldsymbol{u} + \nabla\phi, \tag{27}$$

$$\nabla\cdot\boldsymbol{u} = 0. \tag{28}$$

This decomposition is performed by solving the Poisson equation

$$\Delta\phi = \nabla\cdot\boldsymbol{u}^{**}. \tag{29}$$

By splitting the system and solving one subsystem after the other, we obtain a method which is at best first-order accurate in time, regardless of the temporal accuracy of the discretizations of the subsystems. If desired, Strang splitting could be used to reduce the splitting error.

What is new about the method we propose in this paper is the use of a wave propagation algorithm for advancing the system (23) and (24). This system is of the form

$$\boldsymbol{q}_t + A(\boldsymbol{q})\boldsymbol{q}_x + B(\boldsymbol{q})\boldsymbol{q}_y = 0, \tag{30}$$

where $\boldsymbol{q} = (u, v, \tau_{11}, \tau_{12}, \tau_{22})$,

$$A = \begin{bmatrix} u & 0 & -1 & 0 & 0 \\ 0 & u & 0 & -1 & 0 \\ -2(\tau_{11}+G) & 0 & u & 0 & 0 \\ 0 & -(\tau_{11}+G) & 0 & u & 0 \\ 0 & -2\tau_{12} & 0 & 0 & u \end{bmatrix} \tag{31}$$

and

$$B = \begin{bmatrix} v & 0 & 0 & -1 & 0 \\ 0 & v & 0 & 0 & -1 \\ -2\tau_{12} & 0 & v & 0 & 0 \\ -(\tau_{22}+G) & 0 & 0 & v & 0 \\ 0 & -2(\tau_{22}+G) & 0 & 0 & v \end{bmatrix}. \tag{32}$$

Eq. (30) is hyperbolic if all of the eigenvalues of A and B are real. The eigenvalues of A are $u \pm \sqrt{2(\tau_{11}+G)}$, $u \pm \sqrt{(\tau_{11}+G)}$, and u. Similarly, the eigenvalues of B are $v \pm \sqrt{2(\tau_{22}+G)}$, $v \pm \sqrt{(\tau_{22}+G)}$, and v. Provided

$$G + \tau_{11} \geqslant 0, \tag{33}$$

$$G + \tau_{22} \geqslant 0, \tag{34}$$

all the eigenvalues are real and (30) is hyperbolic. The restrictions (33) and (34) always hold. This can be shown by expressing the upper convected Maxwell equation as an integral over past strains and using the fact that the strain is positive definite [16].

The high-resolution Godunov method used in [5] for the convection terms is based on methods developed for conservation laws. The system (23) and (24) is not conservative, and adapting these methods is not straightforward [14]. Wave propagation methods developed by LeVeque [4] were also developed for conservation laws, but they extend naturally to systems that are not in conservative form.

3.4. Wave propagation

We present the ideas of the wave propagation method for the one-dimensional linear problem

$$\boldsymbol{q}_t + A\boldsymbol{q}_x = 0, \tag{35}$$

where A is diagonalizable with real eigenvalues. We then extend these ideas to the multidimensional problem

$$\boldsymbol{q}_t + A\boldsymbol{q}_x + B\boldsymbol{q}_y = 0, \tag{36}$$

which is similar to Eq. (30), except that the matrices A and B are taken to be constant. We first treat the linear case not only for simplicity, but because the method we present for

Please cite this article in press as: R.D. Guy, A.L. Fogelson, A wave propagation algorithm for viscoelastic fluids with spatially ..., Comput. Methods Appl. Mech. Engrg. (2008), doi:10.1016/j.cma.2007.11.022

viscoelastic flows employs a linearization of (30). The methods described here are discussed more thoroughly in [4].

3.4.1. Upwinding in one dimension

We begin by defining some useful notation. Let R be a matrix of eigenvectors of A and Λ be a diagonal matrix with the eigenvalues of A on the diagonal, so that

$$A = R\Lambda R^{-1}. \tag{37}$$

Define the matrices Λ^+ and Λ^- by

$$\Lambda^+ = \frac{\Lambda + |\Lambda|}{2}, \tag{38}$$

$$\Lambda^- = \frac{\Lambda - |\Lambda|}{2}. \tag{39}$$

In Λ^+, the positive eigenvalues are left alone and the the negative eigenvalues are replaced by zero; similarly for Λ^-. Define A^+, A^-, and $|A|$ by

$$A^+ = R\Lambda^+ R^{-1}, \tag{40}$$

$$A^- = R\Lambda^- R^{-1}, \tag{41}$$

$$|A| = A^+ - A^-. \tag{42}$$

The simplest method for discretizing (35) is based on the upwind method for scalar equations. By changing into coordinates of the eigenspaces of A, (35) is diagonalized, and the problem reduces to scalar advection equations on each eigenspace. Letting $\hat{\boldsymbol{q}} = R^{-1}\boldsymbol{q}$, and noting that $\Lambda = \Lambda^+ + \Lambda^-$, we see that $\hat{\boldsymbol{q}}$ satisfies the equation

$$\hat{\boldsymbol{q}}_t + \Lambda^+\hat{\boldsymbol{q}}_x + \Lambda^-\hat{\boldsymbol{q}}_x = 0. \tag{43}$$

This equation can be easily discretized using the upwind method as

$$\hat{\boldsymbol{q}}_j^{n+1} = \hat{\boldsymbol{q}}_j^n - \frac{\Delta t}{\Delta x}\Lambda^+\left(\hat{\boldsymbol{q}}_j^n - \hat{\boldsymbol{q}}_{j-1}^n\right) - \frac{\Delta t}{\Delta x}\Lambda^-\left(\hat{\boldsymbol{q}}_{j+1}^n - \hat{\boldsymbol{q}}_j^n\right). \tag{44}$$

Changing back to the original variable $\boldsymbol{q}$, this discretization is

$$\boldsymbol{q}_j^{n+1} = \boldsymbol{q}_j^n - \frac{\Delta t}{\Delta x}A^+\left(\boldsymbol{q}_j^n - \boldsymbol{q}_{j-1}^n\right) - \frac{\Delta t}{\Delta x}A^-\left(\boldsymbol{q}_{j+1}^n - \boldsymbol{q}_j^n\right). \tag{45}$$

The discretization in the original variable is more general, because it can be used when A is spatially dependent.

The discretization (45) has another more general interpretation in terms of wave propagation. Recall that the discrete variable $\boldsymbol{q}_j^n$ represents the average value of the function $\boldsymbol{q}(x,t_n)$ over cell j. There are many functions $\boldsymbol{q}(x,t_n)$ that when averaged will give the discrete values $\boldsymbol{q}_j^n$. The simplest is the piecewise constant function which has the value $\boldsymbol{q}_j^n$ over cell j. At each cell interface this function is discontinuous. If we take this piecewise constant function and evolve it according to the original equation (35), the discontinuities propagate as waves into the neighboring cells. The wave speeds are given by the eigenvalues of A. These waves propagate as jumps in $\boldsymbol{q}$ proportional to the corresponding eigenvectors. In (45), the term $A^+(\boldsymbol{q}_j^n - \boldsymbol{q}_{j-1}^n)$ is proportional to the amount that the cell average is changed due to right-moving waves originating from the left edge, and similarly, the term $A^-(\boldsymbol{q}_{j+1}^n - \boldsymbol{q}_j^n)$ is proportional to the change due to left-moving waves originating from the right edge. This idea is made more explicit by writing

$$A^+\left(\boldsymbol{q}_j^n - \boldsymbol{q}_{j-1}^n\right) = \sum_{k:\lambda^{(k)}>0}\lambda^{(k)}\alpha^{(k)}\boldsymbol{r}^{(k)} = \sum_{k:\lambda^{(k)}>0}\lambda^{(k)}\boldsymbol{W}_{j-1/2}^{(k)}, \tag{46}$$

where $\lambda^{(k)}$ is the kth eigenvalue of A, and $\boldsymbol{W}_{j-1/2}^{(k)} = \alpha^{(k)}\boldsymbol{r}^{(k)}$ is the component of $\boldsymbol{q}_j^n - \boldsymbol{q}_{j-1}^n$ in the kth eigenspace. Similarly for the right edge

$$A^-\left(\boldsymbol{q}_{j+1}^n - \boldsymbol{q}_j^n\right) = \sum_{k:\lambda^{(k)}<0}\lambda^{(k)}\alpha^{(k)}\boldsymbol{r}^{(k)} = \sum_{k:\lambda^{(k)}<0}\lambda^{(k)}\boldsymbol{W}_{j+1/2}^{(k)}. \tag{47}$$

In these terms, (45) becomes

$$\boldsymbol{q}_j^{n+1} = \boldsymbol{q}_j^n - \frac{\Delta t}{\Delta x}\left(\sum_{k:\lambda^{(k)}>0}\lambda^{(k)}\boldsymbol{W}_{j-1/2}^{(k)} + \sum_{k:\lambda^{(k)}<0}\lambda^{(k)}\boldsymbol{W}_{j+1/2}^{(k)}\right). \tag{48}$$

This wave propagation representation is more general than (45) since it can be applied to linear and nonlinear problems. To apply the method we must know the wave structure that results from discontinuities at cell edges.

3.4.2. Second-order corrections

The methods described above were originally developed for conservation laws and are often described in this way. Up to this point we have purposely avoided this perspective because the viscoelastic flow problem is not a conservation law. However, for moving from first-order upwinding to a second-order method this perspective is helpful. The first-order method presented above is described in terms of piecewise constant waves propagating from interfaces. Note that these waves only change the value in the cell into which they move. One can interpret second-order corrections as the propagation of piecewise linear waves. These corrections modify both the cell into which they move and the cell from which they originated. Because of this it is useful to think of including them by adding corrective fluxes at the edges of the cells.

Eq. (35) can be written in conservation form as

$$\boldsymbol{q}_t + (A\boldsymbol{q})_x = 0. \tag{49}$$

Numerical methods for conservation laws are typically of the form

$$\boldsymbol{q}_j^{n+1} = \boldsymbol{q}_j^n - \frac{\Delta t}{\Delta x}(\boldsymbol{F}_{j+1/2} - \boldsymbol{F}_{j-1/2}), \tag{50}$$

where $\boldsymbol{F}_{j-1/2}$ represents the numerical flux function between cell $j-1$ and cell j. The numerical flux function for the upwind method is

$$\boldsymbol{F}_{j-1/2}^{\text{upwind}} = A^+\boldsymbol{q}_{j-1} + A^-\boldsymbol{q}_j. \tag{51}$$

Using this flux in (50) gives (45).

A familiar second-order method is the Lax–Wendroff method. It is derived by using finite differences to match

Please cite this article in press as: R.D. Guy, A.L. Fogelson, A wave propagation algorithm for viscoelastic fluids with spatially ..., Comput. Methods Appl. Mech. Engrg. (2008), doi:10.1016/j.cma.2007.11.022

the terms of a Taylor series up to second-order. The method can also be described as a finite volume method. The numerical flux function for Lax–Wendroff is

$$\boldsymbol{F}^{\mathrm{LW}}_{j-1/2} = (A^+\boldsymbol{q}_{j-1} + A^-\boldsymbol{q}_j) + \frac{1}{2}|A|\left(I - \frac{\Delta t}{\Delta x}|A|\right)(\boldsymbol{q}_j - \boldsymbol{q}_{j-1}), \tag{52}$$

which looks like the upwind flux plus a second-order correction. The second-order correction is in terms of the difference in $\boldsymbol{q}$ across the cell edge. In presenting the upwind method as a wave propagation method, this very quantity was decomposed into the eigenvectors of A. The correction term to the flux can thus be interpreted as a second-order correction to the waves propagating from the interface:

$$\frac{1}{2}|A|\left(I - \frac{\Delta t}{\Delta x}|A|\right)(\boldsymbol{q}_j - \boldsymbol{q}_{j-1}) = \sum_k \frac{1}{2}|\lambda^{(k)}|\left(I - \frac{\Delta t}{\Delta x}|\lambda^{(k)}|\right)\boldsymbol{W}^{(k)}_{j-1/2}. \tag{53}$$

Written this way, the flux correction terms appear as correction waves that affect the values in the cells on both sides of the edge.

Combining the method from Section 3.4.1 with the second-order corrections, we obtain the second-order wave propagation method

$$\boldsymbol{q}^{n+1}_j = \boldsymbol{q}^n_j - \frac{\Delta t}{\Delta x}\left(\sum_{k:\lambda^{(k)}>0}\lambda^{(k)}\boldsymbol{W}^{(k)}_{j-1/2} + \sum_{k:\lambda^{(k)}<0}\lambda^{(k)}\boldsymbol{W}^{(k)}_{j+1/2}\right) - \frac{\Delta t}{\Delta x}(\boldsymbol{F}_{j+1/2} - \boldsymbol{F}_{j-1/2}), \tag{54}$$

$$\boldsymbol{F}_{j-1/2} = \sum_k \frac{1}{2}|\lambda^{(k)}|\left(I - \frac{\Delta t}{\Delta x}|\lambda^{(k)}|\right)\boldsymbol{W}^{(k)}_{j-1/2}. \tag{55}$$

3.4.3. Stability

For the wave propagation method to be stable, the time step must satisfy the CFL condition:

$$\Delta t \leqslant \frac{\Delta x}{\max|\lambda^{(k)}|}. \tag{56}$$

Conceptually this means that during one time step waves cannot propagate farther than one grid cell.

3.4.4. High-resolution

The upwinding method is diffusive. When it is applied to problems with sharp interfaces, it causes the interface to be progressively smeared. Adding the second-order corrections eliminates the numerical diffusion, but at a cost. Near sharp gradients in the solution numerical oscillations can develop. The idea behind *high-resolution* methods is to exploit the advantages of both first-order and second-order methods. Where the solution is smooth the method should be second-order, but near sharp gradients the order is reduced to prevent numerical oscillations.

For conservation laws we can replace the Lax–Wendroff flux, (52), by one of the form

$$\boldsymbol{F} = \boldsymbol{F}^{\mathrm{upwind}} + \Phi(\boldsymbol{F}^{\mathrm{LW}} - \boldsymbol{F}^{\mathrm{upwind}}), \tag{57}$$

where Φ represents a limiter function. When $\Phi = 0$ the method reduces to upwinding, and when $\Phi = 1$ this gives Lax–Wendroff. For the wave propagation algorithm, the limiter appears in the second-order correction waves. For more details on different choices of limiter functions and their application in wave propagation, see [4].

3.4.5. Accuracy

For constant coefficient linear problems, the method without limiters is second-order in space and time. With limiters the accuracy is reduced in regions with sharp gradients in the solution. In fact, even without limiters the method is only first-order accurate for variable coefficient problems.

One may wonder why we bother to include the second-order corrections if we can only achieve a first-order accurate method. Upwinding produces large numerical diffusion, which causes the solution profile to be smeared. Adding the second-order terms reduces this smearing. Even though limiters reduce the formal order of accuracy, they produce better resolved solutions, which is more important than the rate of convergence.

3.4.6. Multiple dimensions, transverse waves

Extending the wave propagation algorithm to multiple dimensions is straightforward. Consider the two-dimensional problem

$$\boldsymbol{q}_t + A\boldsymbol{q}_x + B\boldsymbol{q}_y = 0. \tag{58}$$

One could simply apply the one-dimensional algorithm in each direction. This approach is easy, but it has three drawbacks. First, this splitting does not include all the second-order terms so that the resulting scheme is diffusive. Second, it will fail to accurately capture waves that propagate diagonal to the grid. Third the time step required for stability is more restrictive than an unsplit method. These problems are alleviated by including transverse propagation.

The first step of the algorithm proceeds as in one dimension and accounts for the waves normal to each interface. These so called increment waves are then decomposed into waves moving in the transverse direction. This is illustrated in Fig. 2. These transverse waves affect the value in the cell from which they originate as well as the cell into which they are traveling. Thus, as with the one-dimensional second-order corrections, they are included in a flux term. For example the cell indexed by i, j is changed by waves moving across its left edge by an amount proportional to

$$\sum_{k:\lambda^{(k)}>0}\lambda^{(k)}\boldsymbol{W}^{(k)}_{i-1/2,j} = A^+\left(\boldsymbol{q}^n_{i,j} - \boldsymbol{q}^n_{i,j-1}\right). \tag{59}$$

Please cite this article in press as: R.D. Guy, A.L. Fogelson, A wave propagation algorithm for viscoelastic fluids with spatially ..., Comput. Methods Appl. Mech. Engrg. (2008), doi:10.1016/j.cma.2007.11.022

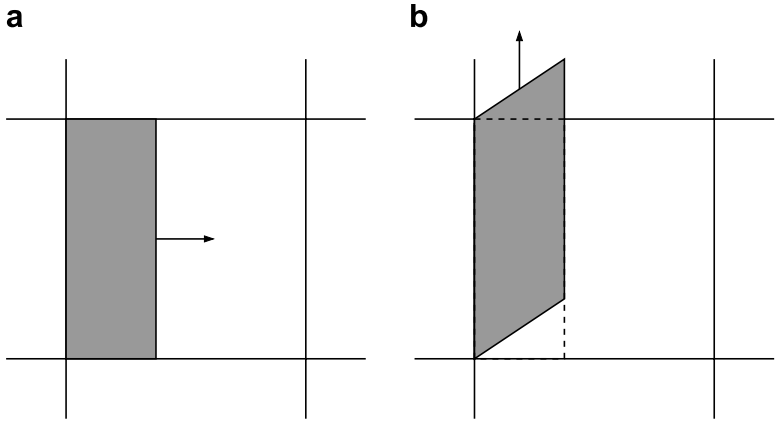

Fig. 2. (a) Normal propagation of a wave across the left edge of a cell. (b) Transverse propagation of normal wave increment affects both the cell from which the wave originates and the cell to which it is traveling.

To account for transverse propagation, this quantity is decomposed into waves across the top interface and the flux function at that interface is updated by

$$G_{i,j+1/2} := G_{i,j+1/2} + \frac{\Delta t}{2\Delta x} B^+ A^+ \left(\boldsymbol{q}^n_{i,j} - \boldsymbol{q}^n_{i,j-1} \right). \tag{60}$$

Similarly, the same amount is decomposed into waves across the bottom interface and the flux function is updated by

$$G_{i,j-1/2} := G_{i,j-1/2} + \frac{\Delta t}{2\Delta x} B^- A^+ \left(\boldsymbol{q}^n_{i,j} - \boldsymbol{q}^n_{i,j-1} \right). \tag{61}$$

For linear systems multiplication by B^+ or B^- is equivalent to decomposing a vector into its upward or downward moving components. However, in practice, these matrices are never formed, but are used here for simplicity of notation. In the algorithm, a wave decomposition is performed to find the upward and downward moving components. This is more general and can be done for nonlinear systems.

4. Wave propagation for viscoelastic fluids

In Section 3.3, we described how to split Eqs. (15)–(18) to obtain a hyperbolic subsystem containing the advection and elastic terms, Eqs. (23) and (24). The difficult part in applying the wave propagation algorithm to this system is performing the wave decompositions. For conservation laws, this is achieved by solving Riemann problems (or approximate Riemann problems). Since our system is not a conservation law, it is not evident how to perform the wave decompositions. We compare two approaches. The first is based on linearizing about the edges, and the second is based on wave propagation for heterogeneous elastic media [11].

Consider first the linearization about the edges. The idea is to choose edge values $\boldsymbol{q}^{\text{edge}} = (u^{\text{e}}, v^{\text{e}}, \tau^{\text{e}}_{11}, \tau^{\text{e}}_{12}, \tau^{\text{e}}_{22})$ at every edge, and perform the decomposition into waves according to the eigenspaces of $A(\boldsymbol{q}^{\text{edge}})$ and $B(\boldsymbol{q}^{\text{edge}})$. For any reasonable choice of edge values, we find that this algorithm is *unstable*, particularly in the case when the viscoelastic fluid is confined to a subset of the domain and the remainder is filled with Newtonian fluid (i.e. $G = 0$ in parts of the domain).

To understand why, consider the increment waves propagating in the x-direction (across a vertical edge). We compute a biorthogonal set of right eigenvectors, $\boldsymbol{r}^{(k)}$, and left eigenvectors, $\boldsymbol{\ell}^{(k)}$, for the matrix $A(\boldsymbol{q}^{\text{edge}})$. These vectors satisfy

$$A\boldsymbol{r}^{(k)} = \lambda^{(k)} \boldsymbol{r}^{(k)}, \tag{62}$$

$$A^{\text{T}} \boldsymbol{\ell}^{(k)} = \lambda^{(k)} \boldsymbol{\ell}^{(k)}, \tag{63}$$

$$\boldsymbol{\ell}^{(i)} \cdot \boldsymbol{r}^{(j)} = \delta_{ij}, \tag{64}$$

where δ_{ij} is the Kronecker delta. The jumps across the interface are then decomposed into waves as

$$\boldsymbol{W}^{(k)} = \alpha^{(k)} \boldsymbol{r}^{(k)} = (\boldsymbol{\ell}^{(k)} \cdot \Delta \boldsymbol{q}) \boldsymbol{r}^{(j)}, \tag{65}$$

where $\Delta \boldsymbol{q}$ is the difference in $\boldsymbol{q}$ across the edge. These waves change the value in the cell into which they propagate by the amount

$$-\frac{\Delta t}{\Delta x} \lambda^{(k)} (\boldsymbol{\ell}^{(k)} \cdot \Delta \boldsymbol{q}) \boldsymbol{r}^{(j)} = -\frac{\Delta t}{\Delta x} \lambda^{(k)} \boldsymbol{r}^{(j)} (\boldsymbol{\ell}^{(k)})^{\text{T}} \Delta \boldsymbol{q}. \tag{66}$$

This simple rearrangement of the inner product suggests another way to think about this update – a series of rank one matrices is applied to the jump in the function across the edge.

To see why the wave propagation method is unstable, consider the eigenvectors corresponding to the eigenvalue $\lambda = u + \sqrt{2(\tau_{11} + G)}$:

$$\boldsymbol{r} = (1, 0, -\sqrt{2(\tau_{11} + G)}, 0, 0)^{\text{T}}, \tag{67}$$

$$\boldsymbol{\ell} = \left(\frac{1}{2}, 0, -\frac{1}{2\sqrt{2(\tau_{11} + G)}}, 0, 0 \right)^{\text{T}}. \tag{68}$$

The $1,3$ element of the matrix $\lambda \boldsymbol{r} \boldsymbol{\ell}^{\text{T}}$ is

$$[\lambda \boldsymbol{r} \boldsymbol{\ell}^{\text{T}}]_{1,3} = -\frac{u + \sqrt{2(\tau_{11} + G)}}{2\sqrt{2(\tau_{11} + G)}}. \tag{69}$$

Please cite this article in press as: R.D. Guy, A.L. Fogelson, A wave propagation algorithm for viscoelastic fluids with spatially ..., Comput. Methods Appl. Mech. Engrg. (2008), doi:10.1016/j.cma.2007.11.022

This term becomes arbitrarily large when $\tau_{11} + G$ is very small. This problem shows up immediately when the viscoelastic fluid is immersed in an Newtonian fluid. Near the edges of the interface between the two fluids, $\tau_{11} + G$ is necessarily small. In simulations of a uniform viscoelastic fluid, this instability shows up in regions of large strain rates which is reminiscent of high Weissenberg instabilities.

This is not the only potentially unbounded term encountered. Each of the five waves contains a similar term. Notice that if $u = 0$, (69) reduces to 1/2. The other unbounded terms also become bounded when the velocity is set to zero. This suggests a solution to this problem: a further splitting. In the matrices A and B given by (31) and (32), respectively, the velocities appear only on the diagonal and arise from the advection terms. If we do not include the advection terms with the elastic terms, then it is as if the velocity has been set to zero in (31) and (32), and all of the waves across edges become nicely bounded.

To stabilize the wave propagation, we split (23) and (24) into a subsystem containing the advection terms

$$\boldsymbol{u}_t + \boldsymbol{u} \cdot \nabla \boldsymbol{u} = 0, \tag{70}$$

$$\boldsymbol{\tau}_t + \boldsymbol{u} \cdot \nabla \boldsymbol{\tau} = 0 \tag{71}$$

and a subsystem containing the elastic terms

$$\boldsymbol{u}_t = \nabla \cdot \boldsymbol{\tau}, \tag{72}$$

$$\boldsymbol{\tau}_t - \boldsymbol{\tau}\nabla \boldsymbol{u} - \nabla \boldsymbol{u}^{\mathrm{T}}\boldsymbol{\tau} = G(\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^{\mathrm{T}}). \tag{73}$$

4.1. Advection

For the advection subsystem, we apply the wave propagation method as described in [17]. Applying this method for advection requires knowing the velocity normal to the edges. Hence, we maintain two sets of velocities: one velocity field is stored at the cell centers and the other is stored in a staggered fashion at the cell edges. Below we describe how the edge velocities are computed as part of the projection.

Before performing the projection, we have the intermediate velocity $\boldsymbol{u}^*$ at the cell centers. We compute the velocity normal to the cell edges by averaging:

$$u^{*\mathrm{e}}_{i+1/2,j} = \frac{u^*_{i,j} + u^*_{i+1,j}}{2}, \quad v^{*\mathrm{e}}_{i,j+1/2} = \frac{u^*_{i,j} + u^*_{i,j+1}}{2}. \tag{74}$$

These edge velocities are then projected onto the space of divergence-free fields by solving

$$\Delta \phi = \nabla \cdot \boldsymbol{u}^{*\mathrm{e}} \tag{75}$$

and updating the edge velocities

$$u^{\mathrm{e}}_{i+1/2,j} = u^{*\mathrm{e}}_{i+1/2,j} - \frac{\phi_{i+1,j} - \phi_{i,j}}{\Delta x}, \tag{76}$$

$$v^{\mathrm{e}}_{i,j+1/2} = v^{*\mathrm{e}}_{i,j+1/2} - \frac{\phi_{i,j+1} - \phi_{i,j}}{\Delta x}. \tag{77}$$

The original cell-centered velocities are then corrected by

$$u_{i,j} = u^*_{i,j} - \frac{\phi_{i+1,j} - \phi_{i-1,j}}{2\Delta x}, \tag{78}$$

$$v_{i,j} = v^*_{i,j} - \frac{\phi_{i,j+1} - \phi_{i,j-1}}{2\Delta x}. \tag{79}$$

This procedure ensures that the edge velocities are discretely divergence-free, but the cell-centered velocities are only approximately discretely divergence-free, so that we are using an approximate projection [18] for the cell-centered velocities.

4.2. Elastic wave propagation

Without the advection terms the system (72) and (73) is still of the form (30). The matrices $A(\boldsymbol{q})$ and $B(\boldsymbol{q})$ are given by (31) and (32), respectively, with the velocities along the diagonal set to zero. We discuss two variations of the wave propagation algorithm that differ in how the wave decomposition is performed. The first is based on linearizing (30) about the edges by averaging the variables on the two sides to the edge. This method is referred to as *method I*. The second method is based on wave propagation for heterogeneous elastic media [11], and we refer to it as *method II*.

4.2.1. Method I – Averaging at edges

To linearize about the edges we need values of τ_{11}, τ_{12}, and G at the vertical edges because these quantities appear in A. Similarly we need values of τ_{22}, τ_{12}, and G at the horizontal edges because these quantities appear in B. We compute these edge values by simple averaging:

$$(\tau^{\mathrm{e}}_{11})_{i-1/2,j} = \frac{(\tau_{11})_{i-1,j} + (\tau_{11})_{i,j}}{2}, \tag{80}$$

$$(\tau^{\mathrm{e}}_{12})_{i-1/2,j} = \frac{(\tau_{12})_{i-1,j} + (\tau_{12})_{i,j}}{2}, \tag{81}$$

$$G^{\mathrm{e}}_{i-1/2,j} = \frac{G_{i-1,j} + G_{i,j}}{2}. \tag{82}$$

With the edge values computed, the algorithm precedes as described for the linear problem in Section 3.4. The only difference is that the eigenvectors and eigenvalues used in the wave decomposition are different at each edge.

4.2.2. Method II – Heterogeneous elastic material

The second method we explore is based on [11], which describes a method for linearly elastic material with a spatially dependent elastic modulus. Consider a vertical edge between two cells in which the values of the elastic modulus are different. There are two waves moving the to right with speeds $\sqrt{2(\tau_{11} + G)}$ and $\sqrt{\tau_{11} + G}$, and two waves moving to the left with speeds $-\sqrt{2(\tau_{11} + G)}$ and $-\sqrt{\tau_{11} + G}$. All of these quantities are known only at the cell centers. The idea from [11] is that the right-going waves propagate according to the speed computed from G and τ_{11} on the right of the cell edge, and the left-moving waves propagate

Please cite this article in press as: R.D. Guy, A.L. Fogelson, A wave propagation algorithm for viscoelastic fluids with spatially ..., Comput. Methods Appl. Mech. Engrg. (2008), doi:10.1016/j.cma.2007.11.022

according to the speed computed with the values on the left side of the cell edge.

For the linearized problem the waves are computed by decomposing the jump across the cell edge into the eigenspaces of the matrix at the edge, and the waves are then of the form $\boldsymbol{W}^{(k)} = \alpha^{(k)}\boldsymbol{r}^{(k)}$. Performing this decomposition is equivalent to solving the linear system

$$R\boldsymbol{\alpha} = \Delta\boldsymbol{q}. \tag{83}$$

To perform the decomposition for the heterogeneous material, a different system must be solved. Let $\boldsymbol{r}_+^{(1)}$ and $\boldsymbol{r}_+^{(2)}$ denote the eigenvectors on the right side of the edge corresponding to the two positive eigenvalues, and let $\boldsymbol{r}_-^{(3)}$ and $\boldsymbol{r}_-^{(4)}$ denote the eigenvectors on the left side of the edge corresponding to the negative eigenvalues. There is one more eigenvector with eigenvalue zero. This eigenvector is $\boldsymbol{r}^{(5)} = (0,0,0,0,1)^{\mathrm{T}}$, which does not contain any state variables in its elements. To carry out the decomposition, we solve a system of the form (83), with R replaced by a matrix with columns $\boldsymbol{r}_+^{(1)}$, $\boldsymbol{r}_+^{(2)}$, $\boldsymbol{r}_-^{(3)}$, $\boldsymbol{r}_-^{(4)}$, and $\boldsymbol{r}^{(5)}$.

There are two important differences between the system considered in [11] and (72) and (73). In [11], the matrices A and B did not depend on the values of the stresses themselves, and secondly the elastic modulus was a given function of space. For the viscoelastic fluid problem, the waves and speeds depend on the stress and the elastic modulus moves with the fluid.

4.3. Viscoelastic fluids in contact with Newtonian fluids

For simulations of a viscoelastic fluid in contact with a Newtonian fluid, we must make several modifications to the algorithm. Consider a numerical cell edge between a viscoelastic fluid and a Newtonian fluid. Using method I as described above, we average the stress and elastic modulus to this edge and propagate waves based on these average values. This procedure causes viscoelastic stresses to develop in cells of Newtonian fluid. To fix this problem, we modify method I at such edges so that waves are propagated as in method II. This prevents elastic stresses from developing in regions where the elastic modulus is zero. We have found that the simulations are much more robust with this modification to method I.

The second modification applies to both methods I and II. In the wave decomposition, terms of the form

$$\frac{\tau_{12}}{\sqrt{\tau_{11}+G}} \tag{84}$$

arise. These terms look as though they have the potential to become unbounded. However, the tensor $\boldsymbol{\tau} + G\boldsymbol{\delta}$ is positive semidefinite. This can be shown by the same argument used to show that (33) and (34) hold. Because $\boldsymbol{\tau} + G\boldsymbol{\delta}$ is positive semidefinite, the determinant is nonnegative, and so

$$\left|\frac{\tau_{12}}{\sqrt{\tau_{11}+G}}\right| \leqslant \sqrt{\tau_{22}+G}. \tag{85}$$

The algorithm presented in this paper does not necessarily preserve the definiteness of the stress, which can lead to instabilities. At the beginning of the time step, we look for grid cells for which $\boldsymbol{\tau} + G\boldsymbol{\delta}$ fails to be positive semidefinite. In these grid cells we reset $\boldsymbol{\tau}$ so that $\boldsymbol{\tau} + G\boldsymbol{\delta}$ is the closest (in two-norm) positive semidefinite tensor. In Appendix A, we explain how this is done.

The wave propagation algorithm does not preserve hyperbolicity, meaning that after propagating waves, the inequalities (33) and (34) may not hold everywhere. When these inequalities are violated, the discrepancy is very small. By enforcing that $\boldsymbol{\tau} + G\boldsymbol{\delta}$ be positive semidefinite as described above, the inequalities (33) and (34) are satisfied and hyperbolicity is maintained.

4.4. Summary of the algorithm

At the beginning of a time step we have the velocity field, pressure, viscoelastic stress, and elastic modulus at the cell centers and a second velocity field at the cell edges. One step of the algorithm is outlined below:

- Advect all cell-centered quantities using the wave propagation algorithm described in [17].
- Ensure that (72) and (73) is hyperbolic and that the stress tensor $\boldsymbol{\tau} + G\boldsymbol{\delta}$ is positive semidefinite, as described in Section 4.3.
- Average the stresses to the cell edges (method I only).
- Propagate elastic waves in the system (72) and (73).
- Update velocity with the viscous terms and background force implicitly.
- Relax the stress implicitly.
- Project the velocity to enforce incompressibility.

5. Computational tests

In this section, we present a series of computational tests. One goal in performing these tests is to determine whether one of the two wave propagation methods offers any advantages over the other. For the first set of tests, we consider a single viscoelastic fluid with a uniform elastic modulus, and then we explore problems which contain both Newtonian and viscoelastic fluids.

5.1. Uniform elastic modulus

The domain is the periodic box $[-0.5, 0.5] \times [-0.5, 0.5]$. The flow is driven by a body force which is chosen to drive a Newtonian fluid with a steady state velocity of

$$u = \sin(2\pi x)\cos(2\pi y), \quad v = -\cos(2\pi x)\sin(2\pi y). \tag{86}$$

Substituting this velocity field into the Navier–Stokes equations, we compute a force field $\boldsymbol{f}_{\mathrm{ss}}$. The force which we apply to the fluid in these tests is

$$\boldsymbol{f}_{\mathrm{bg}} = (1 - \mathrm{e}^{-t})\boldsymbol{f}_{\mathrm{ss}}. \tag{87}$$

Please cite this article in press as: R.D. Guy, A.L. Fogelson, A wave propagation algorithm for viscoelastic fluids with spatially ..., Comput. Methods Appl. Mech. Engrg. (2008), doi:10.1016/j.cma.2007.11.022

When the fluid is viscoelastic, the flow pattern is very similar to the Newtonian case, but the velocities are generally smaller. We fixed the Reynolds number at 25, and explored a wide range of elastic moduli (10^{-3}–10^{3}) and relaxation rates (10^{-3}–10^{3}). The simulations remained stable for all sets of parameters tested. We present refinement studies for two sets of parameters which typify the two behaviors observed.

For the first test the elastic modulus is $G = 1$, the relaxation rate is $\beta = 1$, and the simulation is run until time $t = 1$. The flow field and the stresses are displayed in Fig. 3. The maximum velocity is about 0.05, which is the effective Weissenberg number.

The results of the refinement study are displayed in Fig. 4. The results for v and τ_{22} are not displayed because they are identical to the results for u and τ_{11}. The one-norm and the max-norm for both methods are displayed. The convergence is first-order for all variables and in both norms. The errors in the velocity were essentially the same for the two methods. The errors in the stresses were slightly smaller when waves were propagated as in a heterogeneous material, although the errors were similar for the two methods.

The results from the previous test do not demonstrate a significant difference between the two methods. However, when the elastic modulus and the breaking rate were small the algorithms gave very different results when run for long periods of time. The velocity field and stresses generated by method I for $G = 0.01$ and $\beta = 0.01$ are shown in Fig. 5 at time $t = 10$. This simulation was run longer than the previous one because it took longer to see the effect of the slow relaxation rate. The maximum velocity is again around 0.05, making the effective Weissenberg number 5, which is considered high. Notice that compared to the previous test, the stresses are much larger and the normal stress, τ_{11}, is concentrated in a very narrow region of space.

The results from a refinement study for this parameter set are displayed in Fig. 6. For method II (propagation in a heterogeneous material), none of the variables appear to be converging. When waves are propagated by averaging values to the edges (method I), the velocity appears to be converging, although at a rate less that first-order (0.68 in one-norm and 0.53 in max-norm). However, its not clear that the stresses are converging. From Fig. 5b, we see that the normal stress is highly concentrated at a few grid points. The lack of convergence could be due to severe under-resolution of the stresses.

The numerical solutions from the two methods of wave propagation do not converge to each other. The max-norm of the difference in the velocities is shown in Fig. 7. Thus this test shows that there is a significant difference between the two methods at high Weissenberg number. The averaging method (method I) is converging in the velocity, but it is not clear that it is converging to the correct velocity. To determine if this method is approaching the correct velocity, we must have an analytic solution. In the next section, we use a test for which we know the solution.

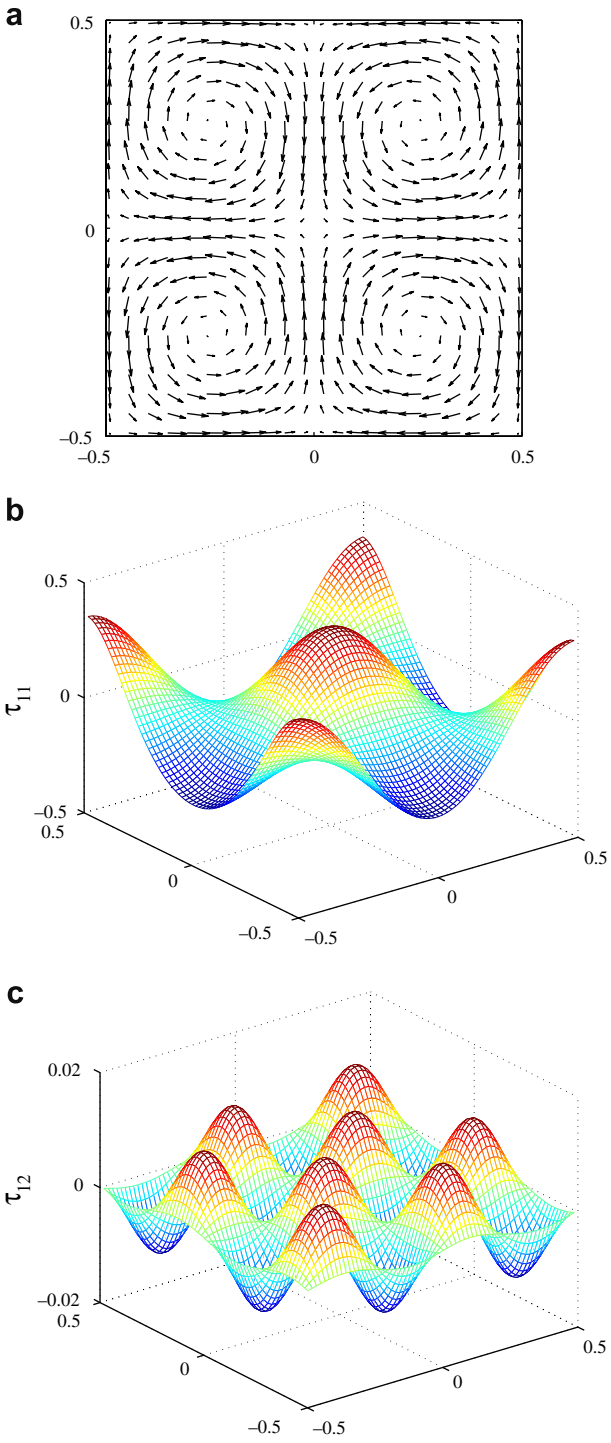

Fig. 3. The (a) velocity field and (b,c) stresses are shown for $G = 1$ and $\beta = 1$ at time $t = 1$. The stress component τ_{22} is not displayed because it is similar to τ_{11} with an appropriate shift and rotation.

5.2. Layered fluid shear flow

To explore further the difference between the two algorithms, we use a simple test problem for which we can generate an analytic solution: steady state, one-dimensional shear flow. When the elastic modulus was uniform, the methods converged at the same rate and the solutions

Please cite this article in press as: R.D. Guy, A.L. Fogelson, A wave propagation algorithm for viscoelastic fluids with spatially ..., Comput. Methods Appl. Mech. Engrg. (2008), doi:10.1016/j.cma.2007.11.022

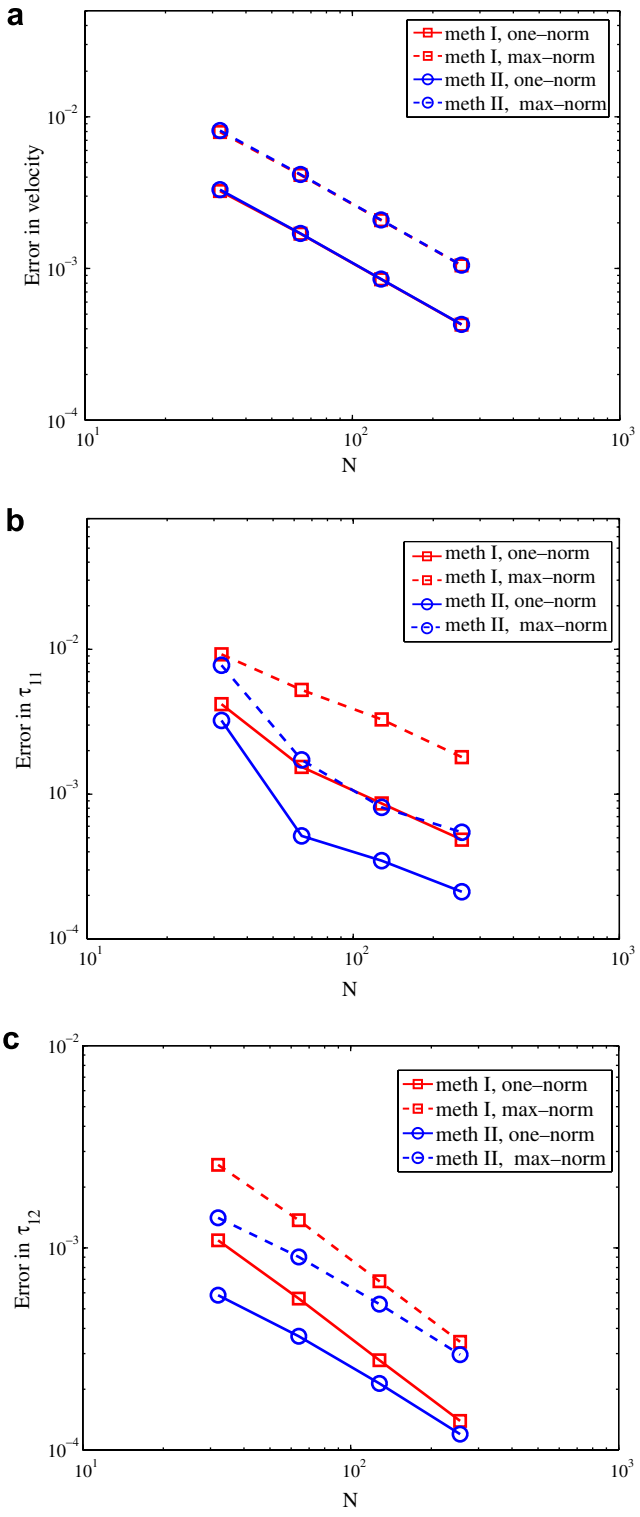

Fig. 4. Results of the refinement study for $G = 1$ and $\beta = 1$. The one-norm and max-norm of the errors are shown for both wave propagation methods for the (a) velocity, (b) τ_{11}, and (c) τ_{12}. The solid lines correspond to the one-norm and the dashed lines correspond to the max-norm. The square markers are used for the averaging method (method I), and the circle markers are used for the method of propagation in a heterogeneous elastic material (method II). The number of grid cells in each direction is denoted by N. The convergence is first-order for both methods, in both norms, and in all variables.

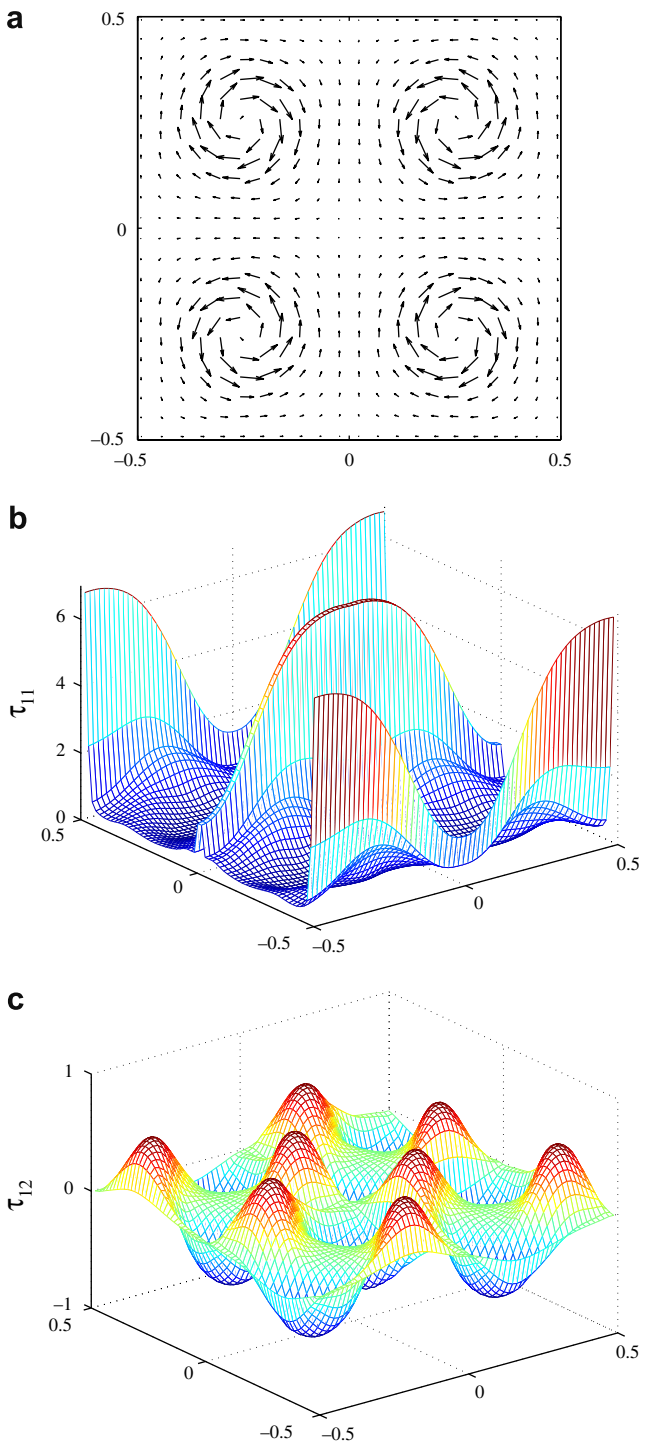

Fig. 5. The (a) velocity field and (b,c) stresses are shown for $G = 0.01$ and $\beta = 0.01$ at time $t = 10$. The stress component τ_{22} is not displayed because it is similar to τ_{11} with an appropriate shift and rotation. The wave propagation was performed using method I.

had the same features (not shown). When the elastic modulus was spatially dependent, but bounded away from zero, the errors from method I (averaging) were generally smaller than for method II (heterogeneous medium), but the results were again similar (not shown). However, in tests containing a viscoelastic fluid in contact with a Newtonian

Please cite this article in press as: R.D. Guy, A.L. Fogelson, A wave propagation algorithm for viscoelastic fluids with spatially ..., Comput. Methods Appl. Mech. Engrg. (2008), doi:10.1016/j.cma.2007.11.022

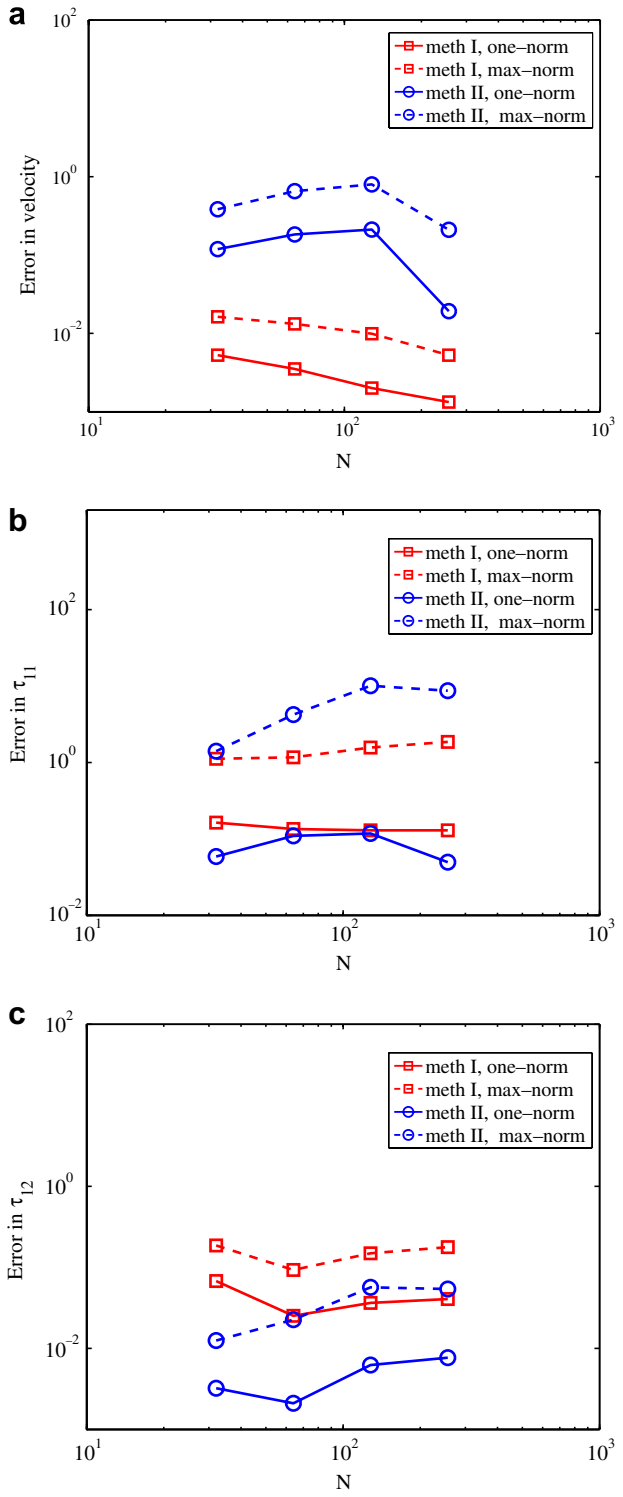

Fig. 6. Results of the refinement study for $G = 0.01$ and $\beta = 0.01$. The one-norm and max-norm of the errors are shown for both wave propagation methods for the (a) velocity, (b) τ_{11}, and (c) τ_{12}. The solid lines correspond to the one-norm and the dashed lines correspond to the max-norm. The square markers are used for averaging method (method I), and the circle markers are used for the method of propagation in a heterogeneous elastic material (method II). The number of grid cells in each direction is denoted by N.

fluid, the methods were very different. These results are presented below.

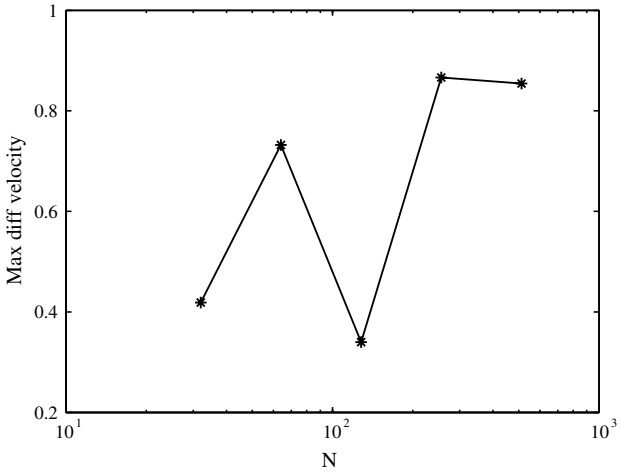

Fig. 7. The max-norm of the difference in the velocity fields produced by the two methods of wave propagation vs. the number of grid cells in each direction (N). The two velocity fields do not converge to each other as the grid is refined.

All parameters given below are nondimensional. The computational domain is periodic in the y-direction with $y \in [-0.5, 0.5]$. The flow is driven by the background force in the x-direction

$$f_{\text{bg}} = \frac{4\pi^2}{\text{Re}} \sin(2\pi(y - 0.5)). \tag{88}$$

For a Newtonian fluid, this force would drive the flow in the x-direction with velocity profile $u = \sin(2\pi(y - 0.5))$. However, for our tests, the middle of the domain contains viscoelastic fluid with the spatially dependent elastic modulus

$$G(y) = \begin{cases} \frac{G_{\max}}{2}\left(1 + \cos\left(\frac{\pi}{2}y\right)\right) & \text{for } |y| < 0.2, \\ 0, & \text{otherwise.} \end{cases} \tag{89}$$

At steady state the velocity and stress satisfy the equations

$$\text{Re}^{-1} u_{yy} + (\tau_{12})_y + f_{\text{bg}} = 0, \tag{90}$$

$$2\tau_{12} u_y - \beta \tau_{11} = 0, \tag{91}$$

$$G(y) u_y - \beta \tau_{12} = 0 \tag{92}$$

and v and τ_{22} are both zero. This steady state system was solved using a finite difference method with an extremely fine mesh ($N = 4096$), and large time results from the two wave propagation methods (with $N = 64$) were compared with this solution. The fluid parameters used were $Re = 25$, $G_{\max} = 10$, and $\beta = 1$.

In Fig. 8, we show the velocity and stress profiles for the top half of the domain. Above the dashed line the fluid is Newtonian. The line without markers represents the "exact" solution generated by the finite difference method. The lines with markers are the solutions produced using the two wave propagation algorithms. The square markers correspond to method I (averaging), and the circles correspond to method II (heterogeneous medium). In Table 1, we give the one-norm and max-norm errors for the two

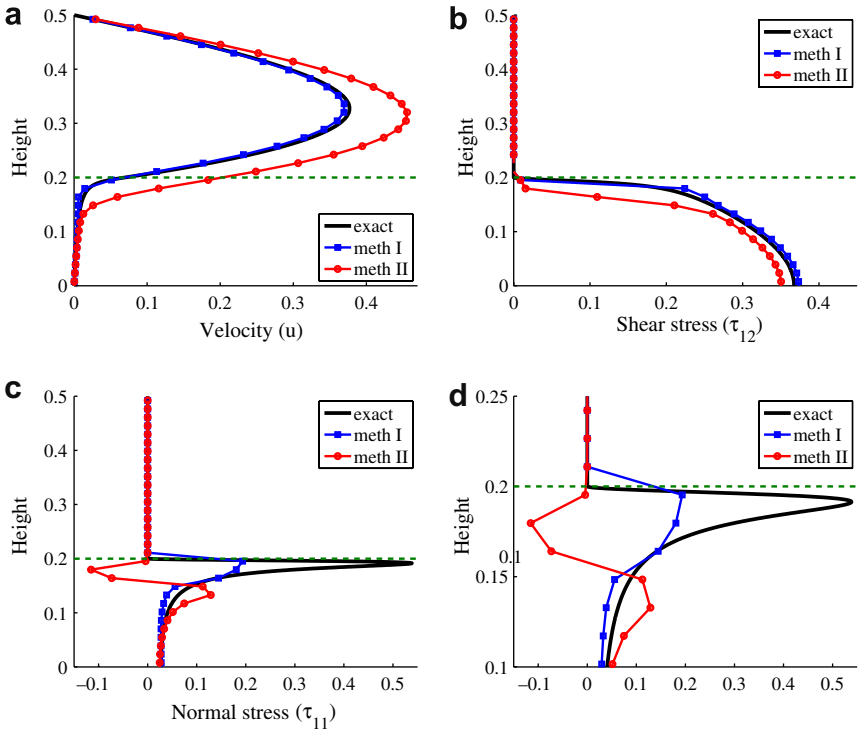

Fig. 8. Velocity and stress profiles for shear flow of a Newtonian fluid over a viscoelastic fluid. Above the dashed line the fluid is Newtonian. The solid line with no markers denotes the exact solution. The line marked with squares corresponds to the wave propagation solution with averaging to the edges (method I), and the line marked with circles corresponds to wave propagation through a heterogeneous elastic medium (method II). In (d) we show a closer view of the results for τ_{11}.

Table 1
Errors in the shear flow test for the two wave propagation algorithms

	Method I	Method II
u		
$\|e\|_1$	3.92×10^{-3}	4.87×10^{-2}
$\|e\|_\infty$	9.56×10^{-3}	1.35×10^{-1}
τ_{11}		
$\|e\|_1$	4.83×10^{-3}	1.72×10^{-2}
$\|e\|_\infty$	4.20×10^{-2}	1.79×10^{-1}
τ_{12}		
$\|e\|_1$	1.30×10^{-2}	3.62×10^{-2}
$\|e\|_\infty$	1.93×10^{-1}	4.14×10^{-1}

methods. These results show that the averaging approach is much more accurate. Comparing the normal stresses (τ_{11}) near the interface where the elastic modulus goes to zero, we see that the averaging approach does a much better job capturing the behavior of the solution.

5.3. Comparing with previous algorithm

The algorithm used in [1] and described more thoroughly in [2], is based on a different splitting of the system in which the viscoelastic stress and the fluid velocity are updated alternately. This splitting is easy to implement, but in some simulations grid-scale oscillations developed over time. Reducing the time step eliminated these oscillations, but it is not clear how to choose the time step in advance to avoid them. This phenomenon was the motivation for exploring the wave propagation algorithms described in this paper. Using the wave propagation algorithm, we are able to compute with a time step very near the stability limit without introducing artificial numerical oscillations. This is demonstrated below.

We use a test problem similar to that presented in Section 5.1, except that the elastic modulus is not uniform. The domain is again the periodic box $[-0.5, 0.5] \times [-0.5, 0.5]$, the fluid is initially at rest, and the motion is driven by the force (87). The fluid is viscoelastic in a circle centered at the origin with radius 0.175, and outside this circle the elastic modulus is initially zero. Inside the circle, the elastic modulus is $G = 50$, which is much higher than in the previous tests. The Reynolds number is 25 and the relaxation rate is 1. The simulation is run until time $t = 3$.

The flow pattern is shown in Fig. 9a, where the single contour represents the boundary between the viscoelastic fluid and the Newtonian fluid. In Fig. 9b, we show a slice of the shear stress, τ_{12}, along the line $x = 0.15$. The results labeled "old algorithm" were generated using the algorithm from [1] which alternates velocity and stress updates. The time step used in the old algorithm was fixed at 10^{-3}. The time step for the wave propagation algorithm was chosen to run at 0.95 of the CFL condition, and was around 2×10^{-3} for the parameters in this test. Both algorithms were stable, but in the old algorithm, some numerical oscillations

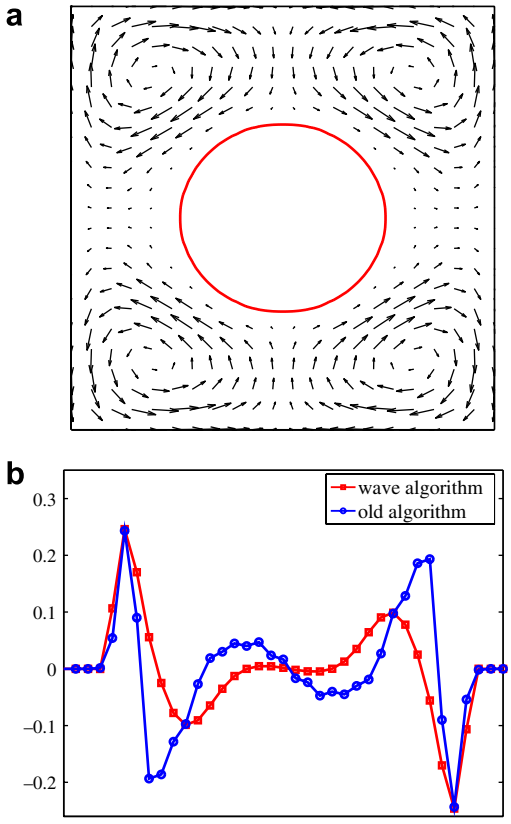

Fig. 9. (a) The contour line approximates the boundary between the viscoelastic fluid and the Newtonian fluid. (b) Slice of the shear stress τ_{12} along the line $x = 0.15$. The stress from the wave propagation algorithm does not show the grid-scale oscillations as does the algorithm from [2].

accumulated. In the wave algorithm, no such oscillations were observed, even though the wave algorithm was run with a larger time step.

6. Conclusions

The algorithm for simulating the platelet model presented in this paper was developed as an improvement to the algorithm used in [1,2]. In simulations with the old algorithm, grid-scale oscillations would accumulate slowly. These oscillations could be eliminated by reducing the time step, but it is unclear how small a time step is needed to avoid them. The wave propagation algorithm does not give these oscillations as long as the time step is chosen to satisfy a CFL constraint. This not only has the advantage of better resolved solutions, but it allows the use of a variable time step. In simulations of growing blood clots, the time step can be larger in the beginning of the simulation before the clot develops. This feature makes for more efficient simulations.

Two different methods of propagating waves were explored, one based on linearizing about the cell edges and the other based on wave propagation in a heterogeneous elastic medium [11]. In tests with a uniform elastic modulus and at low to moderate Weissenberg numbers, the two methods gave very similar results. When the domain contained both a viscoelastic fluid and a Newtonian fluid, the two methods gave very different results. The method of averaging the stresses to the edges gave more accurate results and more accurately captured the behavior of the stresses near the interface where the elastic modulus went to zero. It may be surprising that the method based on heterogeneous elastic materials did not perform well given the success reported in [11]. However, there are some notable differences between the system, (72) and (73), in this paper and the system in [11]. First the coefficients of the system in [11] did not depend on the state variables as they do in (72) and (73). Second the elastic modulus moves with the fluid in the viscoelastic flow system but was stationary for the system in [11].

Though developed for the platelet model, the algorithm applies to more general viscoelastic fluids. Simulating at high Weissenberg number is a significant challenge for algorithms for viscoelastic flow. Our algorithm showed no instabilities at high Weissenberg number, but the results from a refinement study suggested that the results may not be very accurate. It would be interesting to test our wave propagation algorithm using some of the standard benchmark problems for viscoelastic flows. It is worth noting that, as presented, our algorithm does not apply to zero Reynolds number flow. This is because the system in which we propagate waves would no longer be hyperbolic in that parameter range.

Appendix A. Enforcing positive definiteness

As discussed in Section 4.3, when the tensor $\boldsymbol{\tau} + G\boldsymbol{\delta}$ fails to be positive definite, we replace this tensor with the closest symmetric positive semidefinite tensor (s.p.s.d.). Here we describe how to choose this replacement tensor.

Let $\boldsymbol{\sigma}$ be a symmetric tensor. We claim that the closest in two-norm symmetric positive semidefinite tensor is $\boldsymbol{\sigma}^+$. Suppose that $\hat{\boldsymbol{\sigma}}$ is a symmetric positive definite tensor that is closer to $\boldsymbol{\sigma}$ in two-norm than $\boldsymbol{\sigma}^+$. Then

$$\|\boldsymbol{\sigma} - \hat{\boldsymbol{\sigma}}\|_2 \leqslant \|\boldsymbol{\sigma} - \boldsymbol{\sigma}^+\|_2 = \|\boldsymbol{\sigma}^-\|_2 = |\lambda^-|, \tag{A.1}$$

where λ^- is the largest (in absolute value) negative eigenvalue of $\boldsymbol{\sigma}$.

Let $\boldsymbol{v}$ be the normalized eigenvector of $\boldsymbol{\sigma}$ such that

$$\boldsymbol{v}^{\mathrm{T}}\boldsymbol{\sigma}\boldsymbol{v} = \lambda^-. \tag{A.2}$$

We can bound the quadratic form $|\boldsymbol{v}^{\mathrm{T}}(\boldsymbol{\sigma} - \hat{\boldsymbol{\sigma}})\boldsymbol{v}|$ from below by

$$|\boldsymbol{v}^{\mathrm{T}}(\boldsymbol{\sigma} - \hat{\boldsymbol{\sigma}})\boldsymbol{v}| = |\lambda^- - \mu|, \quad \mu > 0, \tag{A.3}$$

$$> |\lambda^-|. \tag{A.4}$$

The same quadratic from is bounded above by

$$|\boldsymbol{v}^{\mathrm{T}}(\boldsymbol{\sigma} - \hat{\boldsymbol{\sigma}})\boldsymbol{v}| \leqslant \|\boldsymbol{v}\|_2 \|(\boldsymbol{\sigma} - \hat{\boldsymbol{\sigma}})\boldsymbol{v}\|_2, \tag{A.5}$$

$$\leqslant \|\boldsymbol{\sigma} - \hat{\boldsymbol{\sigma}}\|_2, \tag{A.6}$$

$$\leqslant |\lambda^-|, \tag{A.7}$$

Please cite this article in press as: R.D. Guy, A.L. Fogelson, A wave propagation algorithm for viscoelastic fluids with spatially ..., Comput. Methods Appl. Mech. Engrg. (2008), doi:10.1016/j.cma.2007.11.022

which contradicts the previous inequality. Therefore, no symmetric positive definite tensor is closer to $\boldsymbol{\sigma}$ than $\boldsymbol{\sigma}^{+}$.

References

[1] A.L. Fogelson, R.D. Guy, Immersed-boundary-motivated models of intravascular platelet aggregation, Comput. Methods Appl. Mech. Engrg., in press, doi:10.1016/j.cma.2007.06.030.
[2] A.L. Fogelson, R.D. Guy, Platelet–wall interactions in continuum models of platelet thrombosis: formulation and numerical solution, Math. Med. Biol. 21 (2004) 293–334.
[3] R.G. Larson, Constitutive Equations for Polymer Melts and Solutions, Butterworth, Stoneham, MA, 1988.
[4] R.J. LeVeque, Finite Volume Methods for Hyperbolic Problems, Cambridge University Press, Cambridge, 2002.
[5] J.B. Bell, P. Colella, H.M. Glaz, A second-order projection method for the incompressible Navier–Stokes equations, J. Comput. Phys. 85 (1989) 257–283.
[6] R.G. Owens, T.N. Phillips, Computational Rheology, Imperial College Press, London, 2002.
[7] M.S. Green, A.V. Tobolsky, A new approach to the theory of relaxing polymeric media, J. Chem. Phys. 14 (1946) 80–92.
[8] M. Yamamoto, The viscoelastic properties of network structure: I. General formalism, J. Phys. Soc. Jpn. 11 (1956) 413–421.
[9] N. Phan-Thien, R.I. Tanner, A new constitutive equation derived from network theory, J. Non-Newtonian Fluid Mech. 2 (1977) 353–365.
[10] R.D. Guy, Asymptotic analysis of PTT type closures for transient network models, J. Non-Newtonian Fluid Mech. 123 (2004) 223–235.
[11] R.J. LeVeque, Finite-volume methods for non-linear elasticity in heterogeneous media, Int. J. Numer. Methods Fluids 40 (2002) 93–104.
[12] P. Colella, Multidimensional upwind methods for hyperbolic conservation laws, J. Comput. Phys. 87 (1990) 171–200.
[13] D. Trebotich, P. Colella, G. Miller, A stable and convergent scheme for viscoelastic flow in contraction channels, J. Comput. Phys. 205 (2005) 315–342.
[14] D. Trebotich, P. Colella, G. Miller, A. Nonaka, T. Marshall, S. Gulati, D. Liepmann, A numerical algorithm for complex biological flow in irregular microdevice geometries, technical report UCRL-JC-150969, Lawrence Livermore National Laboratory, 2004.
[15] J. Sun, N. Phan-Thien, R.I. Tanner, An adaptive viscoelastic stress splitting scheme and its applications, J. Non-Newtonian Fluid Mech. 65 (1996) 281–307.
[16] D.D. Joseph, Fluid Dynamics of Viscoelastic Liquids, Springer, New York, 1990.
[17] R.J. LeVeque, High-resolution conservative algorithms for advection in incompressible flow, SIAM J. Numer. Anal. 33 (1996) 627–665.
[18] A.S. Almgren, J.B. Bell, W.G. Szymczak, A numerical method for the incompressible Navier–Stokes equations based on an approximate projection, SIAM J. Sci. Comput. 17 (1996) 358–369.

Please cite this article in press as: R.D. Guy, A.L. Fogelson, A wave propagation algorithm for viscoelastic fluids with spatially ..., Comput. Methods Appl. Mech. Engrg. (2008), doi:10.1016/j.cma.2007.11.022